Question-Answer Validation Analysis

1. Introduction

The journey of this experiment began with a fundamental question: how can we effectively validate the truthfulness of question-answer pairs using modern machine learning and deep learning techniques? This exploration was not just about achieving the highest accuracy, but about understanding the strengths and limitations of different approaches in natural language processing. The BoolQ dataset provided an ideal testing ground, offering a rich collection of boolean questions and answers that challenged our models to understand both the semantic meaning and logical validity of question-answer pairs.

2. Dataset Exploration and Transformation

The BoolQ dataset was originally designed for question answering, where the task is to determine whether a given answer is true or false. We transformed this into a validation task by generating both correct and incorrect question-answer pairs for each example. This effectively doubled the dataset size while creating a more nuanced version of the original task.

```
Original data point:
{'question': 'do iran and afghanistan speak the same language', 'answer': True}

Transformed examples:
{'text': "The question is 'do iran and afghanistan speak the same language' and the answer is 'True'", 'label': 'correct'}
{'text': "The question is 'do iran and afghanistan speak the same language' and the answer is 'False'", 'label': 'incorrect'}
```

For this particular dataset, both the original and the transformed tasks can be framed as equivalent: the model must decide whether a question-answer pair is valid. However, when the evaluation goes beyond a simple binary judgment—such as in settings that require justifying the answer, ranking alternatives, or identifying inconsistencies—the validation task becomes fundamentally different in nature. It demands a deeper understanding and more sophisticated reasoning capabilities.

This change was crucial because it altered the core objective of the task. In the original BoolQ format, models were required to assess the truth value of a statement. In our validation format, they must determine whether a proposed answer is a coherent and appropriate response to a specific question. This subtle but significant shift aligns more closely with real-world validation scenarios.

3. The Machine Learning Journey

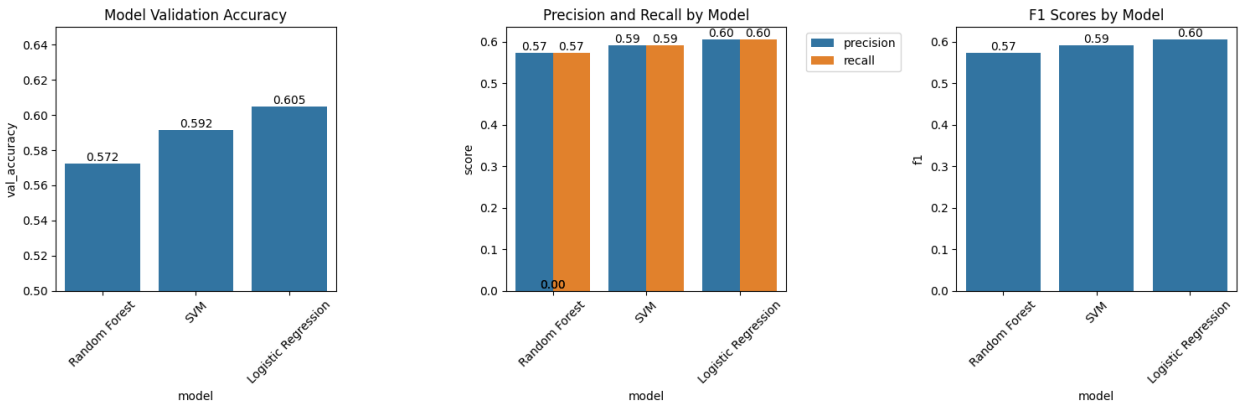
Our exploration of machine learning approaches took two distinct paths, each with its own preprocessing and encoding strategy. This dual approach allowed us to compare traditional word embeddings with modern contextual embeddings, revealing important insights about text representation in natural language processing.

Word2Vec Embeddings

The first approach utilized Word2Vec embeddings, a traditional but powerful method for text representation. The preprocessing pipeline for this approach involved several key steps:

1. Text cleaning: Removing punctuation and converting to lowercase.
2. Tokenization: Breaking text into individual words
3. Stop word removal: Eliminating common words that add little semantic value
4. Word2Vec embedding: Converting each word into a 100-dimensional vector
5. Document representation: Creating a single vector for each question-answer pair by averaging the word vectors

After preprocessing, a grid search over hyperparameters was conducted on three classical machine learning models—Random Forest, Support Vector Machine (SVM), and Logistic Regression—to evaluate the learned representations.



The results of 0.6 accuracy are relatively low, given that the baseline is 0.5 accuracy. However, they are still higher than one might expect, let's reflect on the nature of the task we are trying to solve with this approach.

Task Reflection

In essence, this task is a general knowledge evaluation. To determine whether a question–answer pair is correct, one would typically need access to factual knowledge. While this kind of evaluation is common for general pretrained models, which have been exposed to large corpora, it's more surprising for models trained only on limited labeled datasets.

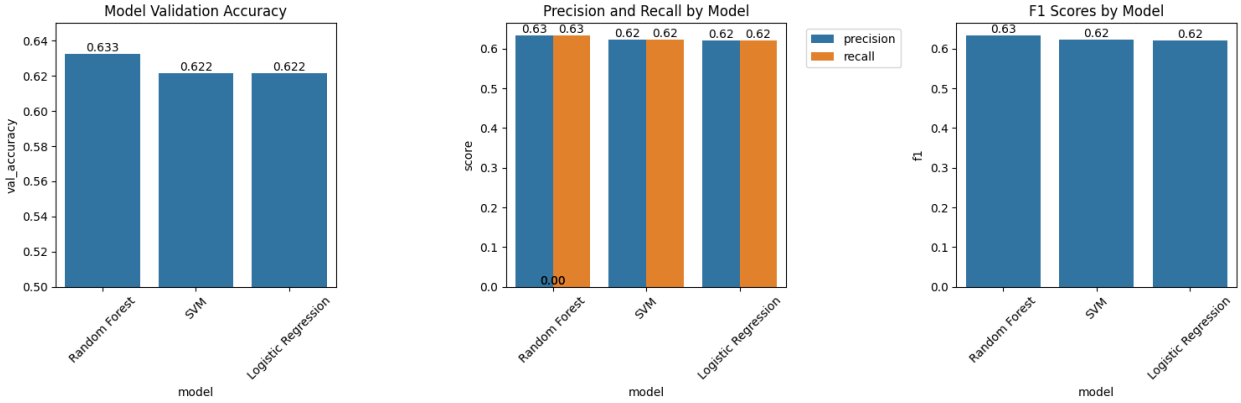
What makes this setup intriguing is that models still perform better than random guessing on factual correctness judgments, even without explicit external knowledge. There are a couple of strong hypotheses for why this might happen:

1. *Implicit Knowledge in Word Embeddings*: Pretrained embeddings like Word2Vec can encode rich semantic relationships based on co-occurrence statistics in large corpora. This allows models to make informed judgments based on the geometry of the embeddings, even without direct supervision on the facts.
2. *Data Leakage or Memorization*: The dataset may inadvertently allow information to leak from the training to the validation/test splits. If similar questions, phrasings, or facts are repeated, the model may learn to associate certain patterns with correctness labels.

While it's sometimes suggested that models might exploit superficial statistical cues (e.g., answers that "sound more right"), the specific format of this task—where both the correct and incorrect answers to a question are given in similar wording—makes this unlikely. The model isn't deciding between alternatives based on plausibility; it's classifying whether a specific question–answer pair is factually correct, which requires deeper reasoning or memorization.

BERT's Contextual Embeddings

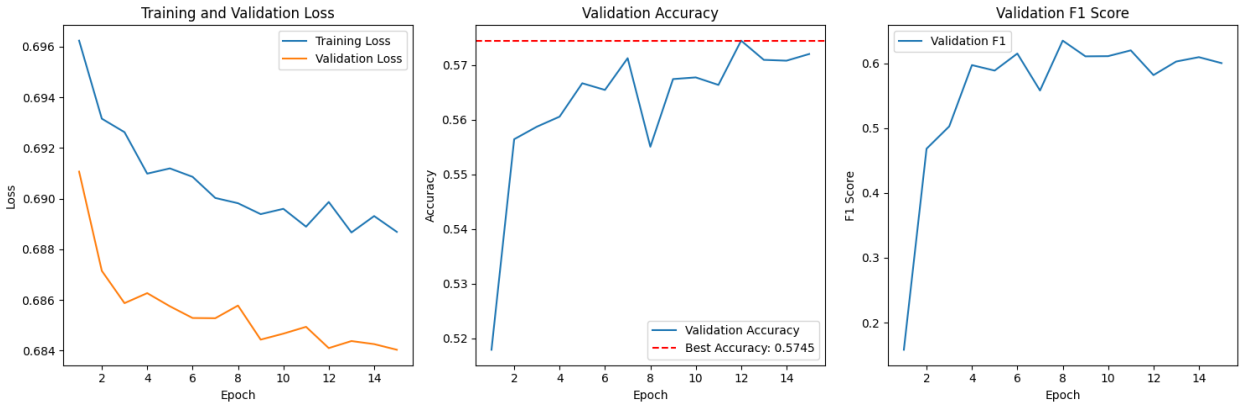
The second approach leveraged BERT's contextual embeddings, representing a more modern and sophisticated method. BERT'S autoencoder handled the preprocessing directly. The representations of the data were obtained, and then the same ML models mentioned before were used.



We observed slightly better performance using the BERT tokenizer compared to the Word2Vec tokenizer, suggesting that BERT is more effective at capturing the semantic meaning of text and benefits from a broader base of embedded general knowledge.

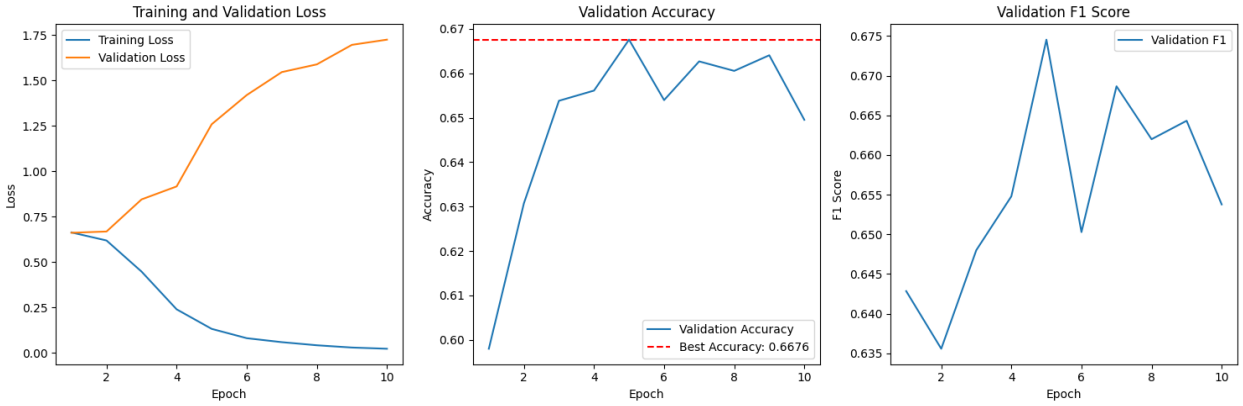
4. Deep Learning Exploration

For the deep learning exploration, the preprocessing step was similar to that described above, using the BERT tokenizer in both cases. We tested two different approaches. In the first, we used the BERT model without fine-tuning—only leveraging its final hidden states as input to a neural classifier head. This approach mirrors the classical machine learning setup with BERT tokenization, but replaces the ML model with a neural classifier.



This explains why the results were similar, though slightly worse (0.58): the neural head alone may not have been as powerful as the ML models, or the number of training epochs may have been insufficient for optimal performance.

In the second approach, we fine-tuned the entire BERT model on the specific task. This setup yielded the highest accuracy among all experiments (0.67). Early stopping was applied after only a few epochs, as shown in the chart on the left, due to early signs of overfitting. Fine-tuning gave the model more degrees of freedom to adapt to the data, resulting in improved performance.



However, the improvement over classical ML models using BERT tokenization was modest. This suggests that, while these models can capture aspects of semantic meaning, they either lack or failed to acquire the level of "general knowledge" needed to reliably assess whether a question has been properly answered.

5. Conclusions

In this study, we evaluated several approaches for determining whether a passage correctly answers a given question. Traditional machine learning models using Word2Vec embeddings showed moderate performance but struggled with deeper semantic understanding. Replacing Word2Vec with BERT tokenization improved results, highlighting the advantage of contextual embeddings. Using fixed BERT embeddings with a neural classifier yielded comparable performance to classical ML models, while fine-tuning the entire BERT model achieved the best results. However, the performance gain was relatively modest, suggesting that even powerful models like BERT face limitations when tasked with nuanced validation.

These findings highlight the inherent difficulty of the task, which requires not only semantic understanding but also a degree of general and world knowledge. Current models, even when fine-tuned, lack the comprehensive background understanding needed to reason effectively about the correctness of answers in complex scenarios. While text embeddings can capture linguistic meaning, they fall short in modeling the broader context necessary for truth evaluation. Future work could explore the use of larger language models, which may be better equipped to handle the reasoning demands of this task.