A Multilingual Reading Comprehension System for more than 100 Languages

Anthony Ferritto[1*] **Sara Rosenthal**[1*] **Mihaela Bornea**[1] **Kazi Hasan**[2]
Rishav Chakravarti[1] **Salim Roukos**[1] **Radu Florian**[1] **Avirup Sil**[1†]

[1]IBM Research AI [2]IBM Watson
aferritto@ibm.com,
{sjrosenthal, mabornea, kshasan, rchakravarti, roukos, raduf, avi}@us.ibm.com

Abstract

This paper presents M-GAAMA, a Multilingual Question Answering architecture and demo system. This is the first multilingual machine reading comprehension (MRC) demo which is able to answer questions in over 100 languages. M-GAAMA answers questions from a given passage in the same or a different language. It incorporates several existing multilingual models that can be used interchangeably in the demo such as M-BERT and XLM-R. The M-GAAMA demo also improves language accessibility by incorporating the IBM Watson machine translation widget to provide additional capabilities to the user to see an answer in their desired language. We also show how M-GAAMA can be used in downstream tasks by incorporating it into an END-TO-END-QA system using CFO (Chakravarti et al., 2019). We experiment with our system architecture on the Multi-Lingual Question Answering (MLQA) and the CORD-19 COVID (Wang et al., 2020; Tang et al., 2020) datasets to provide insights into the performance of the system.

1 Introduction

Recent advances in open domain question answering (QA) have mostly revolved around machine reading comprehension (MRC) (Rajpurkar et al., 2018; Yang et al., 2018). The MRC task is to read and comprehend a given text and then answer questions based on it. Our monolingual MRC approach (Pan et al., 2019) has the capability of being applied to train many Language Models (LMs) such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019). We achieve the 2nd rank[1] on the Google Natural Questions (Kwiatkowski et al., 2019) leaderboard[2]. In this paper, we expand our approach by introducing new multilingual capabilities using models such as Multilingual-BERT (M-BERT) (Devlin et al., 2019) and XLM-R (Conneau et al., 2019). This addition has the capability of transcending language boundaries to *104* languages. Figure 1 shows examples of QA pairs from the MLQA dataset (Lewis et al., 2019). To the best of our knowledge, this is the first published demo of a Multi-Lingual QA system. We achieve this by introducing a novel multilingual component to our QA GAAMA (Go Ahead, Ask Me Anything) (Chakravarti et al., 2019) pipeline.

We introduce M-GAAMA, a new system that performs cross-lingual MRC where a given question and context can be in the same or different languages. The system extracts the answer from the context language. Then, the demo utilizes the SOTA IBM Watson machine translation widget to return the answer translated in the desired language of the user[3]. This breaks the language barrier for users who don't understand the given source text but want their question answered effectively and accurately.

In addition, we also show how M-GAAMA can be used in downstream tasks by incorporating it with CFO (Chakravarti et al., 2019), in an end-to-end QA system and demo. We show that this can be extended to perform multilingual QA by utilizing a language identifier to first gather the (target) language in which

[*] Equal Contribution
[†] Corresponding author.

[1]At the time of writing of this paper.
[2]https://ai.google.com/research/NaturalQuestions/leaderboard
[3]Currently available in 36 languages.

Proceedings of the 27th International Conference on Computational Linguistics, pages 41–47
Barcelona, Spain (Online), December 12, 2020.

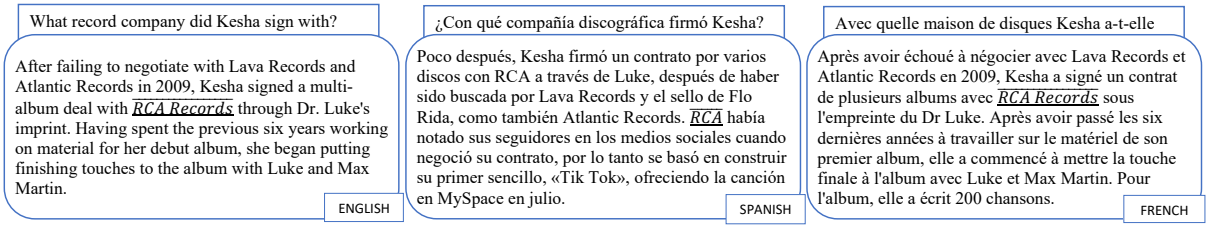

Figure 1: Examples of Q/C pairs about Kesha in three languages that our system answers correctly: English, Spanish, and French. The first two examples originate from the MLQA challenge. The answers are shown as $\overline{answer}$.

the question was asked. END-TO-END-QA then retrieves passages from an index in the appropriate target language and runs our multilingual MRC system on it. Since the answer is extracted from the target language, no translation is required.

We first demonstrate the effectiveness of M-GAAMA on the MLQA dataset and then also show its effectiveness on the CORD-19 (Wang et al., 2020; Tang et al., 2020) corpus which contains research articles regarding COVID-19. The COVID-19 pandemic has caused an abundance of research to be published on a daily basis. Not all of the articles are available in English, and people want to ask questions in their native language. Providing the capability to ask questions on research in all languages is vital for ensuring that important and recent information is not overlooked and available to everyone. We show that M-GAAMA has the capability of providing this information for all language speakers and articles by finding answers in translated CORD-19 articles.

In summary, our contribution is the first published multi-lingual QA demo which works in over 100 languages. It returns an appropriate answer in the language that the question was originally asked. It incorporates several multilingual components including multilingual LMs, machine translation, and indexed corpora in multiple languages.

The rest of the paper is organized as follows: We first discuss related work, then talk about the data used in our experiments and models. Sections 4 and 5 discuss the demo system and model architecture. Finally, we discuss the Model and Runtime Experiments on MLQA (Lewis et al., 2019) and the COVID-19 CORD-19 dataset (Wang et al., 2020; Tang et al., 2020) in Section 6.

2 Related Work

Few other QA demos exist; BERTSerini (Yang et al., 2019), leverages the Anserini IR toolkit (Yang et al., 2017) to extract relevant documents given a question, then uses BERT-based techniques (Devlin et al., 2019) to extract the correct answer. However, their demo is designed to perform only mono-lingual English QA. The GAAMA and CFO (Chakravarti et al., 2019) demos also only performs English QA. In contrast, M-GAAMA and our downstream END-TO-END-QA task perform cross-lingual QA.

Several cross lingual large scale representations have been created by training a large scale transformer (Vaswani et al., 2017) based masked language model on text in multiple languages. The use of pretrained multilingual language models such as M-BERT (Devlin et al., 2019), XLM (Lample and Conneau, 2019), and XLM-R (Conneau et al., 2019) achieve the previous SOTA on cross-lingual tasks including question answering (Lewis et al., 2019) (Conneau et al., 2019). We train our underlying MRC system with these pre-trained language models and achieve results that are consistently as strong as prior work.

Many datasets for English MRC have been introduced with annotated Wikipedia documents including (Rajpurkar et al., 2016; Rajpurkar et al., 2018; Yang et al., 2018; Kwiatkowski et al., 2019). Fewer resources are available for the cross-lingual setting. The MLQA (Lewis et al., 2019) dataset contains parallel instances in 7 languages where the context is found in Wikipedia. The TyDiQA (Clark et al., 2020) dataset containes instances in 11 languages. However, TyDiQA is not parallel and it only has instances where the question and context are in the same language.

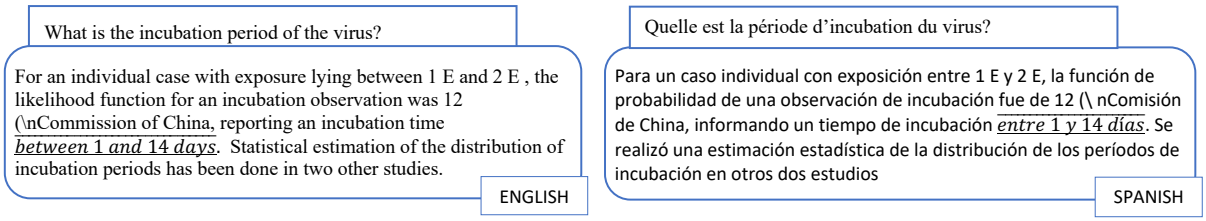

Figure 2: Examples of Q/C pairs about COVID-19. The answers are shown as $\overline{answer}$.

3 Data

We test the multilingual capabilities incorporated into our QA system by running experiments on the MLQA dataset (Lewis et al., 2019). The dataset consists of seven languages: English (en), Spanish (es), German (de), Arabic (ar), Hindi (hi), Vietnamese (vi), and Chinese (zh). To achieve a multilingual parallel QA benchmark the authors apply a novel alignment strategy on Wikipedia articles by identifying Wikipedia sentences with the same meaning in multiple languages. Passages containing these sentences are then presented to the annotators who write questions that are now answerable in multiple languages. We consider this a good resource for evaluating multilingual capabilities on different pairs of languages (e.g., context (c) in English, question (q) in German) due to the parallel q/c pairs available in the corpus. In addition, we use SQUAD 1.1 (Rajpurkar et al., 2016), which is significantly larger, but only contains English data, for training in a zero-shot scenario.

We also explore QA on COVID-19 articles in a zero-shot scenario to show the relevance and importance of multilingual QA in current events. CovidQA 0.1 (Wang et al., 2020) contains 124 question and document pairs. The dataset comprises of (question, scientific article, exact answer) triples that have been manually created from the literature review page of Kaggles COVID-19 Open Research Dataset Challenge (Tang et al., 2020). They manually identified the exact answer span as a verbatim extract from the document. We converted their data into SQUAD format for our experiments. We create a multilingual COVID-19 QA dataset using machine translation. We translate both the questions and context in Spanish and Chinese. We align the gold answer between the English and the translated dataset by marking the gold answer with pseudo-HTML tags prior to translation. We recovered the translated answers for all questions. An example of a QA pair in English and Spanish is shown in Figure 2.

4 Demo

In this section we describe the interface for our M-GAAMA demo. We then show an END-TO-END-QA demo as an example that builds upon M-GAAMA using prior work (Chakravarti et al., 2019).

M-GAAMA is a gRPC (Talvar, 2016) server which wraps our LMs for MRC. M-GAAMA provides an MRC interface which can answer questions in over 100 languages. We use M-BERT and XLM-R LMs to drive M-GAAMA's multilingual support. In addition, we provide a Language Translation component made available as a Javascript widget[4] to allow the user to see the answer in the question language or any other language of choice. The M-GAAMA interface weaves the components together using the ReactJS framework[5]. Providing M-GAAMA as a gRPC server allows it to be quite flexible. This enables it to seamlessly transition between being a standalone system and integrating with larger systems. We show this via the downstream END-TO-END-QA task described below.

END-TO-END-QA builds upon M-GAAMA, with a full IR-MRC pipeline. Information Retrieval is obtained using an Elasticsearch index[6] for each language[7]. The user can ask a question in any language for which an index exists. The language of the question is identified using the 'langid' toolkit (Lui and Baldwin, 2012) to determine the appropriate index. The appropriate index is then searched for documents in the target language. These documents are then evaluated together with the user's question

[4] https://www.ibm.com/watson/services/language-translator/
[5] https://reactjs.org/
[6] https://hub.docker.com/_/elasticsearch/
[7] In our implementation we built an index in English and Spanish as a proof of concept.

	MLQA									COVID-19		
F1	en	es	de	ar	hi	vi	zh	XLT	G-XLT	en	es	zh
ROBERTA$_L$	84.4	-	-	-	-	-	-	-	-	27.1	-	-
M-BERT	80.4	66.7	61.3	51.9	50.7	61.6	60.2	61.8	52.1	22.3	17.0	20.5
XLM-R$_B$	80.1	67.6	63.0	56.3	61.1	66.2	61.6	65.1	41.2	21.9	19.0	17.0
XLM-R$_L$	**83.9**	**74.0**	**69.9**	**66.3**	**71.2**	**74.0**	**69.9**	**72.7**	**67.9**	**27.0**	**28.7**	**25.0**

Table 1: (Left) F1 score on the MLQA test set for the cross-lingual transfer task (XLT) per language and the mean XLT and G-XLT scores. Training data is SQUAD 1.1. *B* is the Base model and *L* is the Large model. (Right) F1 XLT scores on the CORD-19 dataset when training on SQUAD 1.1 in three languages.

by M-GAAMA. Finally, answer spans are de-duplicated and sorted by score before being returned to the user. The END-TO-END-QA demo weaves these components together using the CFO framework (Chakravarti et al., 2019), which is a novel approach for orchestrating services.

5 Model Architecture

Our MRC QA model accepts a single query-document pair as its input and produces a span from the document along with a prediction score as its output. The underlying QA model is based on (Pan et al., 2019). The base layer of the QA system encodes the question and the candidate paragraph using the cross-lingual M-BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2019) representations. An output feed forward layer is added on top of the base layer to produce 3 sets of scores: scores at each token offset marking the likelihood of an answer chunk (1) starting at this offset, (2) ending at this offset, and (3) the entire sequence marking the likelihood of the question being answerable given the current context.

6 Experiments

We experiment with several multi-lingual models on the MLQA test set prior to integration in our system. We explore zero-shot learning as in prior work (Lewis et al., 2019) by training and fine-tuning on SQUAD 1.1 (Rajpurkar et al., 2016) for M-BERT and the XLM-R QA models. Refer to (Chakravarti et al., 2019; Pan et al., 2019) for additional details about model architecture and implementation. We train our models using the Huggingface code[8] with the default parameters except 3e-5 learning rate, 2 training epochs, 32 batch size and, 790 warmup steps.

We show results for the cross-lingual task (XLT), where the question and context are in the same language (e.g. question (q) and context (c) in Chinese) on the left side of Table 1. We find that our comparable re-implementations of models reported in prior work (Lewis et al., 2019) perform significantly better. We expect the improvement is due to using the Hugging Face implementation and hyper-parameter tuning values. Our best results using XLM-R large are consistently as strong as prior work in all languages. We also show results for the generalized cross-lingual task (G-XLT) where the question and context are in different languages in Table 1; XLM-R large achieves the best results in this experiment as well. We also compare the performance of the multilingual models with the performance of the English ROBERTA large model on the English MLQA dataset and find the results are similar.

We also provide additional analysis for the MLQA results by showing the difference per each question type in Table 2, for the XLT task. Having this information is useful for understanding which question types should be explored in more detail. We notice that the XLM-R$_L$ performance is more stable across all question types. All systems obtain the best performance for the "when" questions and the lowest for the "why" questions. We expect this is because "why" questions are more of an explanation making them more challenging while "when" questions tend to be easier because they are usually dates or numbers. We determine the question type by examining the English questions. Since MLQA has parallel examples, we used the question id to determine the question type when the question is in different languages.

Further, we show the value of having a multilingual model by also exploring QA for COVID-19 using the CORD-19 (Wang et al., 2020; Tang et al., 2020) dataset on English and translated data in Spanish and Chinese. The ability to answers questions in other languages is especially important in this use-case

[8] https://github.com/huggingface/transformers

F1	who	why	where	what	how	which	when	other	avg
M-BERT	64.6	44.2	54.9	61.9	62.4	62.6	**64.9**	61.0	61.8
XLM-R$_B$	68.4	56.9	60.7	63.3	67.4	64.1	**75.9**	65.5	65.1
XLM-R$_L$	76.5	66.7	68.9	70.9	74.4	71.9	**81.7**	71.9	72.7

Table 2: F1 score on the MLQA test set for the cross-lingual transfer task (XLT). Training data is SQUAD 1.1. *B* is the Base model and *L* is the Large model. The best performing question types are shown in **bold**. We also include the XLT averages from Table 1 for comparison.

Context	Question	# Examples	T_{GPU}	T_{CPU}
hi	ar	186	50	721
en	de	512	35	1525
zh	hi	189	61	628
en	ar	517	69	1215
zh	ar	188	57	615

Table 3: Dev Set Performance on MLQA benchmarked on the CPU and GPU. Times in seconds.

because the corpus is rapidly growing and some papers may only be available in a single language. The results are shown on the right side of Table 1. Although the overall performance is lower than MLQA, results are consistent across languages. XLM-R is still the best performing model. In contrast to passage level QA in MLQA and SQUAD, the CORD-19 dataset is document level. We expect this causes a large detriment to the performance.

Finally, while the best performing model is XLM-R large, there is merit to including the M-BERT model in the demo due to its reduced size which makes deployment more scalable. We use our M-BERT model for runtime experiments. A single x86-64 Intel® core is used as the CPU whereas one Nvidia® Tesla® V100 is used as the GPU. For brevity we show results for a random subset of five context question language pairs in Table 3. As expected running the model on GPU is faster than CPU: on average a given language pair is processed 19 times faster on GPU than CPU as shown in Table 3. The GPU also produces more consistent runtimes than CPU: standard deviation in CPU runtimes for each language pair is 32 times more than on GPU. We also find that not all languages decode equally quickly. Language pairs including English, particularly as the context, are the quickest to decode on GPU. Chinese and Hindi contexts take 2 to 3 times as long. The same trend holds on CPU, where the multiplier is approximately 1.5. Additionally, these differences are not fully explained by differing context sizes. On average Chinese and Hindi contexts are 1.4 and 0.9 times as long as their English counterparts respectively as seen in Table 4 of (Lewis et al., 2019). Question sizes are an order to two of magnitude shorter than contexts. This indicates that some languages decode faster than others even when accounting for context sizes.

7 Conclusion

In this paper we present our M-GAAMA demo, an interface for interacting with our multilingual QA MRC system. To the best of our knowledge we are the first to present a QA demo with multilingual capabilities in over 100 languages. We enable the user to be able to ask a question in one language, find the answer in another language, and with the use of machine translation the user can see the answer in the question language or another desired language. We also show how M-GAAMA can be used in a downstream task in our END-TO-END-QA demo. Finally, we show that our system achieves results that are consistently as strong as prior work on the MLQA dataset (Lewis et al., 2019) using XLM-R-Large on all seven languages. It can also be used to perform QA in current events via the CORD-19 COVID-19 (Wang et al., 2020; Tang et al., 2020) dataset. In the future we plan on experimenting with additional QA datsets such as Natural Questions (Kwiatkowski et al., 2019) and TyDiQA (Clark et al., 2020).

8 Acknowledgements

We would like to thank Andy Sakrajda for the help with IBM Watson Translation pipeline and Vittorio Castelli and Cezar Pendus with the help in building the multiligual search corpus. We would also like to thank the authors of the MLQA and XLM-R paper for helping us by sharing the hyper-parameters to repeat some of their experiments and help us while we debug the XLM-R models for Chinese.

References

Rishav Chakravarti, Cezar Pendus, Andrzej Sakrajda, Anthony Ferritto, Lin Pan, Michael Glass, Vittorio Castelli, J William Murdock, Radu Florian, Salim Roukos, and Avirup Sil. 2019. CFO: A framework for building production nlp systems. *EMNLP-IJCNLP, Demo Track*.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *TACL*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

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

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. *TACL*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*.

Patrick Lewis, Barlas Ouz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. Mlqa: Evaluating cross-lingual extractive question answering.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea, July. ACL.

Lin Pan, Rishav Chakravarti, Anthony Ferritto, Michael Glass, Alfio Gliozzo, Salim Roukos, Radu Florian, and Avirup Sil. 2019. Frustratingly easy natural question answering.

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

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. *arXiv preprint arXiv:1806.03822*.

Varun Talvar. 2016. grpc design and implementation, 5. Talk by Varun Talwar, Product Manager at Google at Stanford, California [Accessed: 2019 06 20].

Raphael Tang, Rodrigo Nogueira, Edwin M. Zhang, Nikhil Gupta, Phng Ths. Bùi Cm, Kyunghyun Cho, and Jimmy Lin. 2020. Rapidly bootstrapping a question answering dataset for covid-19. *ArXiv*, abs/2004.11339.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*. Curran Associates, Inc.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Michael Kinney, Ziyang Liu, William. Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. Cord-19: The covid-19 open research dataset. *ArXiv*.

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. SIGIR. ACM.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with bertserini.