QA1: EDUQA: EDUCATIONAL DOMAIN QUESTION ANSWERING SYSTEM USING CONCEPTUAL NETWORK MAPPING

Introduction: This paper reviews the concept of dividing the whole process into 3 chunks of processes
1:Entity recognition
2:Question Analysis that filters relevant features
3:Answer Retrieval for extracting the answer based on the above two processes.

Summary: The Entity Recognition is done using the DCN (Dynamic Concept Network). the main task of this module is to extract entities and their relationships
The list of the entities are passed to the Question Analysis module.
This module takes the input as a question and then tokenizes it and extracts the longest prefix sequence from the entity list provided by the DCN module.
After this the information is passed on to the answer retrieval module which tries to match the entities with the most relevant relationship i.e it extracts the
relationships from the entities found by the Question analysis module. If we find any relation then it is passed to the concept network to extract the answer and
then it's given back to the user.
But if we don't find a relation, then the person has to find the answer and mark it and this is updated in the concept network. This process is called as on the fly learning.

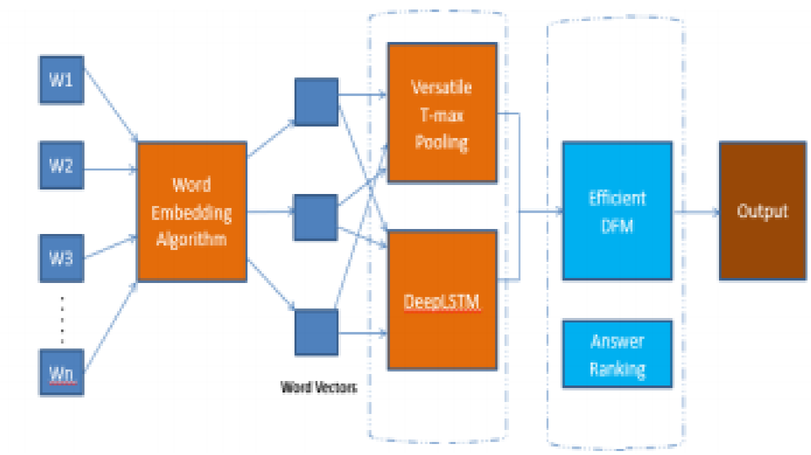
Conclusion: The model was able to correctly answer 80% of the definition based questions and 65% of the other type of questions.

QA2: A Novel Approach for Semantic Similarity Measurement for High Quality Answer Selection in Question Answering using Deep Learning Methods

Introduction : This paper proposes a complete neural based architecture for the QA models i.e there is no NLP included and all the understanding is done
solely by the neural network.
In this paper, there are three steps too:
1: Question analysis
2: Document retrieval
3: Ranking of Answers.

Summary: Here, first the questions are converted into word vectors using the word embedding matrix, they have used and tried of different word embedding matrices
like Word2Vec, fastText, Glove, Baroni, SL999. Among these the highest accuracy was gained by the SL999 matrix due to its wide range of word reach.

The next step of this pipelines consists of 2 important parts of the pipeline, the prediction is done using two techniques mainly Versatile global T=max pooling and
Deep LSTM. Once the prediction is obtained based on the model it is given to the efficient DFM which filters the answers based on ranks. The model was trained on
4 different types of datasets like STSB, wikipedia QA dataset,MRPC and SICK datasets. the average accuracy scored on all of them was about 82-83%
And also the maximum accuracy is always obtained when we use the SL999 word embeddings.

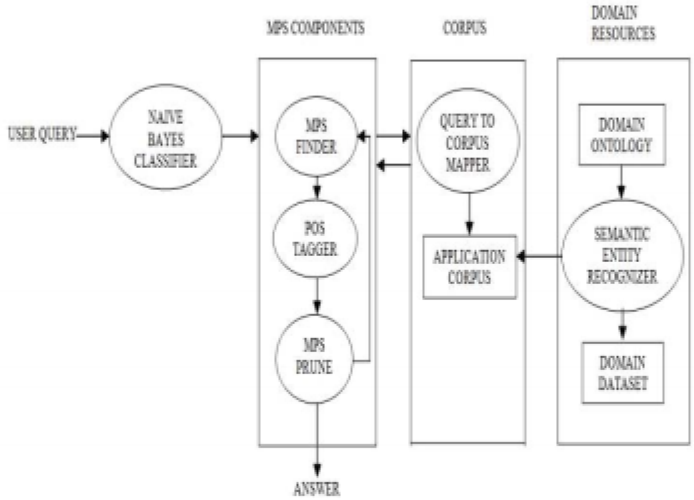


Conclusion: In this Research paper, they provided a Versatile global T-max pooling and DeepLSTM for quality answer prediction.
They have additionally used Efficient DFM to forecast the nice solutions and specially DFM is used for ranking cause.

QA3: Semantics-Enhanced Answer Selection in Closed-domain Question Answering System

Introduction: This paper stresses on the Information retrieval task in question answering systems**.**The overall system architecture of the proposed system consists of three main components namely Domain Resources, Corpus and MPS
Components.

Summary: The Domain Resources refer to the Knowledge and the Database one currently possesses related to the Domain. These act as a backend to the QA system. The Domain Dataset is the document which is to be queried by the users. It can be in any text format.
The next step in this pipeline is the Corpus, that provided the knowledge about the name-entity relation to the model. Here the features of the query is extracted and then given to the next phase of the pipeline that is the MPS(Most probable Sentence). As we know that the features may correspond to various sentences, here the MPS selected the most probable sentence as its name suggests. The features of the query are compared against those of all the sentences in the Domain Dataset and returns the best sentence to the user. The first model that is the Naive Bayes classifier just classified the question into some types of questions so that it is easy for the MPS to find the sentence.



Conclusion: The complete process obtains an overall accuracy of 71% on the real life data as per tested on 500 wikipedia questions that were manually prepared. This system offers betterment of user request along with the improved specification and matching techniques.

QA4: Semantics-Enhanced Answer Selection in Closed-domain Question Answering System

Introduction: This paper proposes an architecture for retrieval of answers based on many documents, for example you have 10 documents and you ask a question, the model will search the most relevant document as well as such the most relevant answer from those documents.

Summary: The first step in this pipeline is to receive the question from the user, after this the type of the answer it wants is detected. Once we have this it extracts the names and the entity from the query and detects the keywords from the query. Once we have all this information some nlp techniques are used to get the meta information too and the final query is formed bringing all this together. This meta data is matched with the documents we have and the most relevant documents are extracted using this metadata. After this the most relevant passages are retrieved from those documents and every passage is given a score based on their relevance of the query. This is passed to the Local Proximity Prioritizer. The goal of this component is to highlight candidate answers that show a denser distribution of matches in their text.
After this to filter out more answers we pass it to the Keywords Overlapping module. Here, a metric is used to measure similarity between texts is defined, which assesses the percentage of overlapping words between the candidate answer and the question text. If the overlapping percentage is greater than a configurable threshold, the score for the candidate answer is increased. The default threshold value is 33%.

This is one of the final stages of the pipeline. Once we have obtained the filtered answer, we pass it to the name entity module which matches the NE of the query and the answer. In case of a match the score is incremented, giving more relevance to the candidate answer. Finally the answer with the highest score is obtained.

Conclusion: The framework is able to retrieve a correct answer for 80% of the questions in the dataset. Precisely it gives the right answer, as first answer, in 66% of the cases.

QA5: Learning to Rank Answers to Closed-Domain Questions by using Fuzzy Logic

Introduction:

Question answering (QA) is a challenging task and has received considerable attention in the last years. Selecting one or more answers from a list of candidate answers have been given utmost importance. This paper proposes a fuzzy approach for ranking and selecting the correct answer among a list of candidates in a state-of-the-art QA system operating with factoid and description questions on Italian corpora pertaining to a closed domain.

Paper Summary:

In detail, the proposed approach is able to learn fuzzy rule based models, by means of which the scores, measuring answer evidence affected by uncertainty and vagueness, are transformed into a unique confidence grade, according to which the answers are given a ranking order, which is used to choose the correct answer. Fuzzy ranking models can explicitly show which measures better explain differences between correct and incorrect answers, depending on the question type. The Architecture employed here consists of 4 modules:

1. **Indexing**: It provides the documents with annotations and respective indexing.
2. **Questions Processing:** This phase processes the question text and extracts a set of features, to generate a query for the IR engine. It has components like ANswer Type Detection, Stop Word Removal, NLP Metadata Extraction, and Query Formulation.
3. **Information Retrieval:** The search engine retrieves a set of documents satisfying the query, and spits them into sentences.
4. **Answer Selection:** This module is made of a collection of components, each one assigning a score to the candidate answers.

These components include Named entity matching, Keywords Overlapping, Proximity with Gap, Proximity Distance Highlighted Text to LAT, Multiple LAT discourager.

Results:

Five fuzzy ranking models are learned from the supervised questions set, one for each question type; therefore, depending on the question type, a fuzzy model based on single scores is chosen for ranking candidate answers to each question.

TABLE II. ACCURACY@1 OF ANSWERING CORRECTNESS

Method	Question Type					All questions
	Description	*Date*	*Entity*	*Location*	*Person*	
Empirical averaging	0.85	0.75	0.80	0.80	0.85	0.79
LFA	0.91	0.84	0.81	0.84	0.84	0.83

From above Table, it can be evinced that using the proposed approach for ranking answers instead of empirically combining single scores results for most of the question types in an accuracy improvement. If all 907 questions of the dataset are considered, the usage of the proposed approach enables us to answer correctly to 756 instead of 721 questions. In detail, the Likelihood-Fuzzy Analysis has been applied in order to learn fuzzy rule-based models able to discern correct (True) from incorrect answers (False). The different scores, affected by uncertainty, have been transformed into a unique confidence grade, according to which the answers have been ranked in order to determine the best candidate as the first-ranked one.

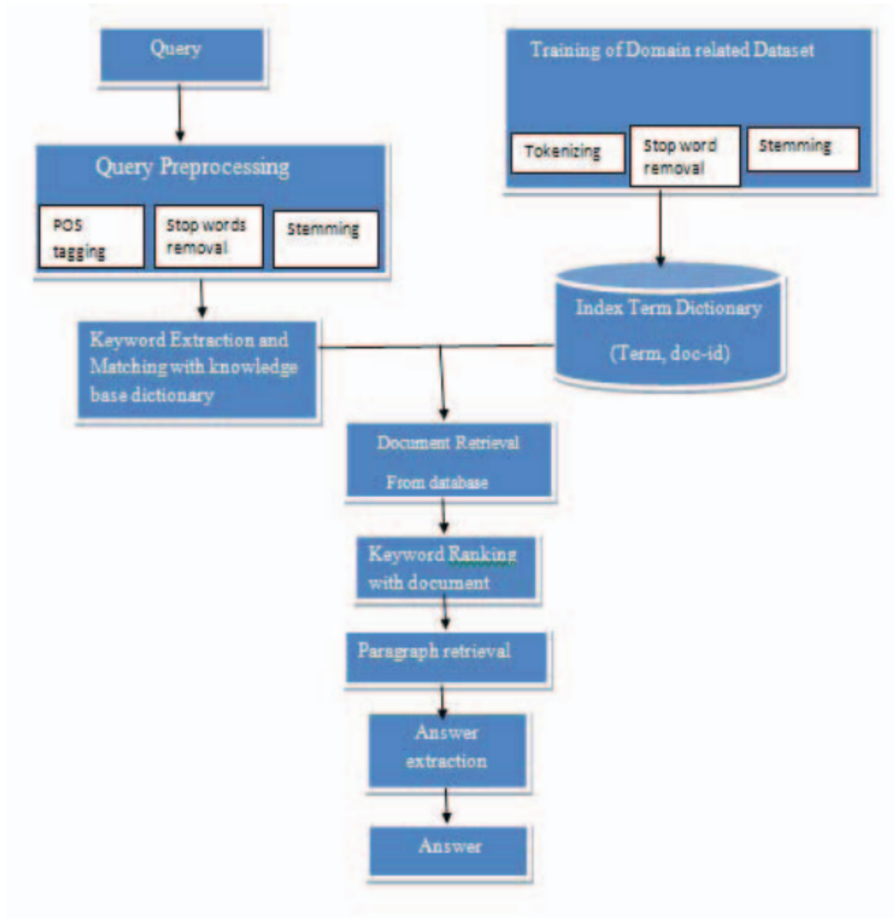
QA6: Question Answering System on Education Acts Using NLP Techniques

Introduction:
This paper proposes an architecture for Question Answering System on Education Acts using NLP. This paper uses CDQA as it gives better accuracy than Open Domain Question Answering Systems.

Architecture:
An input for the proposed system will be a query related to education acts or different information related to education. For example “What is the duty of parents to secure the children's education?”, “What are the funding authorities of school?” The Question keyword is calculated by removing stop words and performing stemming on questions to extract the answer. Metadata will be generated for the dataset related to Education Acts. Using these keywords, the original passage or sentences are tagged to give candidate answers from the answer extractor. According to the given question,the highest score candidate answer will be shown as the final answer. The system will produce the accurate answer for trained questions and then will test to measure the accuracy of untrained questions.

Conclusion:
The QA system for closed domain of documents related to education acts using NLP techniques and information retrieval are proposed to give the accurate and suitably more correct answers for user’ queries.

The following is the architecture diagram for the proposed system:


Query
Training of Domain related Dataset
Tokenizing
Stop word removal
Stemming
Query Preprocessing
POS tagging
Stop words removal
Stemming
Index Term Dictionary
(Term, doc-id)
Keyword Extraction and Matching with knowledge base dictionary
Document Retrieval
From database
Keyword Ranking with document
Paragraph retrieval
Answer extraction
Answer

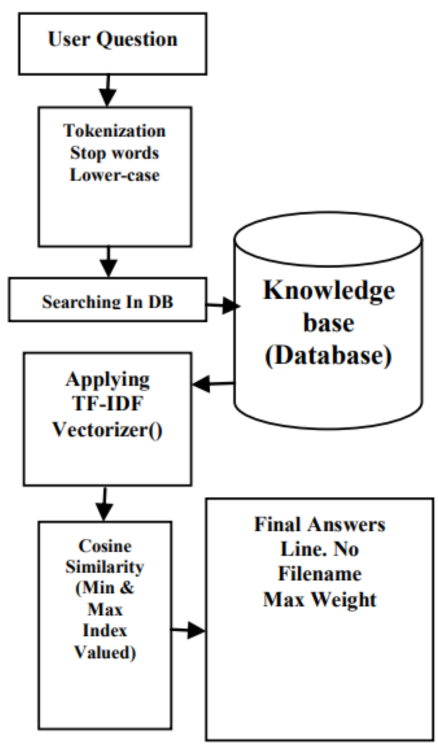
QA7: Evaluating Reasoning in Factoid based Question Answering System by Using Machine Learning Approach

Introduction:

The main purpose of this paper is to provide a Factoid based CDQA system using Machine Learning. Mainly Factoid based Questions consist of 6 types , Wh type like where, which, when, what, why etc. definition questions, Yes-No/True-False Questions, instruction-based questions, explanation questions and list questions. Their system proposes an architecture to solve all types of Factoid Questions.

Summary:

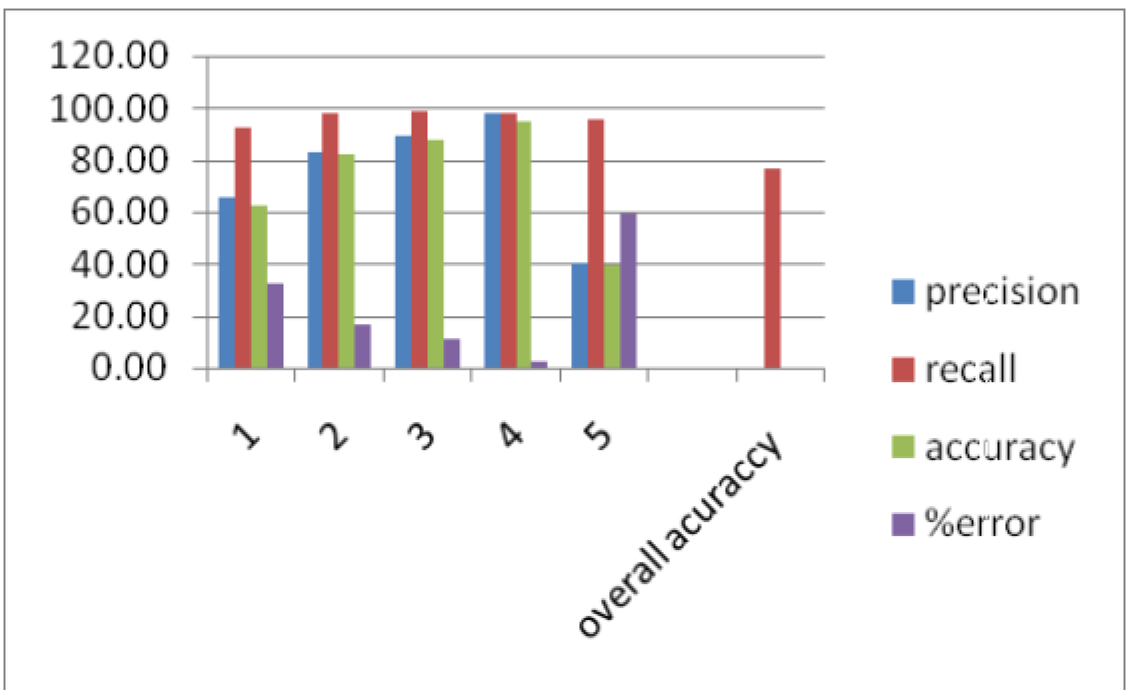
The Proposed Model of the system is :



When a user gives a query, the system pre-process the given question into tokenization, stop-words, converting the given question into lower-case. After cleaning, the system searches the words in the knowledge-based file (database) for example why shivaji was a great man? After preprocessing the words remaining is shivaji great man. It is searched in the database. If the word is found in the database then it is kept in the list and further TF-IDF vectorizer improved algorithms are applied to it. The maximum scored candidate answer is declared as the final answer.

Conclusion:

The following is the result of the proposed system:



They tried to develop a Question Answering System using TF-IDF vectorizer using the Scikit-learn library of Machine Learning. Overall accuracy of their system is 76.32 %.

QA9: Questionator - Automated Question Generation using Deep Learning

Introduction:

The paper proposes a state-of-the-art solution using a pipeline that utilizes natural language processing and image captioning techniques capable of generating questions not only for textual but also for visual inputs. Along with the question, distractors for the generated questions and their answers are also created.

Paper Summary:

Section I	Image Captioning
Section II	Question Generation.
Section III	pipeline of proposed system.

<u>Image Captioning</u>

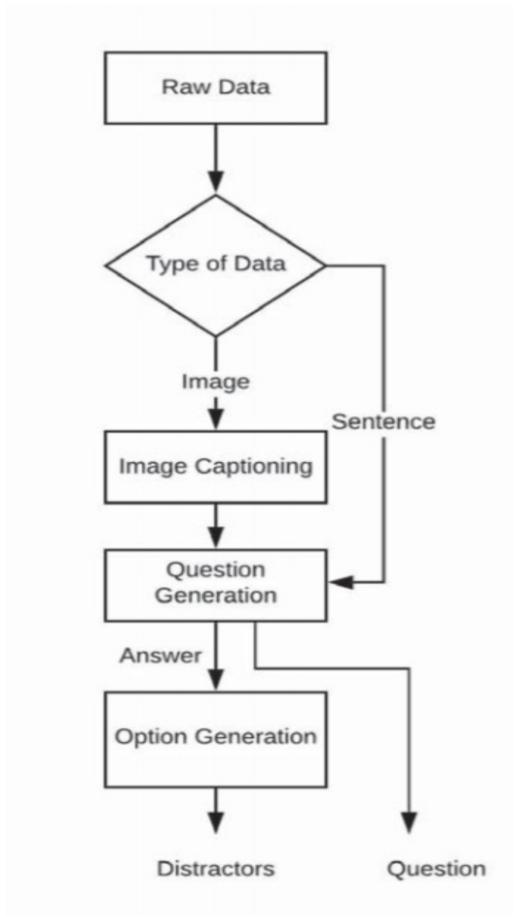
The image captioning module converts a given input image into a natural language description. The natural language provides a good solution for describing the semantic information of images. The caption generated by the image captioning is now fed through the question generation.

<u>Question Generation</u>

The question generation is performed using sentence splitting and Named Entity Recognition. The caption obtained from the previous module is passed through a dependency parser (Stanford CoreNLP), that extracts the subject, object and the predicate in the sentence. These 3 subparts of a sentence are then passed through a POS (Part Of Speech) tagger and a “Wh” question is generated.

MCQ Generation

It suggests question generation as a means for question answering. This module makes use of GloVe to make a Vector representation for words and choose the distractors for the generated questions.



Conclusion

The proposed question generation pipeline has demonstrated that automated generation of questions can be found in the education domain for generating Multiple Choice Questions(MCQ). This is a robust system which not only works for sentences but also for images. Further additions to the captioning dataset , for instance adding complex sentences and then improving the question generation model to generate questions on these complex sentences could yield better results.

QA10. Deep Neural Network Models for Question Classification in Community Question- Answering Forums

Introduction:

In the case of an opinion-based or a yes/no question that wasn't previously answered, an external knowledge source is needed to generate the answer. The paper proposes a LSTM based model that performs question classification into the two aforementioned categories. Given a question as an input, the objective is to classify it into opinion-based or yes/no question. The proposed model was tested on the Amazon community question-answer dataset.

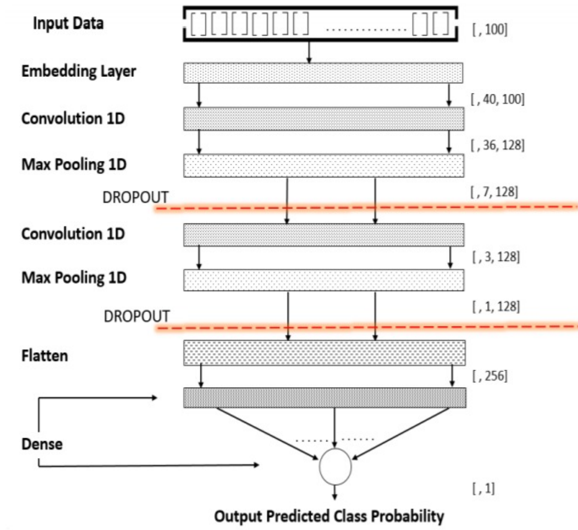
Paper Summary:

Section I	Learning, Soft Computing
Section II	Approach-1 (CNN)
Section III	Approach-2 (LSTM)

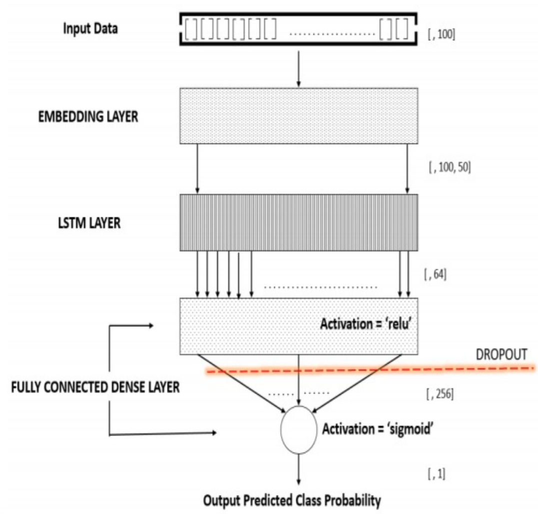
<u>Learning, Soft Computing</u>

Identifying such duplicate and near-duplicate questions automatically and later, automating answer generation for them. When the question is opinion-based or is not currently addressed in the system will the involvement of a human or customer care professional be required.

<u>Approach-1 (CNN)</u>



Approach-2 (LSTM)



Conclusion

The CNN model achieved a maximum accuracy of 72.37% for the electronics category of the dataset, but suffered from overfitting problems. The LSTM model was able to fit the data really well and outperformed CNN by a significant margin by achieving an accuracy of 93.4% for the electronics category of the dataset.

QuGAN: Quasi Generative Adversarial Network for Tibetan Question Answering Corpus Generation

Introduction:

The paper aims to generate a QA corpora for low-resource languages, such as Tibetan. The proposed system is a QA corpus generation model, called QuGAN. This model combines Quasi-Recurrent Neural Networks and Reinforcement Learning.The QRNN used as a generator for GANs, which speeds up the generation of text. At the same time, the reward strategy and Monte Carlo search strategy are optimized to effectively update the generator network. Finally, we use the Bidirectional Encoder Representations from Transformers model to correct the generated questions at the grammatical level.

Paper Summary:

Section I	Generator
Section II	Discriminator
Section III	Reinforcement Learning Optimization
Section IV	Grammar Optimization
Section V	Answer Matching

<u>Generator</u>

Before the model training, MLE(maximum likelihood estimation) on randomly sample data to generate questions more efficiently i.e To derive the maximum probability text sequence. We use the QRNN model as a generator. In QRNN, there are three pooling modes : f-pooling, fo-pooling and ifo-pooling. This paper uses f-pooling.

<u>Discriminator</u>

This paper uses the fundamental LSTM model as the discriminator to judge if the generated text is real. The discriminator scores the generated sequence in whole sentences and feeds the scores back to the generator.

Reinforcement Learning Optimization

Traditional Monte Carlo search is very time consuming. It needs to generate samples every time. Meanwhile, it calculates the previously generated item when calculating some of the later partial sequence reward estimates, resulting in overfit. Therefore, they optimized the Monte Carlo search algorithm and scored the next sequence through the generated partial sequences, so the score of the entire sequence can be quickly obtained.

Grammar Optimization

To eliminate the grammatical errors of the sentences generated by QuGAN, this paper uses the BERT model to modify and optimize the questions, including two parts: random mask and next sentence prediction.

Answer Matching

They used semantic similarity to match question and answer in corpus. The answer with the highest similarity of the question is the answer to the question.

Architecture

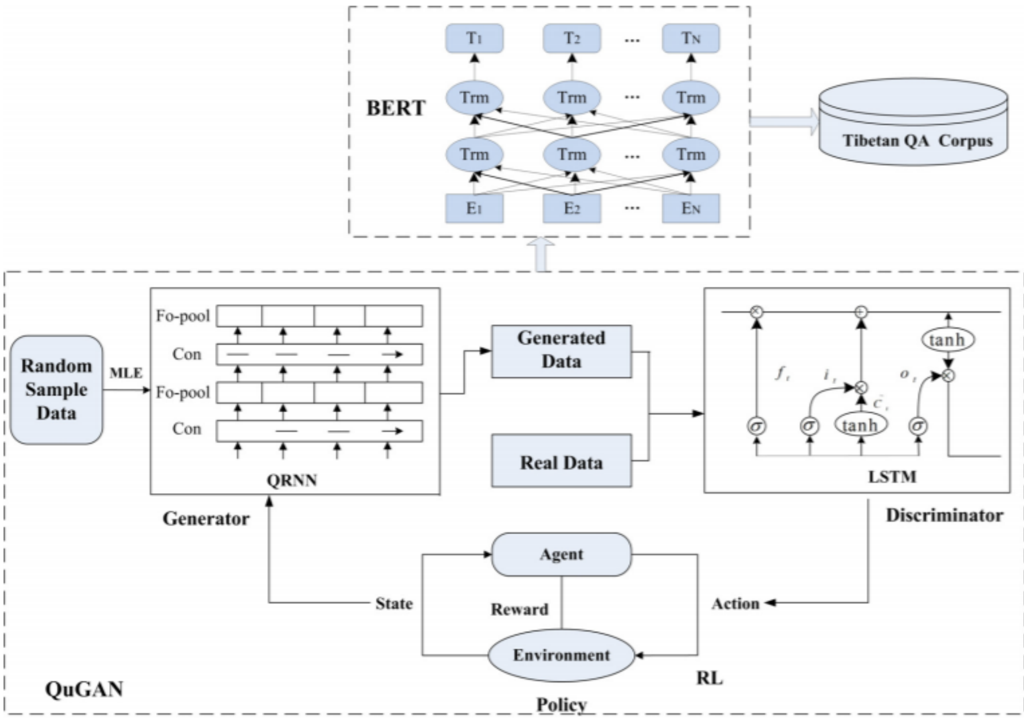


FIGURE 1. **The framework of QuGAN model.**

Results

Random initialization	MLE	Generated by Our Model
སྙན་ངག་གནམ་གཤིས་ཆ་ཚང་། (Poetry, weather, complete.)	ཅི་ཞིག་ལ་སྙན་ངག་ཟེར། $ སྙན་ཞིང་འཇེབས་པ་ཞིག་ལ་ཟེར། (What is poetry? $ Beautiful words, words, sentences.)	ཅི་ཞིག་ལ་སྙན་ངག་ཟེར། $ རྩོམ་གྱི་རིགས་ཤིག (What is poetry? $ A kind of writing.)
དབྱངས་ཞིན་ཕྱིས་དེབ། (What is the practice?)	དཔལ་འགྱོར་དང་དབྱངས་ཟེར། $ སྔོན་འཇུག་དང་རྗེས་འཇུག (What is a consonant letter? $ Prefix words and suffixes.)	ཅི་ཞིག་ལ་དབྱངས་ཟེར། $ གནས་དང་བྱེད་པའི་རྣམ་འགྱུར་མི་གསལ་ཞིང་།རང་ཉིད་གསལ་བྱེད་རྣམས་དང་ཕྲད་པ་ལ་བརྟེན་ནས་མིང་དང་ཚིག་གི་བརྗོད་པའི་སྒྲ་གསལ་པོར་འབྱིན་ཐུབ་ཞིག་ལ་ཟེར། (What is a consonant letter? $ It does not make sense in itself, encountering the base word can be fully issued corresponding tone.)
ཅི་ཞིག་ལ་རྩ་བ་ཟེར། (What's fundamental?)	ཅི་ཞིག་ལ་རྩ་བ་བཞི་ཟེར། $ མ་བྱིན་ལེན། བརྫུན། སྲོག་གཅོད་དེ་བཞི། (What are the four fundamentals? $ No lying, no harm.)	ཅི་ཞིག་ལ་རྩ་བ་ཟེར། $བྱ་དངོས་ཀྱི་འབྱུང་རྩ་དང་གཞི་རྩ་ལ་ཟེར། (What's fundamental? $ The root or basic source or basic of the physical object.)

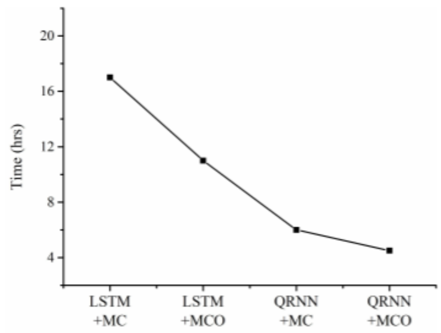


FIGURE 3. **Time consuming of different models.**

Conclusion

In this paper, we propose the QuGAN model which combines QRNN and RL. And the BERT model, optimized reward strategy and Monte Carlo search strategy are used to improve the performance. However, due to not considering the Tibetan grammar characteristics, there are certain invalid questions.