Question-Answer Classification using Machine Learning and Deep Learning Models

Wasi
School of Data and Sciences
Brac University
Dhaka, Bangladesh
julkifl.hasan.wasi@g.bracu.ac.bd

Hossain
School of Data and Sciences
Brac University
Dhaka, Bangladesh
shoaib.hossain1@g.bracu.ac.bd

Abstract

This paper presents a comprehensive study on the classification of question-answer (QA) text into predefined categories using a variety of machine learning and neural network models. We explore different word representation techniques, including Bag of Words (BoW), TF-IDF, GloVe, and Skip-gram, to convert unstructured text data into meaningful numerical features. These features are then used to train and evaluate a suite of models, ranging from traditional machine learning algorithms like Random Forest, Logistic Regression, and Naive Bayes to more advanced deep learning architectures such as Deep Neural Networks (DNN), Simple RNNs, LSTMs, GRUs, and their bidirectional variants. Our experiments show that deep learning models, particularly Bidirectional LSTMs with pre-trained GloVe embeddings, achieve the best performance, highlighting the effectiveness of contextual embeddings for this task.

Keywords—Natural Language Processing, Text Classification, Machine Learning, Neural Networks, Word Embeddings, Question Answering

I. INTRODUCTION

The proliferation of online platforms for information exchange has led to a massive volume of unstructured text data in the form of questions and answers. Automatically classifying this data into relevant categories is crucial for various applications, including content organization, information retrieval, and user experience enhancement. This project aims to address this challenge by systematically evaluating a range of natural language processing (NLP) techniques for QA text classification.

We investigate the performance of both classical machine learning models and more recent neural network architectures. The study is structured around two key components: word representation and classification models. For word representation, we implement and compare four widely-used techniques: Bag of Words (BoW), TF-IDF, GloVe, and Skip-gram. For classification, we train and evaluate a total of 22 models, including three machine learning classifiers and eight neural network architectures, each paired with different word representation methods.

This paper details our methodology, from data preprocessing to model implementation and hyperparameter tuning. We present a comprehensive comparison of the results and discuss the strengths and weaknesses of each approach. The findings of this study provide valuable insights into the effectiveness of different NLP techniques for QA text classification.

II. METHODOLOGY

A. Dataset

The dataset used for this project is the "Question Answer Classification Dataset," which contains QA text from various domains. Each entry in the dataset consists of a question title, question content, and the best answer, all concatenated into a single "QA Text" field. The corresponding "Class" label represents the category of the QA pair. The dataset is balanced across 10 distinct classes: Science & Mathematics, Education & Reference, Politics & Government, Entertainment & Music, Sports, Business & Finance, Society & Culture, Family & Relationships, Computers & Internet, and Health.

B. Data Preprocessing

Before feeding the text data into our models, we performed a series of preprocessing steps to clean and normalize the text. This included:

1. Lowercasing: All text was converted to lowercase to ensure uniformity.
2. Punctuation Removal: All punctuation marks were removed to reduce noise in the data.
3. Stopword Removal: Common English stopwords (e.g., "the," "a," "is") were removed using the NLTK library, as they do not typically contribute to the semantic meaning of the text.

C. Word Representation Techniques

We explored four different techniques to convert the preprocessed text into numerical vectors that can be used by our models:

1. Bag of Words (BoW): This model represents text as a collection of its words, disregarding grammar and word order but keeping track of frequency. We used a CountVectorizer with a max_features limit of 5000.
2. TF-IDF: Term Frequency-Inverse Document Frequency is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. Similar to

BoW, we used a TfidfVectorizer with max_features=5000.
3. GloVe: Global Vectors for Word Representation is a pre-trained word embedding model that captures semantic relationships between words. We utilized the `glove-wiki-gigaword-50` embeddings with a dimension of 50.
4. Skip-gram: We trained our own Skip-gram model using Gensim's Word2Vec on our training corpus. The model was configured with a vector size of 100, a window of 5, and min_count of 2.

D. Classification Models

We implemented a wide range of classification models, which are categorized into Machine Learning (ML) models and Neural Network (NN) models.

1. **Machine Learning Models:**
   - Random Forest: An ensemble learning method that operates by constructing a multitude of decision trees.
   - Logistic Regression: A linear model used for binary and multiclass classification.
   - Naive Bayes: A probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features.
2. **Neural Network Models:**
   - Deep Neural Network (DNN): A feedforward neural network with multiple hidden layers.
   - SimpleRNN: A basic recurrent neural network suitable for processing sequential data.
   - GRU (Gated Recurrent Unit): An advanced RNN architecture with gating mechanisms to control information flow.
   - LSTM (Long Short-Term Memory): A sophisticated RNN designed to handle long-range dependencies in data.
   - Bidirectional Variants: We also implemented bidirectional versions of SimpleRNN, GRU, and LSTM to capture context from both forward and backward directions in the sequence.

All neural network models were built using TensorFlow and Keras. They included an embedding layer, the respective recurrent layers, and a dense output layer with a softmax activation function for multi-class classification. We used the Adam optimizer and sparse categorical cross-entropy as the loss function. Early stopping was employed to prevent overfitting.

III. RESULTS

We conducted a total of 22 experiments, combining the different word representation techniques with the various classification models. The performance of each model was evaluated based on Accuracy, Precision, Recall, and F1-score. The comprehensive results are presented in Table I.

A. Machine Learning Models

The ML models were trained using BoW and TF-IDF features. Logistic Regression consistently outperformed Random Forest and Naive Bayes, with the TF-IDF representation yielding slightly better results than BoW.

B. Neural Network Models

The DNN was trained on BoW and TF-IDF features, showing performance comparable to the best ML models. The RNN-based models were trained on GloVe and Skip-gram embeddings. The bidirectional models, particularly Bidirectional LSTM and GRU, demonstrated superior performance, with GloVe embeddings generally providing better results than our custom-trained Skip-gram model.

Table I. Comprehensive Model Performance

Feature Type	Model	Accuracy	F1-Score
BoW	Random Forest	0.5305	0.5321
	Logistic Regression	0.6396	0.6369
	Naive Bayes	0.6537	0.6515
	DNN	0.6752	0.6548
TF-IDF	Random Forest	0.5317	0.5323
	Logistic Regression	0.6772	0.6752
	Naive Bayes	0.6615	0.6588
	DNN	0.6896	0.6741
GloVe	DNN	0.6345	0.6282
	SimpleRNN	0.2531	0.1594
	GRU	0.6893	0.6838
	LSTM	0.6863	0.6804
	Bidirectional SimpleRNN	0.381	0.3276
	Bidirectional GRU	0.7134	0.7058
	Bidirectional LSTM	0.7067	0.6987
Skip-gram	DNN	0.627	0.6194
	SimpleRNN	0.1995	0.153
	GRU	0.6673	0.6598
	LSTM	0.6765	0.6712
	Bidirectional SimpleRNN	0.347	0.3176
	Bidirectional GRU	0.6947	0.6929
	Bidirectional LSTM	0.6951	0.6934

IV. CONCLUSION

This project successfully implemented and evaluated a wide array of machine learning and neural network models for the task of QA text classification. Our findings indicate that:

- Word Embeddings are Superior: Models using pre-trained GloVe embeddings significantly outperformed those using traditional BoW and TF-IDF features, demonstrating the importance of semantic representations for this task.
- Deep Learning Models Excel: Neural network models, especially the recurrent architectures (LSTM and GRU), consistently achieved higher accuracies than the classical ML models.
- Bidirectionality is Key: The bidirectional variants of the RNN models provided a notable performance boost, with the Bidirectional GRU with GloVe embeddings emerging as the top-performing model with an accuracy of **71.32%**. This highlights the benefit of capturing context from both past and future words in a sequence.

Future work could involve exploring more advanced models like Transformers (e.g., BERT) and experimenting with larger, more diverse datasets to further improve classification performance.

REFERENCES

[1] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014

[2] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in *Advances in Neural Information Processing Systems 26*, 2013

[3] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9

[4] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014