*Natural Language Processing* (2025), **31**, pp. 346-374
doi:10.1017/nlp.2024.12

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

# EHMMQA: English, Hindi, and Marathi multilingual question answering framework using deep learning

Pawan Lahoti, Namita Mittal and Girdhari Singh

Department of Computer Science and Engineering, Malaviya National Institute of Technology Jaipur, Rajasthan, India.
**Corresponding author:** Pawan Lahoti; Email: lahotipawan@gmail.com

(Received 19 November 2022; revised 6 June 2023; accepted 30 July 2023; first published online 24 May 2024)

Special Issue on '**Natural Language Processing Applications for Low-Resource Languages**'

**Abstract**

Multilingual question answering (MQA) is an effective access to multilingual data to provide accurate and precise answers, irrespective of language. Although a wide range of datasets is available for monolingual QA systems in natural language processing, benchmark datasets specifically designed for MQA are considerably limited. The absence of comprehensive and benchmark datasets hinders the development and evaluation of MQA systems. To overcome this issue, the proposed work attempts to develop the EHMQuAD dataset, an MQA dataset for low-resource languages such as Hindi and Marathi accompanying the English language. The EHMQuAD dataset is developed using a synthetic corpora generation approach, and an alignment is performed after translation to make the dataset more accurate. Further, the EHMMQA model is proposed to create an abstract framework that uses a deep neural network that accepts pairs of questions and context and returns an accurate answer based on those questions. The shared question and shared context representation have been designed separately to develop this system. The experiments of the proposed model are conducted on the MMQA, Translated SQuAD, XQuAD, MLQA, and EHMQuAD datasets, and EM and F1-score are used as performance measures. The proposed model (EHMMQA) is collated with state-of-the-art MQA baseline models for all possible monolingual and multilingual settings. The results signify that EHMMQA is a considerable step toward the MQA system utilizing Hindi and Marathi languages. Hence, it becomes a new state-of-the-art model for Hindi and Marathi languages.

**Keywords:** natural language processing; multilingual question answering; deep neural network; neural machine translation; translate-align-retrieve

## 1. Introduction

Question answering (QA) (Mishra and Jain 2016) extracts accurate, relevant, and factual information (in any natural language) from various sources, such as web pages, natural language documents, or structured databases, to provide answers to user queries. Information extraction (IE) (Chang *et al.* 2006), information retrieval (IR) (Singhal *et al.* 2001), and many related natural language processing (NLP) tasks get benefits from QA. Traditional search engines focus on identifying relevant documents, and QA system aims to deliver precise answers within those documents. Multilingual question answering (MQA) (García-Cumbreras *et al.* 2012) broadens QA for multiple languages, allowing users to communicate with the system using their native language and retrieve information from multilingual documents. While monolingual QA systems are

prevalent, the growing volume of multilingual content on the internet necessitates the development of robust MQA systems. MQA systems aim to overcome the challenges posed by typological, grammatical, and syntactical differences between languages (Clark *et al.* 2020) to provide accurate answers across different languages.

The research community has made progress in developing MQA models. However, the development of MQA systems requires adequate linguistic resources. Currently, English has abundant NLP resources, whereas Hindi and Marathi (ranks third and fifteenth in the world,[a] respectively), despite being widely spoken languages, lack sufficient resources and tools. The research community has made efforts to develop datasets for English and Hindi languages, including the MMQA (Gupta *et al.* 2018), Translated SQuAD (Gupta, Ekbal, and Bhattacharyya 2019), XQuAD (Artetxe, Ruder, and Yogatama 2020), and MLQA (Lewis *et al.* 2020) datasets with a maximum sample of 18,454 in the Translated SQuAD dataset. For the Marathi language, no such efforts have been made to construct the dataset. Therefore, there is a need for an MQA benchmark dataset specifically tailored for the English-Hindi-Marathi language. To address these limitations, a comprehensive dataset for English, Hindi, and Marathi languages is created, employing a synthetic corpus generation approach leveraging the characteristics of Hindi and Marathi languages.

Additionally, the existing approaches for designing an MQA system use zero-shot/few-shot, translate-train, and translate-test to transfer knowledge from English to other languages. These approaches were inefficient in developing a unified QA system to understand the various languages. This motivates us to develop a unified MQA framework for English, Hindi, and Marathi languages. The proposed framework is an abstract structure that adapts to many different languages. The proposed model, EHMMQA, utilizes a deep learning approach with an attention-based recurrent neural network (RNN) to generate representations of questions and context in multiple languages. The model is trained on the proposed EHMQuAD dataset and evaluated on numerous benchmark datasets. The result outperforms baseline models in monolingual and multilingual settings and demonstrates its effectiveness and superiority, emerging as a state-of-the-art model for Hindi and Marathi languages.

For a given question $q$, written in English, Hindi, or Marathi, the MQA system will return the precise answer from comparable documents $\{D_i^N\}$. The key contributions of the proposed model are highlighted as follows:

(1) Developed an MQA benchmark dataset for Indian languages accompanying the English language to access multilingual information.
(2) Designed and implemented an end-to-end MQA model employing an RNN and attention mechanism.
(3) Evaluated the proposed EHMQuAD dataset on the EHMMQA model, and a comparative analysis was conducted to assess the EM and F1-score results compared to several baseline models.

The remainder of the paper is structured as follows. Section 2 reviews related work on monolingual and multilingual QA systems. Section 3 presents the methodology, including dataset development, error analysis of the proposed dataset, and the proposed end-to-end MQA system. Section 4 describes the evaluation metrics and experimental setup for monolingual and multilingual settings. Section 5 analyses the results and conducts an ablation study of the proposed work. Finally, Section 6 presents the conclusion and future work.

[a]https://en.wikipedia.org/wiki/List_of_languages_by_total_number_of_speakers

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

## 2. Related work

MQA has gained popularity since the track's inception by the Cross-Language Evaluation Forum (CLEF[b]). Although most of the MQA systems came up with the machine translation of either of the languages to obtain the answer from the context, the translated results were less accurate (García Santiago and Olvera-Lobo, 2010), which deteriorated the system's performance as the capability of machine translation was low before the evolution of Deep Learning (DL) models (Türe and Boschee, 2016). Recent advancement in the field of DL enhances the accuracy of machine translation for MQA systems; however, the accuracy is still low due to the limited resources for various languages. The summary of the recent work on multilingual QA systems for Indic and non-Indic languages is described in the following subsections.

### *2.1 Multilingual QA systems for Indic languages*

There is a noticeable scarcity of research in the literature pertaining to the application of MQA tasks specifically for Indic languages. Gupta *et al.* (2018) initiated the development of MQA for the Indic language considering only the Hindi-English language. In this work, the author has proposed an MQA system that answers the user's question by identifying the probable snippets from the context and also developed the Multilingual & Multi-domain QA (MMQA) dataset for six domains. MMQA dataset was developed from 500 documents in English and Hindi language containing, 5,495 QA pairs. The author combined the Boolean and BM25 vector space models to retrieve and rank the top 30 passages from which the snippets are extracted to get accurate answers. The proposed dataset solely comprises factoid and short-descriptive question-answer pairs without incorporating any complex questions. To extend this work, Gupta *et al.* (2019) proposed a Deep Neural Network (DNN) based MQA model to generate snippets from passages irrespective of the language. This developed model inputs a factoid question and text from a multilingual passage in either English or Hindi and returns an answer based on the context. In this work, a shared representation of the question is learned using soft alignment between the words from both languages. This shared question representation and the original snippet are received as input to the answer extraction layer to generate the answer span. This extraction process considers the relevance of the question and snippet.

In addition, Gupta *et al.* (2019) developed a Translated SQuAD dataset to train the model and evaluated it on the Translated SQuAD and MMQA datasets. For evaluating the MMQA dataset, a novel snippet generation algorithm is proposed that takes questions and multiple documents as input and extracts a snippet that supports evidence containing the answer(s). During this process, a semantic graph for each document is generated (accounting WordNet dictionary and word embeddings) that possesses a lexico-semantic relationship between the pairs of sentences. The model cannot predict the correct answer for various reasons such as anaphora resolution problem, NER proximity error, and translation errors.

Lewis *et al.* (2020) presented the MLQA evaluation benchmark dataset, which uses a multiway aligned extractive technique to evaluate the QA system. It is created for seven languages: English (en), Arabic(ar), German (de), Hindi(hi), Simplified Chinese(zh), Spanish (es), and Vietnamese (vi). The author employed fine-tuning, translate-train, and translate-test approach to design an MQA system where the MLQA dataset is fine-tuned on BERT-Large (Devlin *et al.* 2019), mBERT, and Cross-lingual Language Model (XLM) (Conneau and Lample, 2019) word representation models. While constructing QA pairs, the annotator focuses on a single sentence rather than multiple sentences. This leads to a reduction in sample size for languages other than English.

Another MQA dataset covering typologically diverse languages worldwide named TyDiQA (Clark *et al.* 2020) is created covering typological features of these languages. It comprises 204K

[b]https://www.clef-initiative.eu/

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

QA pairs for 11 languages, including Telugu (te) and Bengali (bn) as Indic languages. The dataset is prepared using a crowd-sourced method instead of a machine translation approach. The qualitative analysis evaluates the linguistic phenomenon of these diverse languages in terms of data quality and example-level quality. The dataset's quality is evaluated using a first passage baseline, mBERT model, and a human evaluator for the passage selection task. In contrast, only mBERT and human evaluation are used for minimal answer span.

### 2.2 Multilingual QA systems for non-Indic languages

Extensive research has also been conducted for non-Indic languages, employing diverse approaches. The subsequent section provides a detailed examination of their endeavors. The work of Chandra *et al.* (2021) presents a thorough analysis of the advancements made in non-English Machine Reading Comprehension (MRC) and open-domain question answering (QA) datasets. The researchers reviewed QA datasets comprehensively across 14 prominent languages. In a notable study by Rogers *et al.* (2023), significant contributions of monolingual and multilingual QA/RC resources beyond the English language are surveyed, offering valuable insights into their typology, scope, and language coverage.

Carrino *et al.* (2020) proposed the Translate Align Retrieve (TAR) method to translate the SQuAD v1.1 dataset to the Spanish (es) language automatically. The author has developed SQuAD-es v1.1 (87595 QA pair) and SQuAD-es-small (46260 QA pair), where the first dataset contains all the translated examples with alignment, and the latter one keeps only those QA pairs without alignment. The author trained the Spanish QA systems by fine-tuning an mBERT model on these datasets. Experiments were performed on the MLQA and XQuAD benchmark datasets to evaluate this QA model.

MKQA-Multilingual Knowledge Question Answering (Longpre, Lu, and Daiber 2021) is a large MQA dataset containing 260K QA pairs for 26 typologically diverse languages. The author used zero-shot and translation methods to extract the answers. During the experiment, the MKQA dataset is fine-tuned by various word representation models such as mBERT, RoBERTa (Liu *et al.* 2019), XLM-R (Conneau *et al.* 2020), and mT5 (Xue *et al.* 2021). During the analysis of the MKQA dataset, it was observed that a few answers were missing from QA pairs. This issue arose due to challenges associated with the Retrieval-Independence problem and the re-annotation process. This led to difficulty capturing and including all answers in the dataset.

SK-QuAD (Hládek *et al.* 2023) is a human-annotated MQA dataset for the Slovak (sk) language comprising 91K QA pairs from various domains. This dataset follows the structure of SQuADv2.0 (Rajpurkar, Jia, and Liang 2018) dataset, where some QA pairs are unanswerable. The zero-shot approach is used along with the mBERT word representation model to evaluate the performance of the developed dataset. For annotation purposes, only two annotators were used instead of an odd number of annotators, which can cause ambiguity in selecting QA pairs. The comprehensive literature review witnessed that most of the MQA datasets and developed MQA system comprise only one language apart from English. This makes these systems either bilingual or cross-lingual but not multilingual.

## 3. Proposed methodology

To overcome the drawbacks found in the literature, a novel methodology is adopted for developing the EHMQuAD dataset (Section 3.1) and end-to-end MQA system (Section 3.3), considering English, Hindi, and Marathi languages. The following subsections elaborate the methodology including dataset development, error analysis of the proposed dataset, and the proposed end-to-end MQA system. Table 1 sums up the acronym and notation used in work for readability purposes.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 1.** List of acronyms and notations used in the work

| Acronyms | | Notations | |
|---|---|---|---|
| Abbreviation | Full Form | Symbols | Description |
| NLP | Natural Language Processing | $q$, $c$ and $a$ | Question, context, and answer |
| MQA | Multilingual Question Answering | $D_i^N$ | Collection of document |
| QA | Question Answering | $pr_n$ and $w_n$ | Precision and positive weight of n-gram |
| EHMQuAD | English Hindi Marathi Question Answering Dataset | $ng$ | Length of n-gram (1-unigram, 2-bigram, and so on) |
| MMQA | Multi-domain Multi-lingual Question-Answering | $\hat{y}$ and $y$ | Predicted translated sentence and Gold standard alignment sentence |
| SQuAD | Stanford Question Answering Dataset | $q_{en}$, $c_{en}$ and $a_{en}$ | Question, context, and answer of English language |
| XQuAD | Cross-lingual Question Answering Dataset | $q_{hi}$, $c_{hi}$ and $a_{hi}$ | Question, context, and answer of Hindi language |
| MLQA | MultiLingual Question Answering | $q_{mr}$, $c_{mr}$ and $a_{mr}$ | Question, context, and answer of Marathi language |
| RNN | Recurrent Neural Network | $a^{begin}$ and $a^{end}$ | Start and end position of answer |
| MRC | Machine Reading Comprehension | $TA_{en-hi/mr}$ | Translation of English to Hindi/Marathi with sentence alignment |
| MT | Machine Translation | $qw_1^{q_{en}}$, $qw_1^{q_{hi}}$ and $qw_1^{q_{mr}}$ | First word of English, Hindi, and Marathi question |
| TAR | Translate-Align-Retrieve | $x_{en}$, $x_{hi}$ and $x_{mr}$ | Number of word in English, Hindi, and Marathi question |
| NMT | Neural Machine Translation | $y_{en}$, $y_{hi}$ and $y_{mr}$ | Number of word in English, Hindi, and Marathi context |
| LSTM | Long Short-Term Memory | $e_t$ and $c_t$ | Word embedding and character embedding |
| Bi-LSTM | Bidirectional Long Short-Term Memory | $E_i$ | Concatenation of word and character embedding |
| BLEU | BiLingual Evaluation Understudy | $u_t^{q_k}$ and $u_t^{c_k}$ | Final word representation of question and context; $k \in \{en, hi, mr\}$ |
| BP | Brevity Penalty | $R_t$ | Shared representation of a particular language considering other two languages |
| OOV | Out-Of-Vocabulary | $R_t^q$ and $R_t^c$ | Shared question and shared context representation of all three languages |
| Bi-GRU | Bidirectional Gated Recurrent Unit | $S_t$ | Question-aware context representation of a particular language |
| EM | Exact Match | $B_t$ | Context–context representation based on self-matching of a particular language |

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

### *3.1 EHMQuAD dataset*

As a classical probe to assess a machine's ability to understand natural language, QA is an essential and challenging task for Machine Reading Comprehension (MRC) (Hermann *et al.* 2015; Gupta and Khade 2020; Baradaran, Ghiasi, and Amirkhani 2022). QA has made many advances in recent years, primarily by fine-tuning the deep pre-trained architectures and creating large-scale datasets (Pires, Schlinger, and Garrette 2019; Conneau and Lample 2019; Conneau *et al.* 2020). However, regarding MQA, the scarcity of annotated corpora for languages other than English poses a significant challenge. Hence, there is a need to develop an MQA dataset for low-resource language. There are two main approaches for creating an MQA dataset: Synthetic corpus generation and Cross-lingual learning methods. Synthetic corpus generation involves leveraging machine translation techniques to generate language-specific QA datasets (Carrino *et al.* 2020), while the cross-lingual learning method uses few-shot and zero-shot (Lee and Lee 2019; Zhou *et al.* 2021) strategies to transfer knowledge between two languages with minimal training samples.

EHMQuAD is a large-scale multilingual QA dataset that was developed using a modified version of the Translate-Align-Retrieve (TAR) method (Carrino *et al.* 2020), which falls under the category of synthetic corpora generation techniques. The TAR method involves translating sentences from the widely used English SQuAD v1.1 (Rajpurkar *et al.* 2016) dataset into Hindi and Marathi languages. The translated sentences are aligned through a supervised alignment approach (Shi *et al.* 2021). The English SQuAD v1.1 dataset is the primary source for generating the multilingual EHMQuAD dataset. SQuAD v1.1 is a well-known large-scale MRC dataset that consists of 87,599 question-answer pairs along with their corresponding context. Each sample in the SQuAD dataset is represented as a tuple $(q_{en}, c_{en}, a_{en})$, where $q_{en}$ represents the question, $c_{en}$ represents the context, and $a_{en}$ represents the answer. Additionally, the dataset includes $a^{begin}$, which indicates the starting index of the answer within the context.

The objective is to utilize the SQuAD dataset as an input and subsequently apply the TAR method to obtain corresponding samples in the target languages, specifically $(q_{hi}, c_{hi}, a_{hi})$ for Hindi and $(q_{mr}, c_{mr}, a_{mr})$ for Marathi, along with their respective answer's start and end positions $a^{begin}_{hi/mr}$ and $a^{end}_{hi/mr}$, respectively, for both the languages. The TAR method encompasses two distinct sub-tasks discussed in the following sections, and a comprehensive understanding of the TAR method is presented in Algorithm 1.

(1) NMT training and translation using Sentence alignment: In this step, a Neural Machine Translation (NMT) model is trained to translate sentences from the source language $(q_{en}, c_{en}, a_{en})$ to the target language $(q_{hi}, c_{hi}, a_{hi})$ and $(q_{mr}, c_{mr}, a_{mr})$ using a sentence alignment mechanism.

(2) Retrieval of correct answer: This strategy filters out target samples where the translated answer does not lie in the translated context.

#### *3.1.1 NMT training and translation using sentence alignment*

There are two main methods to translate the dataset: Statistical Machine Translation (SMT) and NMT. NMT is preferred due to its lower memory requirements and improved translation accuracy. NMT uses an attention mechanism that considers sentences' semantic and syntactic contributions, making it more effective in maximizing translations. However, ensuring accurate sentence alignment between the source and target languages is challenging. To address this, a mechanism called sentence alignment (Shi *et al.* 2021) is used, which aligns the sentences during the decoding phase using a beam search algorithm. This approach helps mitigate translation inaccuracies and ensures better overall translation quality. By employing NMT and sentence alignment, the dataset can be effectively translated while maintaining the integrity and accuracy of the data.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Algorithm 1.** Pseudo Code for modified TAR methodology.

**Input:** SQuAD dataset $(q_{en}, c_{en}, a_{en})$
**Output:** EHMQuAD dataset $(q_{hi/mr}, c_{hi/mr}, a_{hi/mr})$

```
for c_en in context do
    q_hi/mr ← get question translation of q_en;
    c_hi/mr ← get context translation of c_en;
    TA_en−hi/mr ← aligned translated dataset;
    for any q question do
        for a_en in answers do
            a_hi/mr ← get translated answer of a_en;
            if (a_hi/mr in c_hi/mr) then
                return (q_hi/mr, c_hi/mr, a_hi/mr)
            else
                a[ ] ← search sense of a_hi/mr in WordNet;
                for word in a[] do
                    if (word in c_hi/mr) then
                        a_hi/mr ← word of (c_hi/mr);
                        return (q_hi/mr, c_hi/mr, a_hi/mr)
                    else
                        Remove the sample (q_hi/mr, c_hi/mr, a_hi/mr)
```

A large parallel corpus of en-hi and en-mr parallel text is used to train the NMT model for English-Hindi (en-hi) and English-Marathi (en-mr) translation. For this purpose, the Samanantar dataset (Ramesh *et al.* 2022) is utilized, which is a significant publicly available parallel corpus of English-Indic, Indic-English, and Indic-Indic language pair sentences. This corpus contains 10.126M parallel sentences for en-hi and 3.627M parallel sentences for en-mr languages. From this corpus, 10.12 million en-hi parallel sentences and 3.621 million en-mr parallel sentences are used for training the NMT model. Additionally, 5K parallel sentences are used for validation, and 1K parallel sentences are used for testing. To ensure a balanced dataset, simple random sampling is employed. The NMT model is then trained using this dataset, enabling it to learn the translation patterns and improve its performance.

NMT model is trained using three different NMT architectures: Stacked-LSTM (Wu *et al.* 2016; Johnson *et al.* 2017), Stacked-Bi-LSTM (Hasan *et al.* 2019), and Transformer (Vaswani *et al.* 2017). These architectures are selected to explore different approaches for NMT. The training and implementation of these models are carried out using the OpenNMT-py 2.0 toolkit (Klein *et al.* 2017), which provides a reliable framework for building and training NMT models. In the experimental setting, each architecture is configured with specific hyperparameters, such as the number of layers, hidden units, and attention mechanisms, to optimize their performance. The training process involves feeding the parallel training data from the Samanantar dataset into the models and iteratively updating the model parameters using optimization techniques like stochastic gradient descent. During training, the models learn the input–output pairs' statistical patterns and linguistic features, enabling them to generate accurate translations. The additional experimental setting for all these models is given below:

(1) *Stacked-LSTM model:* The Stacked-LSTM model (see Figure 1(a)) is composed of an encoder and decoder, each consisting of three stacked LSTM layers. During the training

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

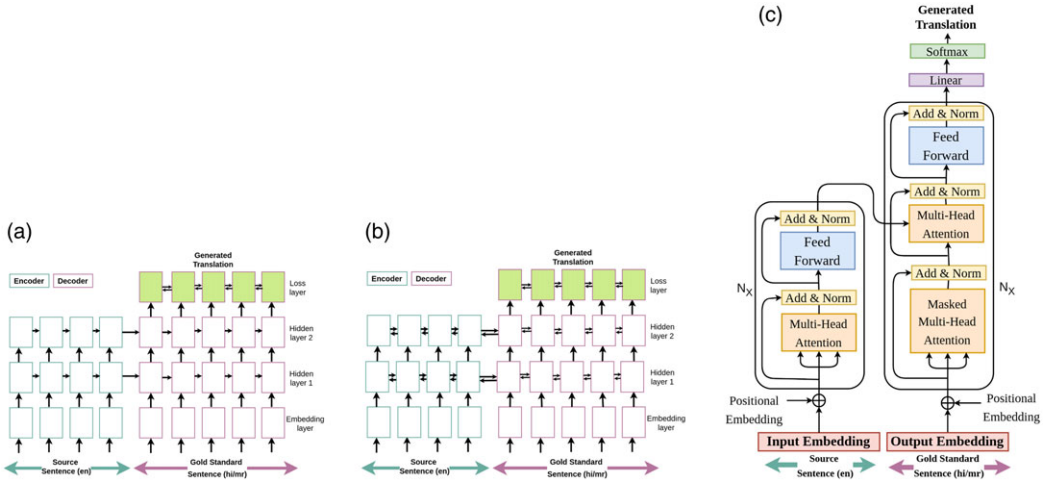


**Figure 1.** NMT Architecture (a) LSTM based model, (b) bi-LSTM based model, (c) transformer based model (Vaswani *et al*. 2017).

process, a pretrained word vector, specifically fastText (Grave *et al.* 2018), is used as input to the model. This choice of word vector assists in addressing the Out-Of-Vocabulary (OOV) issue that may arise. The model dimensions for the LSTM, Bi-LSTM, and attention models are 512. The Adam optimizer is employed, with a smoothing label value of 0.1 and gradient clipping set to 1.0. The learning rate is configured to 0.005, while the dropout rate is set to 0.3 for Hindi and 0.1 for Marathi, respectively.

(2) *Stacked-Bi-LSTM model:* The Stacked-Bi-LSTM model (see Figure 1(b)) follows the same experimental setup as the Stacked-LSTM model. Still, it uses bidirectional LSTM instead of unidirectional LSTM. This modification allows the model to consider past and future contexts when making predictions.

(3) *Transformer model:* This model (see Figure 1(c)) follows the experimental setup of (Vaswani *et al*. 2017). It incorporates 16 attention heads, with input and output vectors of dimension 1024. The feed-forward network in the inner layer has a dimension of 4096. For the Hindi language, a dropout rate of 0.3 is applied, while for the Marathi language, a dropout rate of 0.1 is utilized. The hyperparameter settings include an Adam optimizer with a smoothing label value of 0.1 to improve optimization stability. The learning rate is set to 0.003, ensuring an appropriate adjustment rate during training.

The effectiveness of translation is measured using the BLEU metric (Papineni *et al.* 2002), which quantifies the similarity between the translated output and reference translations. BLEU is calculated using equation (1). This metric provides a quantitative measure of translation quality.

$$BLEU = \exp\left(\sum_{n=1}^{ng} w_n log pr_n\right) \cdot BP \tag{1}$$

where $pr_n$ is n-gram precision, and BP is Brevity Penalty, refer equation (2)

$$BP = \begin{cases} 1, & \text{if } len(\hat{y}) > len(y) \\ \exp\left(1 - \frac{len(y)}{len(\hat{y})}\right), & \text{if } len(\hat{y}) \leq len(y) \end{cases} \tag{2}$$

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 2.** BLEU score of en-hi and en-mr translation

| Model | English-Hindi (en-hi) | English-Marathi (en-mr) |
|---|---|---|
| Stacked-LSTM | 47.13 | 37.56 |
| Stacked-BiLSTM | 49.42 | 39.89 |
| Transformer | **51.74** | **41.07** |

where $\hat{y}$ represents the predicted translated sentence while $y$ is the gold standard alignment sentence. Table 2 reports the accomplished BLEU score for all three models. The transformer model outperforms the other models with a BLEU score of 51.74 for en-hi translation and 41.07 for en-mr translation on the test set. This score is considered a good BLEU score and can be used for en-hi and en-mr translation. The transformer model effectively produces accurate and fluent translations for the en-hi and en-mr language pairs.

The trained NMT model is utilized to translate the entire SQUAD dataset, with each tuple being translated into Hindi and Marathi languages. During this translation process, it is observed that for certain tuples, the generated translated answer does not appear in the translated context. This occurrence is particularly noticeable for verbs, abbreviations (such as US, U.S., USA, America for the United States of America), numbers (specifically for the Marathi language), and other similar cases. This situation arises because the NMT model translates the same word differently based on the context. For example, if the translated context contains information about the "Bank of River," indicating that the context is about the river, but the translated answer outputs "Financial Institution Bank," which is unrelated to the context, then the answer is considered inaccurate. In such cases, the tuple consisting of the translated question, context, and answer will not be considered for training the model. To address this problem, retrieving the correct answer is necessary even when its translation is not explicitly present in the context. The process of retrieving the correct answer is discussed in Section 3.1.2.

#### *3.1.2 Retrieval of correct answer*

The retrieval of accurate answers is performed to obtain the complete translated version of the SQuAD dataset in both languages for cases where the translated answer ($a_{hi}$ and $a_{mr}$) is not present in the translated context. To ensure the retrieval of correct answers, the best strategy involves utilizing Hindi and Marathi WordNet (Bhattacharyya 2017), which serves as a thesaurus available for various languages. WordNet is used to acquire different synsets (i.e., sets of synonymous words) for a given word (lemma). It aids in replacing the existing translated answer ($a_{hi/mr}$) with an appropriate alternative found within the translated context ($c_{hi/mr}$).

In this strategy, the word in the translated answer ($a_{hi/mr}$) is searched within WordNet to obtain its various senses. The process provides a pool of searched senses and their matches for a word in the translated context ($c_{hi/mr}$). The translated answer is replaced with the corresponding word from the translated context if a match is found. This ensures that the retrieved answer accurately reflects the information present within the context. By applying this approach, the translated NMT model can effectively retrieve the correct answers, enhancing the overall accuracy and reliability of the translated SQuAD dataset.

The two subtasks described above have resulted in the creation of two versions of the EHMQuAD dataset. The first version consists of translating the original SQuAD dataset with sentence alignment. This version includes 48,769 translated samples in Hindi and 42,192 translated samples in Marathi. On the other hand, the second version incorporates the translation of the dataset with sentence alignment, along with retrieving the correct answer (Section 3.1.2). This version comprises 63,298 translated samples in Hindi and 51,359 translated samples in Marathi. The

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 3.** Statistics of EHMQuAD dataset over SQuADv1.1

| | EHMQuAD (first version) | | EHMQuAD (second version) | |
|---|---|---|---|---|
| | Hindi | Marathi | Hindi | Marathi |
| # of translated sample | 48,769 | 42,192 | 63,298 | 51,359 |
| Average *q* length | 142 | 137 | 144 | 138 |
| Average *c* length | 13 | 12 | 14 | 12 |
| Average *a* length | 3 | 2 | 3 | 2 |


"question_english": "How many universities and academies participated in the 2013 trans-genome study?",
"answer_english": "13 universities and academies",
"context_english": "A 2013 trans-genome study carried out by 30 geneticists, from 13 universities and academies, from 9 countries, assembling the largest data set available to date, for assessment of Ashkenazi Jewish genetic origins found no evidence of Khazar origin among Ashkenazi Jews. 'Thus, analysis of Ashkenazi Jews together with a large sample from the region of the Khazar Khaganate corroborates the earlier results that Ashkenazi Jews derive their ancestry primarily from populations of the Middle East and Europe, that they possess considerable shared ancestry with other Jewish populations, and that there is no indication of a significant genetic contribution either from within or from north of the Caucasus region', the authors concluded.",
"answer_english_start": 62,
"answer_english_end": 90,
"question_hindi": "2013 के ट्रांस-जीनोम अध्ययन में कितने विश्वविद्यालयों और अकादमियों ने भाग लिया?",
"answer_hindi": "13 विश्वविद्यालय और अकादमियां",
"context_hindi": "अशकेनाज़ी यहूदी आनुवंशिक उत्पत्ति के आकलन के लिए, 9 देशों के 13 विश्वविद्यालय और अकादमियां के 30 आनुवंशिकीविदों द्वारा 2013 में किए गए एक ट्रांस-जीनोम अध्ययन, आज तक उपलब्ध सबसे बड़े डेटा सेट को इकट्ठा करते हुए, अशकेनाज़ी यहूदियों के बीच खज़र मूल का कोई सबूत नहीं मिला। इस प्रकार, खजर खगनेट के क्षेत्र से एक बड़े नमूने के साथ एशकेनाज़ी यहूदियों का विश्लेषण पहले के परिणामों की पुष्टि करता है कि एशकेनाज़ी यहूदी मुख्य रूप से मध्य पूर्व और यूरोप की आबादी से अपने पूर्वजों को प्राप्त करते हैं, कि उनके पास अन्य यहूदी आबादी के साथ काफी साझा वंश है, और यह कि काकेशस क्षेत्र के भीतर या उत्तर से महत्वपूर्ण अनुवांशिक योगदान का कोई संकेत नहीं है, लेखकों ने निष्कर्ष निकाला।",
"answer_hindi_start": 61,
"answer_hindi_end": 90,
"question_marathi": "2013 च्या ट्रान्स-जीनोम अभ्यासात किती विद्यापीठे आणि अकादमींनी भाग घेतला?",
"answer_marathi": "13 विद्यापीठे आणि अकादमी",
"context_marathi": "अश्केनाझी ज्यू जनुकीय उत्पत्तीचे मूल्यांकन करण्यासाठी 9 देशांतील 13 विद्यापीठे आणि अकादमी मधील 30 जनुकशास्त्रज्ञांनी केलेल्या 2013 च्या ट्रान्स-जीनोम अभ्यासात, आजपर्यंतचा सर्वात मोठा डेटा संच एकत्रित करून, अश्केनाझी ज्यूंमध्ये खझार उत्पत्तीचे कोणतेही पुरावे आढळले नाहीत. अशाप्रकारे, खझर खगनाटेच्या प्रदेशातील मोठ्या नमुन्यासह अश्केनाझी ज्यूंचे विश्लेषण, पूर्वीच्या निकालांना पुष्टी देते की अश्केनाझी यहूदी त्यांचे वंशज मुख्यतः मध्य पूर्व आणि युरोपमधील लोकसंख्येमधून प्राप्त झाले होते, की त्यांच्याकडे इतर ज्यू लोकसंख्येसह लक्षणीय सामायिक वंशज आहेत. आणि काकेशस प्रदेशाच्या आतून किंवा उत्तरेकडून लक्षणीय अनुवांशिक योगदानाचे कोणतेही संकेत नाहीत, लेखकांनी निष्कर्ष काढला.",
"answer_marathi_start": 65,
"answer_marathi_end": 89


**Figure 2.** An example of the EHMQuAD dataset.

second version demonstrates an improvement in the number of tuples available in the EHMQuAD dataset. Table 3 presents the statistical details of both versions of the dataset, which were derived from the original 87,599 samples of the SQuADv1.1 dataset.

The proposed work utilizes the second version of the EHMQuAD dataset to develop an MQA framework, leveraging the enhanced quality and accuracy achieved through answer retrieval. Figure 2 provides a visual representation of a sample from the EHMQuAD dataset, showcasing the structure and format of the translated tuples. In the second version of the dataset, the start position ($a^{begin}hi/mr$) and end position ($a^{end}hi/mr$) of the translated answer are determined

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

by searching for the answer span within the translated context. These positions provide the necessary information to accurately locate the corresponding answer within the translated context. A comprehensive and enriched multilingual question-answering dataset is constructed through these steps, facilitating the development and evaluation of MQA models for Hindi and Marathi languages.

### 3.2 Error analysis

A detailed error analysis is done on the second version of the EHMQuAD dataset to validate the quality of translated data. There were some translated errors in the first version; hence, the retrieval of the correct answer approach (Section 3.1.2) is applied to generate the second version. But still, some errors are identified in the samples, and hence, there is a need for a qualitative analysis wherein the produced error and the reason behind these errors are detected and rectified accordingly. To get a better understanding of the answers generated from the context, we concluded a set of observations:

- Type I error: Sometimes, the translator cannot identify "Noun" in the context and wrongly translate it into another word. This is due to the part-of-speech (POS) tagging error proximity.
- Type II error: Some words in the dataset are represented in the form of orthographic variants, i.e., similar words having different writing styles due to language, regional, and personal preferences.
- There are some translation errors (Type III error and Type IV error) that occur during the translation process which affects the quality of the dataset. The Type III error is due to the year's translation into an unstructured number, while the Type IV error is due to translating the numbers into Devanagari script instead of Latin script, which confuses the model to identify the numerical data from the translated context.

The detailed description of these errors and the example are presented in Figure 3. Although these errors are responsible for downgrading the dataset's quality, we have corrected these errors, and encouragingly, there were only a few such contexts.

### 3.3 Proposed MQA framework

The proposed MQA framework (EHMMQA) is based on DNN architecture (see Figure 4). It is designed to handle MQA tasks in English, Hindi, and Marathi. The model is trained using $< q, c, a >$ samples from the EHMQuAD dataset, allowing it to acquire knowledge for each language. The model is evaluated on various datasets, including MMQA, Translated SQuAD, XQuAD, MLQA, and the proposed EHMQuAD datasets. The EHMMQA model accepts multilingual input through the *Question and Context Encoding layer*. The *Shared encoding layer* enables the model to process multilingual input by creating separate shared representations for the question and context. The *Question-Context Attention layer* performs question-context matching using attention-based RNN, generating a question-aware representation of the context. However, RNNs have limitations in capturing extensive context information, particularly when candidate answers are scattered. To address this, the *Context-Context Attention layer* dynamically refines the context representation by incorporating aggregated question information. This refinement process enhances the network's ability to extract the exact answer. Finally, the *Answer Extraction layer* employs a pointer network to determine the start ($a^{begin}$) and end ($a^{end}$) positions of the answers in all three languages. Each language has its pointer-based network. Details of each layer are elaborated in subsequent sections.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

| | |
|---|---|
| "question_english": "What is the name of the university's core curriculum?",<br>"answer_english": "the Common Core",<br>"context_english": "Undergraduate students are required to take a distribution of courses to satisfy the university's core curriculum known as the **Common Core**. In 2012-2013, the Core classes at Chicago were limited to 17 students, and are generally led by a full-time professor (as opposed to a teaching assistant). As of the 2013–2014 school year, 15 courses and demonstrated proficiency in a foreign language are required under the Core. Undergraduate courses at the University of Chicago are known for their demanding standards, heavy workload and academic difficulty; according to Uni in the **USA**, Among the academic cream of American universities – Harvard, Yale, Princeton, MIT, and the University of Chicago – it is UChicago that can most convincingly claim to provide the most rigorous, intense learning experience.",<br><br>"question_hindi": "विश्वविद्यालय के मुख्य पाठ्यक्रम का नाम क्या है?",<br>"answer_hindi": "आम कोर",<br>"context_hindi": "**आम कोर** के रूप में ज्ञात विश्वविद्यालय के मुख्य पाठ्यक्रम को संतुष्ट करने के लिए स्नातक छात्रों को पाठ्यक्रमों का वितरण करना आवश्यक है। 2012-2013 में, शिकागो में कोर कक्षाएं 17 छात्रों तक सीमित थीं, और आम तौर पर एक पूर्णकालिक प्रोफेसर (एक शिक्षण सहायक के विपरीत) के नेतृत्व में होती हैं। 2013-2014 के स्कूल वर्ष के अनुसार, कोर के तहत 15 पाठ्यक्रम और एक विदेशी भाषा में दक्षता का प्रदर्शन आवश्यक है। शिकागो विश्वविद्यालय में स्नातक पाठ्यक्रम उनके मांग मानकों, भारी कार्यभार और अकादमिक कठिनाई के लिए जाने जाते हैं; **संयुक्त राज्य अमेरिका** में यूनी के अनुसार, अमेरिकी विश्वविद्यालयों की अकादमिक क्रीम में - हार्वर्ड, येल, प्रिंसटन, एमआईटी, और शिकागो विश्वविद्यालय - यह यूशिकागो है जो सबसे कठोर, गहन सीखने का अनुभव प्रदान करने का दावा कर सकता है।",<br><br>"question_marathi": "विद्यापीठाच्या मुख्य अभ्यासक्रमाचे नाव काय आहे?",<br>"answer_marathi": "सामान्य कोर",<br>"context_marathi": "अंडरग्रेजुएट विद्यार्थ्यांना **सामान्य कोर** म्हणून ओळखल्या जाणाऱ्या विद्यापीठाच्या मुख्य अभ्यासक्रमाचे समाधान करण्यासाठी अभ्यासक्रमांचे वितरण करणे आवश्यक आहे. 2012-2013 मध्ये, शिकागो येथील मुख्य वर्ग 17 विद्यार्थ्यांपुरते मर्यादित होते, आणि सामान्यत: पूर्ण-वेळ प्राध्यापक (शिक्षण सहाय्यकाच्या विरुद्ध) नेतृत्व करतात. 2013-2014 शालेय वर्षानुसार, कोअर अंतर्गत 15 कोर्सेस आणि परदेशी भाषेतील प्राविण्य दाखवणे आवश्यक आहे. शिकागो विद्यापीठातील अंडरग्रेजुएट अभ्यासक्रम त्यांच्या मागणीच्या दर्जासाठी, कामाचा प्रचंड ताण आणि शैक्षणिक अडचण यासाठी ओळखले जातात; **यूएसए** मधील युनिच्या मते, हार्वर्ड, येल, प्रिन्सटन, एमआयटी आणि शिकागो विद्यापीठ - अमेरिकन विद्यापीठांच्या शैक्षणिक क्रिममध्ये - हे UChicago आहे जे सर्वात कठोर, तीव्र शिक्षण अनुभव प्रदान करण्याचा दावा करू शकते." | **Type I error:** During the translation, sometimes the translator identify the noun but convert it into actual meaning of the word instead of keeping the same word. In the given example **"Common Core"** is a name of University but during translation in both Hindi and Marathi it should be **"कॉमन कोअर "** instead of **"आम कोर"** in Hindi and **"सामान्य कोर"** in Marathi.<br><br>**Type II error:** Some words are represented in the dataset in the form of orthographic variants, i.e., similar words having different writing styles due to the language, regional and personal preferences. In the given example, the orthographic variant of the word **"USA"** in Hindi and Marathi language is written as "संयुक्त राज्य अमेरिका" and **"यूएसए"** respectively. |
| "question_english": "What year was Casimir Pulaski born in Warsaw?",<br>"answer_english": **"1745"**,<br>"context_english": "One of the most famous people born in Warsaw was Maria Skłodowska-Curie, who achieved international recognition for her research on radioactivity and was the first female recipient of the Nobel Prize. Famous musicians include Władysław Szpilman and Frédéric Chopin. Though Chopin was born in the village of Żelazowa Wola, about 60 km (37 mi) from Warsaw, he moved to the city with his family when he was seven months old. Casimir Pulaski, a Polish general and hero of the American Revolutionary War, was born here in **1745**.",<br><br>"question_hindi": "कासिमिर पुलस्की का जन्म किस वर्ष वारसॉ में हुआ था?",<br>"answer_hindi": **"1745"**,<br>"context_hindi": "वारसॉ में पैदा हुए सबसे प्रसिद्ध लोगों में से एक मारिया स्कोलोडोव्स्का-क्यूरी थीं, जिन्होंने रेडियोधर्मिता पर अपने शोध के लिए अंतरराष्ट्रीय स्तर पर पहचान हासिल की और नोबेल पुरस्कार की पहली महिला प्राप्तकर्ता थीं। प्रसिद्ध संगीतकारों में व्लादिस्लॉ स्ज़िपिलमैन और फ्रेडरिक चोपिन शामिल हैं। हालांकि चोपिन का जन्म ज़ेलाज़ोवा वोला गाँव में हुआ था, जो वारसॉ से लगभग 60 किमी (37 मील) दूर था, लेकिन जब वह सात महीने का था, तब वह अपने परिवार के साथ शहर चला गया। पोलिश जनरल और अमेरिकी क्रांतिकारी युद्ध के नायक कासिमिर पुलस्की का जन्म यहां **1 74 5** में हुआ था।",<br><br>"question_marathi": "Casimir Pulaski चा जन्म वॉर्सा येथे कोणत्या वर्षी झाला?",<br>"answer_marathi": **"१७४५"**,<br>"context_marathi": "वॉर्सा येथे जन्मलेल्या सर्वात प्रसिद्ध लोकांपैकी एक मारिया स्कोडोव्स्का-क्युरी होती, जिने किरणोत्सर्गीतेवरील संशोधनासाठी आंतरराष्ट्रीय मान्यता मिळवली आणि नोबेल पारितोषिक मिळविणारी पहिली महिला होती. प्रसिद्ध संगीतकारांमध्ये Władysław Szpilman आणि Frédéric Chopin यांचा समावेश आहे. चोपिनचा जन्म वॉर्सा पासून सुमारे **६०** किमी (**३७** मैल) झोलाझोवा वोला गावात झाला असला तरी तो सात महिन्यांचा असताना तो आपल्या कुटुंबासह शहरात गेला. कॅसिमिर पुलस्की, एक पोलिश सेनापती आणि अमेरिकन क्रांतिकारी युद्धाचा नायक, येथे **1745** मध्ये जन्म झाला." | **Type III error:** In few samples, space is inserted between the numbers which converts the year into an unstructured number and confuses the model to get the correct span of answer (year) from the translated context. For example, the year **"1745"** is translated as **"1 74 5"**.<br><br>**Type IV error:** In the Marathi snippet, sometimes the numbers are converted into Devanagri script and sometimes it remains in latin script. For example in the translated context, **"60" is converted into "६०" while "1745" remains "1745".** |

**Figure 3.** Examples showing type of errors.

### *3.3.1 Question & Context encoding layer*

Let us consider the English question $q_{en} = \left\{qw_1^{q_{en}}, qw_2^{q_{en}}, \ldots, qw_{x_{en}}^{q_{en}}\right\}$ and context $c_{en} = \left\{cw_1^{c_{en}}, cw_2^{c_{en}}, \ldots, cw_{y_{en}}^{c_{en}}\right\}$, Hindi question $q_{hi} = \left\{qw_t^{q_{hi}}\right\}_{t=1}^{x_{hi}}$ and context $c_{hi} = \left\{cw_t^{c_{hi}}\right\}_{t=1}^{y_{hi}}$, and Marathi question $q_{mr} = \left\{qw_t^{q_{mr}}\right\}_{t=1}^{x_{mr}}$ and context $c_{mr} = \left\{cw_t^{c_{mr}}\right\}_{t=1}^{y_{mr}}$. This layer is responsible for obtaining the encoding of the questions and contexts. From the given question and context, word embedding $\left(\left\{e_t^{q_{en}}\right\}_{t=1}^{x_{en}}, \left\{e_t^{c_{en}}\right\}_{t=1}^{y_{en}}, \left\{e_t^{q_h}\right\}_{t=1}^{x_h}, \left\{e_t^{c_h}\right\}_{t=1}^{y_h}, \left\{e_t^{q_{mr}}\right\}_{t=1}^{x_{mr}}\right.$, and $\left.\left\{e_t^{c_{mr}}\right\}_{t=1}^{y_{mr}}\right)$ are generated from the pre-trained word embedding model. According to (Lee, Cho, and Hofmann 2017; Xia, Ding, and Liu 2020), character embedding is an extension of word embedding as it deals with OOV words. Hence, character-level embedding $\left(\left\{c_t^{q_{en}}\right\}_{t=1}^{x_{en}}, \left\{c_t^{c_{en}}\right\}_{t=1}^{y_{en}}, \left\{c_t^{q_{hi}}\right\}_{t=1}^{x_{hi}}, \left\{c_t^{c_{hi}}\right\}_{t=1}^{y_{hi}}, \left\{c_t^{q_{mr}}\right\}_{t=1}^{x_{mr}}\right.$, and

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

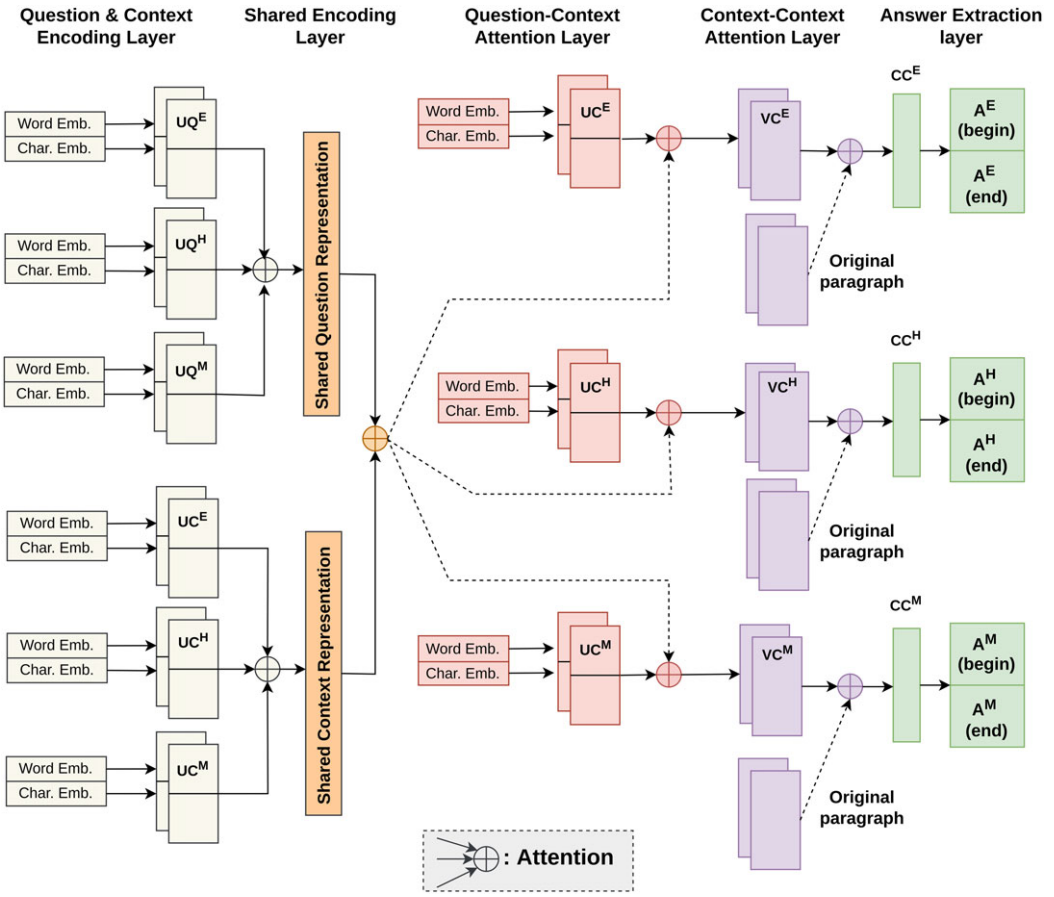


**Figure 4.** Proposed unified model of MQA.

$\left\{c_t^{c_{mr}}\right\}_{t=1}^{y_{mr}}$) is also employed in this work. These character-level embeddings are produced as the final output of a bi-directional gated recurrent unit (Bi-GRU), employed for the token's character embeddings. The word embeddings and character-level embeddings are concatenated to enhance the word representation.

$$E_i = concat\left(e_t^{q_k}, c_t^{q_k}\right)$$

For each question and context, the final word representations $u_t^{q_k}$ and $u_t^{c_k}$ are computed using Bi-GRU as follows:

$$\begin{aligned} u_t^{q_k} &= Bi\text{-}GRU\left(u_{t-1}^{q_k}, [E_i]\right) \\ u_t^{c_k} &= Bi\text{-}GRU\left(u_{t-1}^{c_k}, [E_i]\right) \end{aligned} \tag{3}$$

where $k \in \{en, hi, mr\}$ denotes the English(en), Hindi(hi), and Marathi(mr) languages. Bi-GRU is chosen over LSTM due to its computational efficiency.

### *3.3.2 Shared encoding layer*

The shared representation of the question and context is obtained separately for each language from the encoding of the previous layer. This shared representation improves the representation of question–question and context–context relationships. Since the dataset contains

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

the same information for every question and context, this shared representation is achieved through soft alignment between words. A single separate representation of question and context is formed by combining the representation of individual languages. $\left(R_t^{q_{en}},\ R_t^{q_{hi}} \text{and } R_t^{q_{mr}}\right)$ and $\left(R_t^{c_{en}},\ R_t^{c_{hi}} \text{and } R_t^{c_{mr}}\right)$ are the English, Hindi, and Marathi question and context representations, with the awareness of other two languages. The English question representation, taking into account the Hindi and Marathi question representations, is computed using a Bi-GRU as shown in equation (4):

$$R_t^{q_{en}} = Bi\text{-}GRU\left(R_{t-1}^{q_{en}}, v_t^{q_{hi}}, v_t^{q_{mr}}\right) \tag{4}$$

$$\begin{aligned} l_j^t &= N^T tanh\left(\left[W_u^{q_{hi}}, u_j^{q_{hi}}\right]\left[W_u^{q_{mr}}, u_j^{q_{mr}}\right]\left[W_u^{q_{en}}, u_t^{q_{en}}\right]\left[W_r^{q_{en}}, R_{t-1}^{q_{en}}\right]\right) \\ v_t^{q_{hi}} &= \sum_{i=1}^{x_{hi}}\left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{x_{hi}} exp\left(l_j^t\right)}\right)\cdot u_i^{q_{hi}} \\ v_t^{q_{mr}} &= \sum_{i=1}^{x_{mr}}\left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{x_{mr}} exp\left(l_j^t\right)}\right)\cdot u_i^{q_{mr}} \end{aligned} \tag{5}$$

where $N^T$ is a weight vector, $W_u^{q_{en}}, W_u^{q_{hi}}, W_u^{q_m}, W_r^{q_{en}}$ are the learnable weight matrices, and $< v_t^{q_{hi}}$ & $v_t^{q_{mr}} >$ are attention-based pooling vectors. The English question representation $\left(R_t^{q_{en}}\right)$ is computed using Bi-GRU by concatenating the pooling vector $v_t^{q_{hi}}$ & $v_t^{q_{mr}}$ with the previous (t-1) representation of $\left(R_t^{q_{en}}\right)$. Equation (5) computes the pooling vector $v_t^{q_{hi}}$ & $v_t^{q_{mr}}$ of English question representation $u_t^{q_{hi}}$ & $u_t^{q_{mr}}$ at time t. Likewise, the other two question representations, i.e., $R_t^{q_{hi}}$ and $R_t^{q_{mr}}$ are computed in the similar manner. In the end, the single shared question representation is computed by combining the individual question representations from all three languages as shown in equation (6):

$$\left\{R_t^q\right\}_{t=1}^{(x_{en}+x_{hi}+x_{mr})} = \left\{R_t^{q_{en}}\right\}_{t=1}^{x_{en}} \oplus \left\{R_t^{q_{hi}}\right\}_{t=1}^{x_{hi}} \oplus \left\{R_t^{q_{mr}}\right\}_{t=1}^{x_{mr}} \tag{6}$$

Similarly, the shared context representation is computed by concatenating the context representations from all three languages, as indicated in equation (7):

$$\left\{R_t^c\right\}_{t=1}^{(y_{en}+y_{hi}+y_{mr})} = \left\{R_t^{c_{en}}\right\}_{t=1}^{y_{en}} \oplus \left\{R_t^{c_{hi}}\right\}_{t=1}^{y_{hi}} \oplus \left\{R_t^{c_{mr}}\right\}_{t=1}^{y_{mr}} \tag{7}$$

In equations (6) and (7), $\oplus$ denotes the concatenation operation, resulting in a unified representation encompassing information from all three languages.

### *3.3.3 Question-context attention layer*

The question-aware-context representation is computed in this layer using an attention-based RNN mechanism. This approach captures the relevance and significance of context information given a specific question. The question-aware-context representation is computed separately for all three languages, considering the shared question representation and shared context representation. Since the shared question representation only includes a limited number of words, it provides a condensed representation. On the other hand, the shared context representation incorporates a larger number of words, as the context vocabulary is generally more extensive than that of the corresponding question. This facilitates a more comprehensive and enriched question-aware-context representation. The English question-aware-context representation is denoted by equation (8):

$$S_t^{c_{en}} = Bi\text{-}GRU\left(S_{t-1}^{c_{en}}, z_t^{c_{en}}\right) \tag{8}$$

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

$$l_j^t = N^T tanh\left(\left[W_r^q, R_j^q\right]\left[W_r^c, R_j^c\right]\left[W_u^{c_{en}}, u_t^{c_{en}}\right]\left[W_s^{c_{en}}, S_{t-1}^{q_{en}}\right]\right)$$

$$z_t^{c_{en}} = \sum_{i=1}^{x_{en}+x_{hi}+x_{mr}}\left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{x_{en}+x_{hi}+x_{mr}} exp\left(l_j^t\right)}\right)\cdot R_i^q + \sum_{i=1}^{x_{en}+x_{hi}+x_{mr}}\left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{x_{en}+x_{hi}+x_{mr}} exp\left(l_j^t\right)}\right)\cdot R_i^c \quad (9)$$

where $W_r^q$, $W_r^c$, $W_u^{c_{en}}$, $W_s^{c_{en}}$ are the learnable weight matrices and $z_t^{c_{en}}$ is an attention-based pooling vector (equation (9)). Likewise, the Hindi question-aware-context representation $\left(S_t^{c_{hi}}\right)$ and Marathi question-aware-context representation $\left(S_t^{c_{mr}}\right)$ are derived by applying the same principles as the English representation.

### *3.3.4 Context-context attention layer*

In the previous layer, the question-aware-context representation highlights the important features of the context, but the problem with this representation is that it handles very limited information. Hence, one potential answer fails to overlook the significant information in the context beyond its window. This approach is designed to address lexical or syntactical differences between the question–answer pairs in the SQuAD and EHMQuAD datasets. By dynamically matching the question-aware-context representation with the original context, the model can capture the relevant information from the context and encode it appropriately with the corresponding question, resulting in a more comprehensive and accurate context representation. For the English context, the final context representation is computed using the following equation (10):

$$B_t^{c_{en}} = Bi\text{-}GRU\left(B_{t-1}^{c_{en}}, S_t^{c_{en}}, z_t^{c_{en}}\right) \quad (10)$$

$$l_j^t = N^T tanh\left(\left[W_{b'}^{c_{en}}, W_{b''}^{c_{en}}\right]\left[S_j^{c_{en}}, S_t^{c_{en}}\right]^T\right)$$

$$z_t^{c_{en}} = \sum_{i=1}^{y_{en}}\left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{y_{en}} exp\left(l_j^t\right)}\right)\cdot S_i^{c_{en}} \quad (11)$$

where $W_{b'}^{c_{en}}$ and $W_{b''}^{c_{en}}$ are the learnable weight matrices. Similarly, the final context representation of Hindi $\left(\left\{B_t^{c_{hi}}\right\}_{t=1}^{y_{hi}}\right)$ and Marathi $\left(\left\{B_t^{c_{mr}}\right\}_{t=1}^{y_{mr}}\right)$ is computed in the same fashion by equations (10) and (11).

### *3.3.5 Answer layer*

A pointer network is employed to handle the multilingual nature of the MQA framework and generate answers in multiple languages. The pointer network is based on the seq-to-seq pointer network (Chen *et al.* 2021), which is responsible for extracting the start position $\left(a^{begin}\right)$ and end position $\left(a^{end}\right)$ of the answer from either the English, Hindi, or Marathi context. Given the context in any language as input, the answer layer's objective is to determine the appropriate start and end positions within the context that correspond to the answer. This is achieved by computing the following equation (12):

$$l_j^t = N^T tanh\left(\left[W_p^{c_{en}}, B_j^{c_{en}}\right]\left[W_h^{a_{en}}, h_{t-1}^{a_{en}}\right]\right)$$

$$a_i^t = \left(\frac{exp\left(l_i^t\right)}{\sum_{j=1}^{y_{en}} exp\left(l_j^t\right)}\right) \quad (12)$$

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 4.** Statistics of MQA dataset

| | MMQA | Translated SQuAD | XQuAD | MLQA | EHMQuAD |
|---|---|---|---|---|---|
| # of sample (English) | 5,495 | 18,454 | 1,190 | 12,738 | 63,298 |
| # of sample (Hindi) | 5,495 | 18,454 | 1,190 | 5,425 | 63,298 |
| # of sample (Marathi) | - | - | - | - | 51,359 |

where $h_{t-1}^{a_{en}}$ is the last hidden state of the answer layer. The attention pooling vector $a_i^t$ depends on the recently predicted probability $a^t$, which serves as an input to this layer. The hidden state of the answer layer is computed by equation (13):

$$h_t^{a_{en}} = Bi\text{-}GRU\left(h_{t-1}^{a_{en}}, z_t^{c_{en}}\right)$$
$$z_t^{c_{en}} = \sum_{i=1}^{y_{en}} \left(a_i^t B_i^{c_{en}}\right) \tag{13}$$

The answer for the English language is obtained from the answer network by predicting the start position at time step $t = 1$, and subsequently predicting the end position at the next time step. This prediction is based on the probabilities assigned to each position in English. The prediction of the answer is determined by selecting the maximum probability among the predicted positions.

$$a_{en}^t = argmax\left(a_1^t, a_2^t, \ldots, a_{y_{en}}^t\right) \tag{14}$$

For Hindi and Marathi languages, the answer's start and end positions are also predicted from their corresponding context using equation (14). This approach allows the answer network to identify the most likely start and end positions of the answer for each language, facilitating the generation of language-specific answers tailored to the corresponding context.

## 4. Experiments

### *4.1 Datasets*

The performance of the proposed model is evaluated using five different MQA datasets, which consist of multilingual question-answer pairs along with their corresponding contexts. These datasets provide valuable resources for assessing the model's ability to handle multiple languages. Among these datasets, MMQA, Translated SQuAD, XQuAD, and MLQA contain samples in the format of $< q, c, a, a^{begin} >$ for two languages, namely English and Hindi. On the other hand, the proposed EHMQuAD dataset is unique as it provides samples in all three languages. Table 4 shows the statistics of these datasets. This comprehensive evaluation using diverse, multilingual datasets thoroughly assesses the model's performance across different language pairs.

#### *4.1.1 MMQA*

MMQA[c] dataset (Gupta *et al.* 2018) is an MQA dataset for English and Hindi languages. This dataset has comparable corpora of 500 documents for both English and Hindi languages but not the parallel corpora along with the sample of $< q, c, a >$. Gupta *et al.* (2019) proposed a novel algorithm that uses the MMQA dataset of 5,495 QA pairs to evaluate and create a snippet for each

[c]http://www.iitp.ac.in/ai-nlp-ml/resources.html

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

pair. These selected QA pairs and their corresponding created snippets are used to evaluate the EHMMQA model. This dataset is converted into SQuAD-like sample instances.

#### *4.1.2 Translated SQuAD*

The Translated SQuAD[d] dataset (Gupta *et al.* 2019) is derived from the original SQuAD dataset and involves the translation (using Google Translate API) of 18,454 samples from English to the Hindi language. For each context available in the original SQuAD dataset, a sample consisting of the question ($q$), context ($c$), and answer ($a$) is selected and transformed into the format of $< q, c, a, a_{start}, a_{end} >$ for both English and Hindi languages. This ensures that the translated dataset covers all the contexts present in the SQuAD dataset while providing bilingual information, which undergoes a preprocessing step to address any discrepancies that may arise from the direct translation between languages. These discrepancies are likely due to the inherent syntax, grammar, and language structure differences between English and Hindi. The preprocessing step helps mitigate these issues, ensuring the dataset's quality and reliability for training and evaluation purposes.

#### *4.1.3 XQuAD*

DeepMind has developed a Cross-lingual Question Answering Dataset (XQuAD)[e] (Artetxe *et al.* 2020) for MQA tasks. XQuAD covers 12 languages, including Hindi, and is a reliable benchmark for evaluating MQA models. The dataset consists of 1,190 question–answer pairs for each of the 12 languages. These pairs are derived from 240 contexts originally found in the SQuAD dataset. XQuAD is particularly valuable for evaluating English-Hindi Machine Reading Comprehension (MRC) tasks since it has been translated by linguistic experts, making this dataset more reliable for evaluating MQA models. XQuAD is a monolingual variant of the SQuAD dataset allowing, it to be utilized for evaluating the monolingual settings of the EHMMQA model in monolingual settings, such as English-English ($Q_{en} - C_{en}$) and Hindi-Hindi ($Q_{hi} - C_{hi}$) scenarios.

#### *4.1.4 MLQA*

MLQA[f] (Lewis *et al.* 2020) is an MQA dataset containing QA pairs along with the context in seven languages, including Hindi. MLQA is a comprehensive and widely used benchmark for evaluating multilingual and cross-lingual QA models. This dataset contains 5,425 QA pairs and follows a structure similar to SQuAD, ensuring familiarity and compatibility with existing MQA models. The inclusion of Hindi in MLQA enables the specific evaluation of MQA models within the context of this language. By leveraging MLQA, researchers can conduct rigorous and reliable evaluations of multilingual and cross-lingual QA models, facilitating advancements in the field.

### *4.2 Experimental setup*

The EHMMQA framework is designed in two different experimental setups.

(1) ***Setup-I:*** In Setup-I, the training focuses on the language pair of English and Hindi, excluding the Marathi sub-module of the EHMMQA framework. The model is trained using triplets consisting of an English question $< q_{en} >$, English context $< c_{en} >$, and English answer $< a_{en} >$, as well as Hindi counterparts $< q_{hi}, c_{hi}, a_{hi} >$. The entire EHMQuAD dataset is used for training and validation, while evaluation is performed on multiple

[d] http://www.iitp.ac.in/ai-nlp-ml/resources.html
[e] https://github.com/deepmind/xquad
[f] https://github.com/facebookresearch/MLQA

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 5.** Details of monolingual and multilingual settings for experimental setup-I

| Settings | Input | Expected output | Description |
|---|---|---|---|
| Monolingual | $Q_{en} - C_{en}$ | $A_{en}$ | The question and context are in the same language (English or Hindi). |
| | $Q_{hi} - C_{hi}$ | $A_{hi}$ | |
| Multilingual | $Q_{en} - C_{(en+hi)}$ | $A_{en}$ and $A_{hi}$ | The question is in either English or Hindi, but the context is in both. |
| | $Q_{hi} - C_{(en+hi)}$ | | |
| | $Q_{(en+hi)} - C_{(en+hi)}$ | $A_{en}$ and $A_{hi}$ | The question and context are in English and Hindi. |

**Table 6.** Details of monolingual and multilingual settings for experimental setup-II

| Settings | Input | Expected output | Description |
|---|---|---|---|
| Monolingual | $Q_{en} - C_{en}$ | $A_{en}$ | The question and context are in the same language (English, Hindi, or Marathi). |
| | $Q_{hi} - C_{hi}$ | $A_{hi}$ | |
| | $Q_{mr} - C_{mr}$ | $A_{mr}$ | |
| | $Q_{en} - C_{(en+hi+mr)}$ | $A_{en}$, $A_{hi}$ and $A_{mr}$ | The question is in either English, Hindi, or Marathi, but the context is in all languages. |
| | $Q_{hi} - C_{(en+hi+mr)}$ | | |
| | $Q_{mr} - C_{(en+hi+mr)}$ | | |
| Multilingual | $Q_{(en+hi)} - C_{(en+hi+mr)}$ | $A_{en}$, $A_{hi}$ and $A_{mr}$ | The question is in either of the two languages, while context is in all three languages (English, Hindi, and Marathi). |
| | $Q_{(en+mr)} - C_{(en+hi+mr)}$ | | |
| | $Q_{(hi+mr)} - C_{(en+hi+mr)}$ | | |
| | $Q_{(en+hi+mr)} - C_{(en+hi+mr)}$ | $A_{en}$, $A_{hi}$ and $A_{mr}$ | The question and context are in English, Hindi, and Marathi. |

datasets, including MMQA, Translated SQuAD, XQuAD, MLQA, and EHMQuAD itself. For the experiment, the EHMQuAD dataset is split into training (consisting of 54,298 QA pairs), validation (consisting of 3,000 QA pairs), and test (consisting of 6,000 pairs) sets.

(2) ***Setup-II:*** In Setup-II, the EHMMQA model is trained using triplets of all three languages, including English ($< q_{en}, c_{en}, a_{en} >$), Hindi ($< q_{hi}, c_{hi}, a_{hi} >$), and Marathi ($< q_{mr}, c_{mr}, a_{mr} >$). The model is trained, validated, and tested on the proposed EHMQuAD dataset. The total number of common samples in all three languages is 46,489. Therefore, the EHMQuAD dataset is split into training (37,489 QA pairs), validation (3,000 QA pairs), and test (6,000 pairs) sets for experimental purposes.

The EHMMQA model is trained on the EHMQuAD dataset to reduce the summation of the negative log probabilities of the gold answer's start and end positions for all three languages. For evaluating the proposed framework, the predicted answer should be in all three languages, regardless of the question and context language. To conduct the experiment, *IndicFT* (Kakwani *et al.* 2020) and *fastText* (Grave *et al.* 2018), a publicly available pretrained word embeddings of dimension 300 is used for Hindi and Marathi language and English language, respectively. Multilingual word embedding is done by aligning the monolingual word vectors of all these languages in a single vector space. Tables 5 and 6 show the various monolingual and multilingual settings for the proposed MQA framework for two different experimental setups, respectively.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 7.** Details of monolingual model

| Baseline model | Description |
|---|---|
| Monolingual model (English)<br>Monolingual model (Hindi)<br>Monolingual model (Marathi) | The baseline model is a simplified version of the proposed model, focusing on a single language. It utilizes language-specific questions andcontexts as input, excluding the shared question and shared context representations. In this model, the question and context embeddings for aparticular language are fed into the question-context attention layer, and the answer layer predicts the start and end positions of the answer.This approach enables the model to handle the task using only one language-specific input. |

### *4.3 Evaluation metrics*

The performance metrics used to evaluate the MQA model include Recall, Precision, F1-score, Accuracy, and Exact Match (EM). Accuracy measures the proportion of correctly identified answers from the total number of answers in the dataset. In the context of MQA, which is based on MRC, where the answer is extractive, accuracy is equivalent to EM (Dzendzik, Foster, and Vogel 2021). EM is calculated by comparing the extracted answer with the pool of ground truth answers for a given QA pair. If the extracted answer matches any ground truth answers, EM is set to 1; otherwise, it is set to 0. Since MQA is an extractive task, the model may provide ambiguous, partial, or redundant responses. Hence, Precision, Recall, and F1-score are used to evaluate the model's performance, even if the extracted answer does not exactly match the ground truth answer. These metrics consider the closeness between the extracted and the ground truth answers, even if they are not an exact match. EM measures the exact match between the extracted answer and the gold answer, while the F1-score quantifies the closeness between the extracted answer and the gold answer. In the monolingual settings ($Q_{en} - C_{en}$, $Q_{hi} - C_{hi}$ and $Q_{mr} - C_{mr}$), the predicted answer is considered correct if the model extracts the accurate answer in the same language as the question. In the case of multilingual settings, the predicted answer is accepted only if the model extracts the accurate answer in all three languages or in any two languages depending on multiple settings.

## 5. Results and discussion

The performance of an EHMMQA model is evaluated and compared with various baseline models, including an IR-based model (Gupta *et al.* 2018), monolingual (English, Hindi, and Marathi) models (refer Table 7 for more details), deep canonical correlation analysis (Deep CCA) model (Andrew *et al.* 2013) and deep neural network (DNN) model (Gupta *et al.* 2019). The details of these models are discussed as follows:

- IR Model: This translation-based model consists of four steps: Document Processing, Question Processing, Candidate Answer Extraction, and Candidate Answer Scoring. The context is processed in the Document Processing step to generate snippets for each question. The Question Processing step involves Question Classification (QC) and query formulation. QC categorizes the question into different classes, while query formulation identifies nouns, verbs, and adjectives in the question to create a query. The output of QC plays a crucial role in extracting the candidate answer, where the passage is tagged with a named entity tagger to generate a list of candidate answers. Each candidate answer is assigned a score in the Candidate Answer Scoring step based on the relevance of its corresponding sentence.
- Deep CCA Model: Deep CCA is a multivariate statistical technique used for learning and extracting highly correlated representations between two sets of variables or views.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

It extends the traditional CCA method by incorporating DNN to capture nonlinear and complex relationships between the views. Deep CCA optimizes the model's parameters to ensure maximum correlation between the final representations of the two views. This is typically done by minimizing a loss function to quantify the disparity between the original views' correlations and their mapped representations' correlations.

- DNN Model: In this model, a shared representation of the question is learned using soft alignment between the words from both languages. This shared question representation and the original context are received as input to the answer extraction layer to generate the answer span. This answer extraction process considers the question and context's relevance. In addition, the pointer network is used to get the answer span's start and end position from the context.

In experimental setup-I, the MQA model's monolingual and multilingual settings results on all datasets are presented in Tables 8 and 9, respectively. For experimental setup-II, the results of the MQA model for monolingual and multilingual settings on the proposed EHMQuAD dataset are reported in Tables 10 and 11, respectively. It is evident from the results that the EHMMQA framework outperforms all baseline models, demonstrating its superiority in handling the MQA task.

### *5.1 Result analysis*

The results of the experiments are analyzed to see the impact of shared context representation and various word representations. The overall observations have been made based on the impact of the shared context representation:

(1) Improved Performance: The inclusion of the shared context representation in the EHMMQA model has shown a notable enhancement in performance compared to the baseline models. This suggests that the shared context representation enables better cross-lingual understanding and information integration.

(2) Enhanced Cross-Lingual Capability: The shared context representation facilitates effective information exchange between different languages. This enables the model to leverage relevant information from multiple languages, improving accuracy and coverage in generating answers.

(3) Language-Agnostic Context Understanding: The shared context representation allows the model to capture language-agnostic aspects of the context, enabling a more comprehensive understanding of the information. This is particularly beneficial in multilingual settings where the context can be in different languages.

(4) Robustness to Language Variations: The shared context representation helps the model handle language variations within the context more effectively. This is crucial in scenarios where the context may contain linguistic differences or code-switching between multiple languages.

Overall, the shared context representation in the EHMMQA model has proven advantageous, leading to improved performance and robustness in MQA tasks. The results of the shared context representation for two experimental setups are discussed in the following section. The results of the EHMMQA model on all five datasets are presented in Tables 8, 9, 10 and 11.

- ***Monolingual and Multilingual Settings-Experimental Setup-I:*** For the monolingual settings, the EHMMQA model outperforms the baseline model for the EHMQuAD dataset exhibiting the highest EM (F1) values, achieving 64.48 (69.72) for English and 59.69 (64.93) for Hindi. Conversely, the MMQA dataset exhibited the lowest EM (F1) scores for both

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 8.** Performance analysis of proposed model (monolingual settings—experimental setup-I) with the other various models

| | MMQA | | Translated SQuAD | | XQuAD | | MLQA | | EHMQuAD | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ |
| Model | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) |
| IR model | 34.44 (40.52) | 33.59 (39.21) | 37.60 (42.08) | 34.20 (39.51) | 38.23 (43.96) | 34.88 (39.37) | 37.94 (42.21) | 34.32 (39.90) | 39.28 (44.56) | 37.11 (42.37) |
| Monolingual (English) | 43.17 (49.35) | 35.52 (41.11) | 55.56 (60.29) | 44.65 (49.16) | 56.66 (61.47) | 45.42 (50.96) | 55.82 (60.57) | 45.15 (50.64) | 57.03 (61.97) | 47.82 (52.23) |
| Monolingual (Hindi) | 36.20 (42.53) | 40.71 (45.79) | 46.69 (51.08) | 50.34 (55.87) | 47.42 (52.96) | 51.26 (57.83) | 47.09 (52.72) | 50.88 (56.28) | 49.18 (53.89) | 53.10 (56.28) |
| Deep CCA | 41.22 (46.48) | 37.79 (43.23) | 46.44 (52.12) | 42.84 (48.79) | 47.23 (53.05) | 43.66 (49.56) | 47.04 (53.91) | 43.34 (50.09) | 49.65 (54.19) | 45.49 (49.90) |
| DNN model | 43.78 (49.27) | 41.46 (47.14) | 56.95 (62.92) | 53.04 (58.17) | 57.10 (62.02) | 54.93 (59.07) | 57.49 (62.76) | 53.49 (58.81) | 59.62 (64.31) | 55.07 (59.45) |
| Proposed model | **47.86 (53.19)** | **44.11 (50.60)** | **61.22 (66.31)** | **57.95 (63.40)** | **62.67 (67.85)** | **59.01 (64.57)** | **62.13 (67.18)** | **58.52 (63.07)** | **64.48 (69.72)** | **59.69 (64.93)** |

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 9.** Performance analysis of proposed model (multilingual settings—experimental setup-I) with the other various models

| | MMQA | | Translated SQuAD | | | | MLQA | | EHMQuAD | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $Q_{en} - C_{(en+hi)}$ | $Q_{hi} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{en} - C_{(en+hi)}$ | $Q_{hi} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{en} - C_{(en+hi)}$ | $Q_{hi} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{en} - C_{(en+hi)}$ | $Q_{hi} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ |
| Model | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) |
| IR model | 32.17 (39.67) | 30.78 (37.97) | 33.25 (39.93) | 36.95 (39.50) | 33.44 (37.85) | 38.74 (42.36) | 37.61 (41.05) | 34.10 (38.68) | 39.14 (43.27) | 39.23 (42.69) | 36.45 (40.86) | 41.25 (46.02) |
| Monolingual (English) | 39.18 (46.64) | 35.17 (41.29) | 40.56 (45.37) | 54.23 (58.35) | 47.46 (51.65) | 55.98 (60.28) | 55.02 (59.97) | 48.16 (52.73) | 56.48 (60.98) | 57.67 (60.79) | 49.94 (54.30) | 58.21 (63.14) |
| Monolingual (Hindi) | 38.31 (44.61) | 38.71 (44.94) | 41.22 (45.73) | 49.38 (53.25) | 50.71 (55.84) | 53.53 (59.15) | 50.04 (53.80) | 51.17 (55.43) | 54.24 (59.83) | 52.41 (57.37) | 53.20 (58.05) | 56.36 (61.38) |
| Deep CCA | 39.76 (42.23) | 38.23 (42.19) | 42.68 (45.90) | 52.82 (56.68) | 48.47 (53.14) | 55.36 (60.07) | 53.56 (57.21) | 49.15 (53.97) | 56.12 (60.79) | 55.81 (59.95) | 51.03 (55.79) | 58.29 (63.58) |
| DNN model | 42.28 (47.01) | 40.06 (47.49) | 44.64 (50.88) | 54.63 (58.26) | 53.22 (58.91) | 57.76 (62.24) | 55.38 (59.06) | 53.85 (58.48) | 58.41 (63.93) | 57.64 (61.82) | 56.23 (60.84) | 60.17 (65.02) |
| EHMMQA model | **46.55 (51.32)** | **45.01 (51.61)** | **49.36 (54.69)** | **58.59 (62.40)** | **57.09 (62.17)** | **62.45 (67.82)** | **59.39 (64.29)** | **57.91 (62.90)** | **63.16 (68.40)** | **61.57 (66.61)** | **60.12 (65.32)** | **65.51 (71.28)** |

**Table 10.** Performance analysis of proposed model (monolingual settings—experimental setup-II) with the other various models

| | EHMQuAD | | |
|---|---|---|---|
| | $Q_{en} - C_{en}$ | $Q_{hi} - C_{hi}$ | $Q_{mr} - C_{mr}$ |
| Model | EM (F1) | EM (F1) | EM (F1) |
| IR model | 45.28 (50.67) | 43.98 (48.38) | 42.72 (47.15) |
| Monolingual (English) | 57.39 (62.88) | 45.22 (51.77) | 43.62 (48.03) |
| Monolingual (Hindi) | 46.48 (51.21) | 53.92 (58.68) | 43.29 (47.33) |
| Monolingual (Marathi) | 45.08 (50.15) | 42.24 (47.89) | 47.26 (51.91) |
| Deep CCA | 53.62 (58.23) | 50.58 (55.84) | 45.92 (50.90) |
| DNN model | 59.38 (64.47) | 55.95 (60.11) | 49.18 (54.77) |
| EHMMQA model | **65.82 (70.65)** | **61.59 (66.74)** | **53.47 (57.67)** |

English and Hindi, with values of 47.86 (53.19) and 44.11 (50.60), respectively. This can be attributed to the fact that these four datasets apart from MMQA, have larger contexts, allowing the model to capture a more comprehensive question-aware-context representation. The proposed EHMMQA model showcased significant improvements in both EM and F1-score compared to the baseline models for multilingual settings. Among all four datasets, the EHMMQA model achieved the best results on the EHMQuAD dataset, while the MMQA dataset recorded the lowest performance.

- ***Monolingual and Multilingual settings-Experimental Setup-II:*** In this setup, the EHMMQA model incorporates the Marathi language, alongside English and Hindi, to enhance its multilingual capabilities. The model is trained and validated on the EHMQuAD dataset, which comprises samples in all three languages. The trained model is then evaluated on the test set of 6K QA pairs to measure the performance of the proposed model. Remarkably, the EHMMQA model surpasses all baseline models in monolingual and multilingual settings, exhibiting superior results. Compared to the DNN model, the proposed framework exhibits significant improvements, achieving an average increase of 4.5 points in settings where the Marathi language is absent. However, in settings involving the Marathi language, the average increase is only 2.8 points. This can be attributed to certain challenges encountered while translating Marathi instances. It should be noted that certain inconsistencies slightly hamper the performance of the EHMMQA model. For example, during translation, spaces within a year might be incorrectly separated (e.g., 1986 becomes 1 98 6). Additionally, numerical values may be converted into the Devanagari script but not properly back-translated into the Latin script (e.g., 23 is translated as २३). These discrepancies contribute to a slight degradation in performance.

#### *5.1.1 Statistical test results*

The statistical test results confirm whether the improvement of the EHMMQA model over the baseline models is statistically significant or not significant. The improvement of the result is confirmed by using the Mann–Whitney U Test based on a significant level of 95% (i.e., 0.05) and a two-tailed hypothesis. The statistical results of the EHMMQA model over baseline models in terms of the EM and F1-score are presented in Table 12. The table has $p$-value and $z$-score information, where two sample sets are significant if $p$-value is less than 0.05, and therefore, the null

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 11.** Performance analysis of proposed model (multilingual settings—experimental setup-II) with the other various models

| | EHMQuAD | | | | | | |
|---|---|---|---|---|---|---|---|
| | $Q_{en} - C_{(en+hi+mr)}$ | $Q_{hi} - C_{(en+hi+mr)}$ | $Q_{mr} - C_{(en+hi+mr)}$ | $Q_{(en+hi)} - C_{(en+hi+mr)}$ | $Q_{(en+mr)} - C_{(en+hi+mr)}$ | $Q_{(hi+mr)} - C_{(en+hi+mr)}$ | $Q_{(en+hi+mr)} - C_{(en+hi+mr)}$ |
| Model | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) |
| IR model | 44.65 (49.21) | 42.78 (47.19) | 41.51 (46.12) | 43.34 (47.87) | 42.98 (47.48) | 43.67 (47.81) | 45.17 (50.58) |
| Monolingual (English) | 54.30 (58.04) | 45.78 (50.45) | 42.63 (47.54) | 53.70 (58.47) | 51.64 (55.86) | 44.52 (49.18) | 55.33 (61.03) |
| Monolingual (Hindi) | 46.26 (50.81) | 53.01 (58.27) | 44.13 (48.84) | 50.39 (54.77) | 46.51 (51.19) | 49.77 (54.44) | 54.26 (60.02) |
| Monolingual (Marathi) | 44.58 (49.49) | 45.23 (50.97) | 48.59 (53.04) | 44.61 (49.35) | 46.89 (51.86) | 47.42 (52.09) | 49.83 (55.20) |
| Deep CCA | 53.43 (57.31) | 50.58 (55.27) | 50.42 (54.83) | 51.63 (55.55) | 50.40 (55.24) | 50.86 (55.28) | 57.97 (62.54) |
| DNN model | 56.97 (61.33) | 54.32 (58.96) | 51.18 (56.32) | 54.54 (59.14) | 51.34 (55.62) | 51.93 (56.06) | 60.95 (65.45) |
| EHMMQA model | **63.03 (68.37)** | **61.81 (66.33)** | **57.25 (62.66)** | **61.73 (67.28)** | **56.58 (60.94)** | **56.39 (61.85)** | **66.15 (71.37)** |

**Table 12.** Statistical test results showing the comparison between the proposed model and the baseline model

| | Measure | EM | F1 | Measure | EM | F1 | Measure | EM | F1 |
|---|---|---|---|---|---|---|---|---|---|
| Baseline models | Experimental Setup-I Monolingual settings | | | Experimental Setup-I Multilingual settings | | | Experimental Setup-II Multilingual settings | | |
| IR model | *p*-value | 0.00018 | 0.00018 | *p*-value | 0.00001 | 0.00001 | *p*-value | 0.00214 | 0.00214 |
| | *z*-value | 3.74185 | 3.74185 | *z*-value | 4.12805 | 4.12805 | *z*-value | 3.06661 | 3.06661 |
| | Significant | √ | √ | Significant | √ | √ | Significant | √ | √ |
| Monolingual (English) model | *p*-value | 0.00466 | 0.00362 | *p*-value | 0.01208 | 0.01016 | *p*-value | 0.00214 | 0.00328 |
| | *z*-value | 2.83473 | 2.91030 | *z*-value | 2.51147 | 2.56921 | *z*-value | 3.06661 | 2.93883 |
| | Significant | √ | √ | Significant | √ | √ | Significant | √ | √ |
| Monolingual (Hindi) model | *p*-value | 0.00578 | 0.00578 | *p*-value | 0.01016 | 0.00736 | *p*-value | 0.00214 | 0.00214 |
| | *z*-value | 2.75914 | 2.75914 | *z*-value | 2.56921 | 2.68468 | *z*-value | 3.06661 | 3.06661 |
| | Significant | √ | √ | Significant | √ | √ | Significant | √ | √ |
| Monolingual (Marathi) model | *p*-value | - | - | *p*-value | - | - | *p*-value | 0.00214 | 0.00214 |
| | *z*-value | - | - | *z*-value | - | - | *z*-value | 3.06661 | 3.06661 |
| | Significant | - | - | Significant | - | - | Significant | √ | √ |
| Deep CCA model | *p*-value | 0.001 | 0.001 | *p*-value | 0.01016 | 0.01208 | *p*-value | 0.00736 | 0.00496 |
| | *z*-value | 3.28829 | 3.28829 | *z*-value | 2.56921 | 2.51147 | *z*-value | 2.68328 | 2.81106 |
| | Significant | √ | √ | Significant | √ | √ | Significant | √ | √ |
| DNN model | *p*-value | 0.02088 | 0.01732 | *p*-value | 0.04036 | 0.03486 | *p*-value | 0.0151 | 0.01046 |
| | *z*-value | 2.30558 | 2.38118 | *z*-value | 2.04959 | 2.10733 | *z*-value | 2.42773 | 2.55551 |
| | Significant | √ | √ | Significant | √ | √ | Significant | √ | √ |

hypothesis is rejected. For monolingual settings of experimental setup-II, statistical significance between the EHMMQA model over the baseline model is not calculated because the number of samples is only three, and this test required at least five samples. However, it can be observed from the results obtained for monolingual settings of experimental setup-II that the proposed model has significant improvement (for EM) of around 10.84%, 10.08%, and 8.72% for English, Hindi, and Marathi languages, respectively. In all other cases, the model has significantly improved compared to all baseline models.

### *5.2 Ablation study*

A comprehensive evaluation of different word representation models, including FT + W (Bojanowski *et al.* 2017), FT + WC (Grave *et al.* 2018), mBERT (Pires *et al.* 2019), and the IndicFT (Kakwani *et al.* 2020) model, is done to assess their performance on MQA datasets in the context of the Indic language. The IndicFT model stood out with its superior performance, demonstrating its potential for effectively representing and understanding the Indic language. The underlying reasons for this superior performance can be attributed to multiple factors.

- Language-specific training: The IndicFT model is specifically designed and trained for the Indic language. It is trained on IndicCorp, the largest publicly available monolingual corpora of the Indic language. This language-specific training allows the model to capture the unique linguistic patterns, semantic nuances, and syntactic structures specific to the Indic language. In contrast, the other models have been trained on more general or diverse datasets, which might not adequately capture the intricacies of the Indic language.
- Corpus size: The size of the training corpus plays a crucial role in the performance of word representation models. IndicFT benefits from being trained on large monolingual corpora, IndicCorp. The larger the training corpus, the more exposure the model has to a wide range of language contexts and variations, leading to better representation learning.
- Fine-tuning: It refers to adapting a pretrained model to a specific task or domain. The IndicFT model might have undergone fine-tuning specifically for the MQA datasets or the Indic language tasks, allowing it to better align with the requirements of the evaluation task. The other models, such as mBERT, FT + W, and FT + WC, are not fine-tuned specifically for the MQA datasets or the Indic language, leading to a potential performance gap.
- Model architecture and training techniques: The differences in model architecture and training techniques can also contribute to the performance variations. The IndicFT model employs specific architectural choices or training strategies better suited for capturing the complexities of the Indic language. These choices could include attention mechanisms, contextual embeddings, or optimization methods specifically tailored to the Indic language.

Overall, the better performance of the IndicFT model compared to mBERT, FT + W, and FT + WC models can be attributed to its language-specific training on a large monolingual corpus, potential fine-tuning on the evaluation task, and architectural and training choices that are specifically optimized for the Indic language. Table 13 reports the ablation study of the various word representation models on the MQA system.

## 6. Conclusion and future scope

MQA is an application of NLP that answers the user's question irrespective of the language. Although the MQA systems used zero-shot/few-shot, translate-train, and translate-test approaches to transfer knowledge from English to other languages, these systems do not provide

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Table 13.** Ablation study of various word representations

| | MMQA | Translated SQuAD | MLQA | EHMQuAD | EHMQuAD |
|---|---|---|---|---|---|
| | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{(en+hi)} - C_{(en+hi)}$ | $Q_{(en+hi+mr)} - C_{(en+hi+mr)}$ |
| Model | EM (F1) | EM (F1) | EM (F1) | EM (F1) | EM (F1) |
| IndicFT | 49.36 (54.69) | 62.45 (67.82) | 63.16 (68.40) | 65.51 (71.28) | 66.15 (71.37) |
| FT + W | 46.25 (51.23) | 60.59 (65.37) | 61.41 (65.76) | 62.82 (67.90) | 63.68 (68.73) |
| FT + WC | 47.66 (52.62) | 61.31 (65.85) | 62.23 (67.61) | 63.76 (69.09) | 64.34 (69.49) |
| mBERT | 48.94 (53.58) | 61.72 (66.69) | 62.79 (68.20) | 64.04 (68.18) | 64.88 (68.94) |

the abstract structure to understand multiple languages. Also, the lack of benchmark datasets for low-resource languages hinders the growth of MQA systems. To overcome these challenges, the proposed EHMQuAD dataset is created for English, Hindi, and Marathi languages using the TAR approach, and the proposed EHMMQA model is developed. This framework is an abstract structure that can be adaptable to any number of different languages. The EHMMQA model employs a deep neural network approach considering English, Hindi, and Marathi languages. The results obtained are significant on the MQA datasets and are considered state-of-the-art performance. The specific concern regarding the improvement in the performance of the system is to be addressed in the future. Multilingual complex, descriptive, and reasoning questions are a QA system's major concern and can be handled in the future.

**Competing interest.** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

**Andrew G., Arora R., Bilmes J. and Livescu K.** (2013). Deep canonical correlation analysis. In *Proceedings of the 30th International Conference on Machine Learning. Proceedings of Machine Learning Research*, vol. 28, pp. 1247–1255, *Proceedings of Machine Learning Research*, PMLR.

**Artetxe M., Ruder S. and Yogatama D.** (2020). On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4623–4637.

**Baradaran R., Ghiasi R. and Amirkhani H.** (2022). A survey on machine reading comprehension systems. *Natural Language Engineering* **28**(6), 1–50.

**Bhattacharyya P.** (2017, IndoWordNet. Springer Singapore, pp. 1–18.

**Bojanowski P., Grave E., Joulin A. and Mikolov T.** (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* **5**, 135–146.

**Carrino C. P., Costa-jussà M. R. and Fonollosa J. A.** (2020). Automatic Spanish translation of SQuAD dataset for multilingual question answering. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 5515–5523.

**Chandra A., Fahrizain A., Laufried S. W. and et al.** (2021). A survey on non-English question answering dataset, arXiv preprint arXiv:2112.13634.

**Chang C.-H., Kayed M., Girgis M. R. and Shaalan K. F.** (2006). A survey of web information extraction systems. *IEEE Transactions on Knowledge and Data Engineering* **18**(10), 1411–1428.

**Chen Y., Zhang Y., Hu C. and Huang Y.** (2021). Jointly extracting explicit and implicit relational triples with reasoning pattern enhanced binary pointer network. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5694–5703.

**Clark J. H., Choi E., Collins M., Garrette D., Kwiatkowski T., Nikolaev V. and Palomaki J.** (2020). TyDiQA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics* **8**, 454–470.

**Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V.** (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Conneau A. and Lample G.** (2019). Cross-lingual language model pretraining. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 7059–7069.

**Devlin J.**, **Chang M.-W.**, **Lee K. and Toutanova K.** (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1 (Long and Short Papers), pp. 4171–4186.

**Dzendzik D.**, **Foster J. and Vogel C.** (2021). English machine reading comprehension datasets: A survey. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 8784–8804.

**García-Cumbreras M.**, **Martínez-Santiago F. and Ureña-López L.** (2012). Architecture and evaluation of BRUJA, a multilingual question answering system. *Information Retrieval* **15**(5), 413–432.

**García Santiago M. D. and Olvera-Lobo M. D.** (2010). Automatic web translators as part of a multilingual question-answering (QA) system. *Translation Journal* **14**(1).

**Grave E.**, **Bojanowski P.**, **Gupta P.**, **Joulin A. and Mikolov T.** (2018). Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). European Language Resources Association (ELRA).*

**Gupta D.**, **Ekbal A. and Bhattacharyya P.** (2019). A deep neural network framework for English Hindi question answering. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)* **19**(2), 1–22.

**Gupta D.**, **Kumari S.**, **Ekbal A. and Bhattacharyya P.** (2018). MMQA: A multi-domain multi-lingual question-answering framework for English and Hindi. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

**Gupta S. and Khade N.** (2020). BERT based multilingual machine comprehension in English and Hindi. arXiv preprint arXiv:2006.01432.

**Hasan M. A.**, **Alam F.**, **Chowdhury S. A. and Khan N.** (2019). Neural machine translation for the Bangla-English language pair. In *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6.

**Hermann K. M.**, **Kocisky T.**, **Grefenstette E.**, **Espeholt L.**, **Kay W.**, **Suleyman M. and Blunsom P.** (2015). Teaching machines to read and comprehend. *Advances in Neural Information Processing Systems* **28**.

**Hládek D.**, **Staš J.**, **Juhár J. and Koctúr T.** (2023). Slovak dataset for multilingual question answering. *IEEE Access* **11**, 32869–32881.

**Johnson M.**, **Schuster M.**, **Le Q. V.**, **Krikun M.**, **Wu Y.**, **Chen Z.**, **Thorat N.**, **Viégas F.**, **Wattenberg M.**, **Corrado G. and et al.** (2017). Google's multilingual neural machine translation system: enabling zero-shot translation. *Transactions of the Association for Computational Linguistics* **5**, 339–351.

**Kakwani D.**, **Kunchukuttan A.**, **Golla S.**, **G. N. C.**, **Bhattacharyya A.**, **Khapra M. M. and Kumar P.** (2020). IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of EMNLP.*

**Klein G.**, **Kim Y.**, **Deng Y.**, **Senellart J. and Rush A.** (2017). OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pp. 67–72.

**Lee C.-H. and Lee H.-Y.** (2019). Cross-lingual transfer learning for question answering, arXiv preprint arXiv:1907.

**Lee J.**, **Cho K. and Hofmann T.** (2017). Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics* **5**, 365–378.

**Lewis P.**, **Oguz B.**, **Rinott R.**, **Riedel S. and Schwenk H.** (2020). MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7315–7330.

**Liu Y.**, **Ott M.**, **Goyal N.**, **Du J.**, **Joshi M.**, **Chen D.**, **Levy O.**, **Lewis M.**, **Zettlemoyer L. and Stoyanov V.** (2019). RoBERTa: A robustly optimized Bert pretraining approach. arXiv preprint arXiv:1907.

**Longpre S.**, **Lu Y. and Daiber J.** (2021). MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *Transactions of the Association for Computational Linguistics* **9**, 1389–1406.

**Mishra A. and Jain S. K.** (2016). A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences* **28**(3), 345–361.

**Papineni K.**, **Roukos S.**, **Ward T. and Zhu W.-J.** (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318.

**Pires T.**, **Schlinger E. and Garrette D.** (2019). How multilingual is multilingual BERT? In*Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4996–5001.

**Rajpurkar P.**, **Jia R. and Liang P.** (2018). Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Melbourne, Australia: Association for Computational Linguistics, pp. 784–789.

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

**Ramesh G.**, **Doddapaneni S.**, **Bheemaraj A.**, **Jobanputra M.**, **R. A. K.**, **Sharma A.**, **Sahoo S.**, **Diddee H.**, **Kakwani D.**, **Kumar N.**, **et al.** (2022). Samanantar: The largest publicly available parallel corpora collection for 11 Indic languages. *Transactions of the Association for Computational Linguistics* **10**, 145–162.

https://doi.org/10.1017/nlp.2024.12 Published online by Cambridge University Press

**Rogers A.**, **Gardner M. and Augenstein I.** (2023). QA dataset explosion: A taxonomy of NLP resources for question answering and reading comprehension. *ACM Computing Surveys* **55**(10), 1–45.

**Shi X.**, **Huang H.**, **Jian P. and Tang Y.-K.** (2021). Improving neural machine translation by achieving knowledge transfer with sentence alignment learning. *Neurocomputing* **420**, 15–26.

**Singhal A. and et al.** (2001). Modern information retrieval: A brief overview. *IEEE Data Engineering Bulletin* **24**(4), 35–43.

**Türe F. and Boschee E.** (2016). Learning to translate for multilingual question answering. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 573–584.

**Vaswani A.**, **Shazeer N.**, **Parmar N.**, **Uszkoreit J.**, **Jones L.**, **Gomez A. N.**, **Kaiser L. and Polosukhin I.** (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* **30**, 1–11.

**Wu Y.**, **Schuster M.**, **Chen Z.**, **Le Q. V.**, **Norouzi M.**, **Macherey W.**, **Krikun M.**, **Cao Y.**, **Gao Q.**, **Macherey K.**, **et al.** (2016). Google's neural machine translation system: Bridging the gap between human and machine translation, arXiv preprint arXiv:1609.08144.

**Xia H.**, **Ding C. and Liu Y.** (2020). Sentiment analysis model based on self-attention and character-level embedding. *IEEE Access* **8**, 184614–184620.

**Xue L.**, **Constant N.**, **Roberts A.**, **Kale M.**, **Al-Rfou R.**, **Siddhant A.**, **Barua A. and Raffel C.** (2021). mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 483–498.

**Zhou Y.**, **Geng X.**, **Shen T.**, **Zhang W. and Jiang D.** (2021). Improving zero-shot cross-lingual transfer for multilingual question answering over knowledge graph. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5822–5834.

**Cite this article:** Lahoti P, Mittal N and Singh G (2025). EHMMQA: English, Hindi, and Marathi multilingual question answering framework using deep learning. *Natural Language Processing* **31**, 346–374. https://doi.org/10.1017/nlp.2024.12