Rule-based and Traditional Machine Learning Approaches to Question Answering using the Stanford Question Answering Dataset (SQuAD)

Abstract

This project aims to explore rule-based and traditional machine learning techniques for question-answering tasks, using the Stanford Question Answering Dataset (SQuAD). Unlike common deep learning approaches, this study focuses on the application of conventional NLP methods and machine learning models such as Random Forests, Gradient Boosting Machines, and SVMs. The goal is to evaluate the efficacy of these methods in comparison to more resource-intensive deep learning solutions, examining their potential advantages and limitations in the context of question answering.

1.0 Introduction

The evolution of question-answering systems is a key aspect of progress in natural language processing (NLP). Amidst the prevalent use of deep learning, this project revisits and emphasizes the relevance of traditional NLP and machine learning techniques. By applying these established methods to the Stanford Question Answering Dataset (SQuAD), the project seeks to evaluate their effectiveness in accurately deciphering and responding to questions. This study is not only an examination of their performance but also a comparison to the more commonly used deep learning models, offering a comprehensive perspective on the diverse methodologies in NLP.

2.0 Related Work

The landscape of question-answering (QA) systems has been profoundly influenced by various approaches spanning from traditional algorithms to advanced deep learning techniques. Our study situates itself amidst this diverse array of methodologies.

1. **Deep Learning Approaches**: The advent of deep learning models like BERT, introduced by Devlin et al. in 2018, marked a significant shift in QA systems. These models, leveraging transformer architectures, have demonstrated exceptional capabilities in understanding context and language nuances. Further advancements were seen with models such as GPT (Generative Pretrained Transformer) and ELMo (Embeddings from Language Models), which underscored the power of contextual embeddings in NLP tasks.

2. **Traditional Machine Learning Models**: Prior to the deep learning era, QA systems relied heavily on models such as Support Vector Machines (SVMs) and Random Forests. These models were used for feature-based classification of questions and identification of potential answer spans. For instance, research by Moschitti et al. (2007) employed SVMs for question classification, a fundamental step in QA systems.

3. **Rule-based Systems**: Early QA systems, like the ones developed during the TREC QA tracks in the late 1990s and early 2000s, were predominantly rule-based. These systems

utilized hand-crafted rules and heuristics to parse questions and retrieve answers from structured databases or text corpora.

4. **Hybrid Approaches**: There have been efforts to combine traditional and modern techniques to leverage the strengths of both. For example, the work by Cui et al. (2017) on Attention-over-Attention Neural Networks for reading comprehension combines neural attention mechanisms with traditional NLP features.

5. **Ensemble Methods**: Ensemble methods, which combine predictions from multiple models, have also been explored in QA. These methods often blend the robustness of machine learning models with the nuanced understanding of deep learning, aiming to improve overall system performance.

6. **Semantic Parsing and Retrieval-based Models**: Before the dominance of neural networks, semantic parsing and retrieval-based models were prominent. They focused on converting natural language questions into logical forms and retrieving answers from knowledge bases or textual data.

In our study, we revisit and re-evaluate the effectiveness of traditional machine learning and rule-based approaches in the context of the SQuAD dataset. By comparing these approaches with modern deep learning techniques, we aim to provide a nuanced understanding of their applicability and performance in current QA systems.

3.0 Dataset Description

The dataset used is the SQuAD 1.1, chosen for its straightforward question-answer format and the absence of unanswerable questions. It comprises over 100,000 question-answer pairs from more than 500 Wikipedia articles. The dataset is a standard for evaluating reading comprehension in NLP. You can find the dataset at the link below:
https://www.kaggle.com/datasets/stanfordu/stanford-question-answering-dataset

4.0 Methodology

1. **Data Acquisition and Transformation**: The project starts with uploading and loading the SQuAD 1.1 dataset into a Google Colab environment. The JSON structure of the dataset is analyzed to understand its hierarchy, including keys like data, paragraphs, questions, and answers.

2. **Data Normalization**: The dataset is normalized into a flat structure, making it easier to work with. This involves extracting relevant information such as the title, context, question, answer, and answer start position from the dataset.

3. **Exploratory Data Analysis (EDA)**: EDA is performed to understand the dataset better. This includes analyzing the number of articles, paragraphs, questions, answers, and

calculating average lengths of contexts, questions, and answers. The presence of unanswerable questions is also noted.

4. **Data Cleaning**: The raw data is cleaned to remove any unnecessary or redundant information.

5. **Data Pre-Processing**: The text data (context and questions) undergo pre-processing, which includes tokenization, lemmatization, and removal of stop words and punctuation. This step is essential to prepare the data for feature extraction and modeling.

6. **Feature Engineering**: Using TF-IDF Vectorization, features are extracted from the preprocessed text data. This is a critical step in transforming textual data into a format suitable for machine learning models.

7. **Model Training And Testing**: A Support Vector Machine (SVM) with a linear kernel is trained to classify whether an answer is present in a given context. The dataset is split into training and testing sets for this purpose. The model was later used to perform predictions of the test data.

8. **Answer Extraction Function**: A function is developed to extract answers from a given context using the trained SVM model. This function utilizes sentence tokenization and the TF-IDF vectorized data to predict the presence of answers in segments of the context.

5.0 Experimental Discussion

Model Evaluation: Initially, the data was split into train and test using a ratio of 80:20 respectively. After training, we evaluated the model using metrics such as precision, recall, and F1. The SVM model's performance is evaluated using precision, recall, and F1-score metrics. The model achieved a precision of 0.55 for class 0 (absence of answer) and 0.71 for class 1 (presence of answer), with respective recalls of 0.32 and 0.86. The F1 scores were 0.41 for class 0 and 0.78 for class 1. Overall, the model exhibited an accuracy of 68%, with a weighted average F1-score of 0.65. These results indicate a better performance in identifying the presence of an answer in the context as compared to identifying its absence.

Furthermore, it is noteworthy to compare our results with a strong baseline established by Stanford University. Stanford produced a logistic regression model for the SQuAD dataset, which achieved an F1 score of 51%. This comparison highlights the enhanced performance of our SVM model against a well-established baseline in the field.
See below table summarizing the results:

	Precision	Recall	F1-score	Support
0	0.55	0.32	0.41	6036
1	0.71	0.86	0.78	11484
Accuracy			0.68	17520
Macro avg	0.63	0.59	0.59	17520
Weighted avg	0.65	0.68	0.65	17520

Application of the Model – Example Case: The practical application of the model is illustrated with an example. When the question, "What was the primary purpose of the Great Wall of China?" was posed, the model predicted this entire context passage as the answer: "The Great Wall of China is a series of fortifications made of stone, brick, tamped earth, wood, and other materials, generally built along an east-to-west line across the historical northern borders of China." This indicates that the model was successful in identifying relevant context containing the answer, but may need further refinement to pinpoint the exact answer span within the context.

Analysis of Results: The results demonstrate the model's capability in question answering tasks using traditional machine learning approaches. However, the tendency to predict larger segments of text as answers suggests a need for further optimization. This could involve enhancing the feature extraction process or refining the model to improve its precision in identifying exact answer spans.

Future Work: Based on these findings, future work could focus on improving the answer extraction accuracy. This might include experimenting with different feature extraction techniques, implementing more complex algorithms for answer span prediction, or integrating rule-based methods to narrow down the answer segments.

6.0 Contribution

The successful completion of this project was a result of the collaborative effort of four team members, each contributing significantly to different aspects of the study.

1. Haritha Deevi: Data Acquisition and Preprocessing

- Responsible for acquiring the Stanford Question Answering Dataset (SQuAD) and preparing it for analysis.
- Conducted initial data exploration to understand the dataset structure and its key components.

- Implemented data normalization processes to transform the dataset into a usable format for subsequent analysis.

2. Sanjith Sivapuram: Exploratory Data Analysis (EDA) and Feature Engineering

- Led the exploratory data analysis to gain insights into the dataset, including the analysis of article lengths, question types, and answer characteristics.
- Developed and implemented feature engineering techniques, including TF-IDF vectorization, to extract relevant features from the text data for model training.

3. Gokulnath Anand: Model Development and Training

- Focused on the development and training of the SVM model.
- Managed the splitting of the dataset into training and testing sets and tuned the model parameters to optimize performance.
- Conducted the initial model evaluation, analyzing metrics such as precision, recall, and F1-score.

4. Gbenga Ladapo: Model Evaluation, Optimization, and Documentation

- Took charge of the comprehensive evaluation of the model, including a detailed analysis of the model's performance and comparison with the Stanford logistic regression baseline.
- Created and tested an alternative random forest machine learning model for comparison
- Led efforts to refine and optimize the SVM model based on initial testing, implementing adjustments to improve accuracy and efficiency.
- Responsible for compiling and documenting the research findings, ensuring clarity and thoroughness in the final report.

Each member's distinct contributions were integral to the project's success, demonstrating effective teamwork and specialized skills in different areas of NLP and machine learning.

7.0 Conclusion

This project embarked on an exploratory journey to evaluate the efficacy of traditional machine learning techniques, specifically using an SVM model, in the context of question answering with the Stanford Question Answering Dataset (SQuAD). Our findings underscore several key insights and implications for the field of Natural Language Processing (NLP).

1. **Performance of Traditional Methods**: The SVM model demonstrated a commendable performance, with an accuracy of 68% and a weighted average F1-score of 0.65. These metrics, especially the F1-score of 0.78 for class 1 (presence of answer), indicate a strong capability in identifying relevant contexts containing answers. However, the model

showed limitations in precisely pinpointing the exact span of answers within larger text segments.

2. **Comparison with Deep Learning Baselines**: When compared to Stanford's logistic regression baseline, which achieved an F1 score of 51.0%, our model showed a significant improvement in performance. This comparison highlights the potential of traditional machine learning methods in certain NLP tasks, offering a viable alternative to more resource-intensive deep learning models.

3. **Future Directions**: The results point towards several avenues for future research. Enhancing the model's ability to accurately extract specific answer spans, possibly through integration with rule-based methods or advanced feature engineering, presents a promising area of development. Additionally, experimenting with hybrid models that combine traditional and deep learning techniques could yield further improvements in performance.

4. **Broader Implications**: This study contributes to the ongoing discourse in the NLP community about the viability of various methodologies in question answering tasks. It underscores the importance of considering a range of approaches, from traditional machine learning to cutting-edge deep learning, depending on the specific requirements and constraints of the task at hand.

In conclusion, our research reaffirms the value of traditional machine learning techniques in the realm of NLP, while also highlighting areas for enhancement. As the field continues to evolve, such explorations are vital in advancing our understanding and capabilities in building effective and efficient question answering systems.

References

1. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805.

2. Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. arXiv:1606.05250.

3. Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. European Conference on Machine Learning.

4. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.

5. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. arXiv:1802.05365.

6. Moschitti, A., Quarteroni, S., Basili, R., & Manandhar, S. (2007). Exploiting Syntactic and Shallow Semantic Kernels for Question Answer Classification. ACL.

7. Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., & Hu, G. (2017). Attention-over-Attention Neural Networks for Reading Comprehension. ACL.

8. Voorhees, E. M., & Tice, D. M. (2000). Building a question answering test collection. Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval.

9. Freund, Y., Iyer, R., Schapire, R. E., & Singer, Y. (2003). An Efficient Boosting Algorithm for Combining Preferences. The Journal of Machine Learning Research, 4.

10. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.

11. Salton, G., & McGill, M. J. (1986). Introduction to Modern Information Retrieval. McGraw-Hill