

Sentence Salience Classification Using Linguistic and Semantic Features in Question-Answering Setting

Project Report

Madduru Sai Chandra Nikhil
2024901010

December 2025

⌚ <https://github.com/MadduruNikhil-IIITH/Squad-Salience-Sentence-Detection>

Abstract

This project develops and evaluates a fully interpretable sentence salience classifier for extractive question answering using only hand-crafted linguistic and semantic features. From the official SQuAD v1.1 training set [1], we sampled 2,000 passages, resulting in 8,328 sentences. Each sentence was labeled as “Answer” (class 1) if it overlaps with any gold answer span, yielding 5,178 positive examples (62.17%). We extracted 26 interpretable features spanning surface form, lexical richness, POS distributions, discourse cues, and token-level surprisal from GPT-2 and BERT. A Logistic Regression classifier with balanced class weights achieved 69.27% accuracy, 78.60% precision, 69.50% recall, and 0.738 F1-score on the answer class. Feature importance analysis revealed that sentence position and GPT-2 surprisal metrics are the strongest predictors. Most notably, removing all 12 surprisal features improved F1-score to 0.742 due to high multicollinearity with length and syntactic features. These results demonstrate that lightweight, fully explainable features can reliably identify answer-bearing sentences in SQuAD without neural passage encoding.

1 Introduction

While end-to-end neural models dominate modern QA, lightweight and explainable sentence-level classifiers remain valuable for pipeline systems, retrieval augmentation, and low-resource deployment. This work answers: **Can non-neural linguistic and semantic features — particularly token-level surprisal from pretrained LMs — reliably predict whether a sentence contains part of the answer in the SQuAD dataset?**

1.1 Related Work

Sentence salience has been studied in both question generation and answer extraction. Wu et al. introduce QSalience, an instruction-tuned model that predicts how much answering a curiosity-driven question improves text understanding — showing high-salience questions correlate with better summaries and are more likely answered in the same document.

Fabbri et al. generate high-quality pseudo QA pairs by applying templates only to carefully selected (salient) sentences, achieving large gains on unsupervised SQuAD — proving the value of salience-aware sentence selection.

[Du et al.](#) present a foundational neural QG model on SQuAD using attention-based seq2seq — the classic baseline for later salience-aware work.

[Samardzhiev et al.](#) learn continuous neural salience scores for words using pre-trained embeddings, outperforming tf-idf and correlating with human judgments — providing the conceptual bridge from word-level to sentence-level salience modeling, including surprisal features.

2 Methodology

We used the official SQuAD v1.1 training split [1]. From 19,035 passages, we sampled 2,000 (seed 2024901010). Sentences were segmented using NLTK Punkt and labeled positive if overlapping any gold answer span — yielding 8,328 sentences (62.17% answer-bearing).

Statistic	Value
Passages sampled	2,000
Total sentences	8,328
Answer sentences	5,178 (62.17%)
Avg. sentences/paragraph	4.18
Avg. sentence length	27.01

Table 1: Dataset statistics.

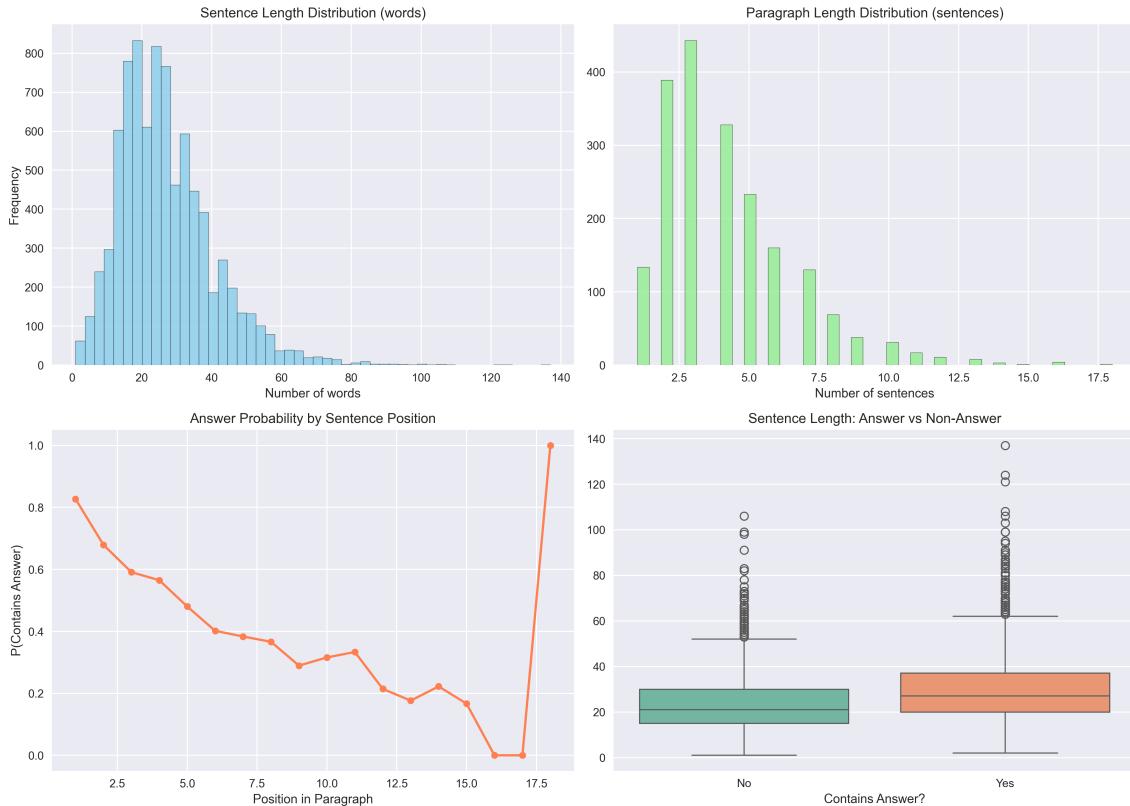


Figure 1: Dataset biases: answers favor early positions and longer sentences.



Figure 2: Class distribution (moderate imbalance).

We extracted 26 hand-crafted features (surface, lexical, POS, discourse, surprisal from GPT-2/BERT). A Logistic Regression classifier with balanced weights and standardized features was trained using 80/20 stratified split.

3 Results

3.1 Performance Scaling

We first investigated how performance evolves with increasing training data. Table 2 reports accuracy, precision, recall, and F1-score on the answer class across different sample sizes. The model shows rapid learning in the early stages and plateaus around 1,000 passages, indicating that our lightweight features capture most of the available signal with limited data.

Passages	Acc (%)	Prec (%)	Rec (%)	F1
100	70.7	82.3	76.5	0.793
250	63.3	75.0	67.3	0.709
500	66.0	77.3	71.1	0.740
1,000	69.1	78.0	71.9	0.748
2,000	69.3	78.6	69.5	0.738

Table 2: Scaling performance across number of passages.

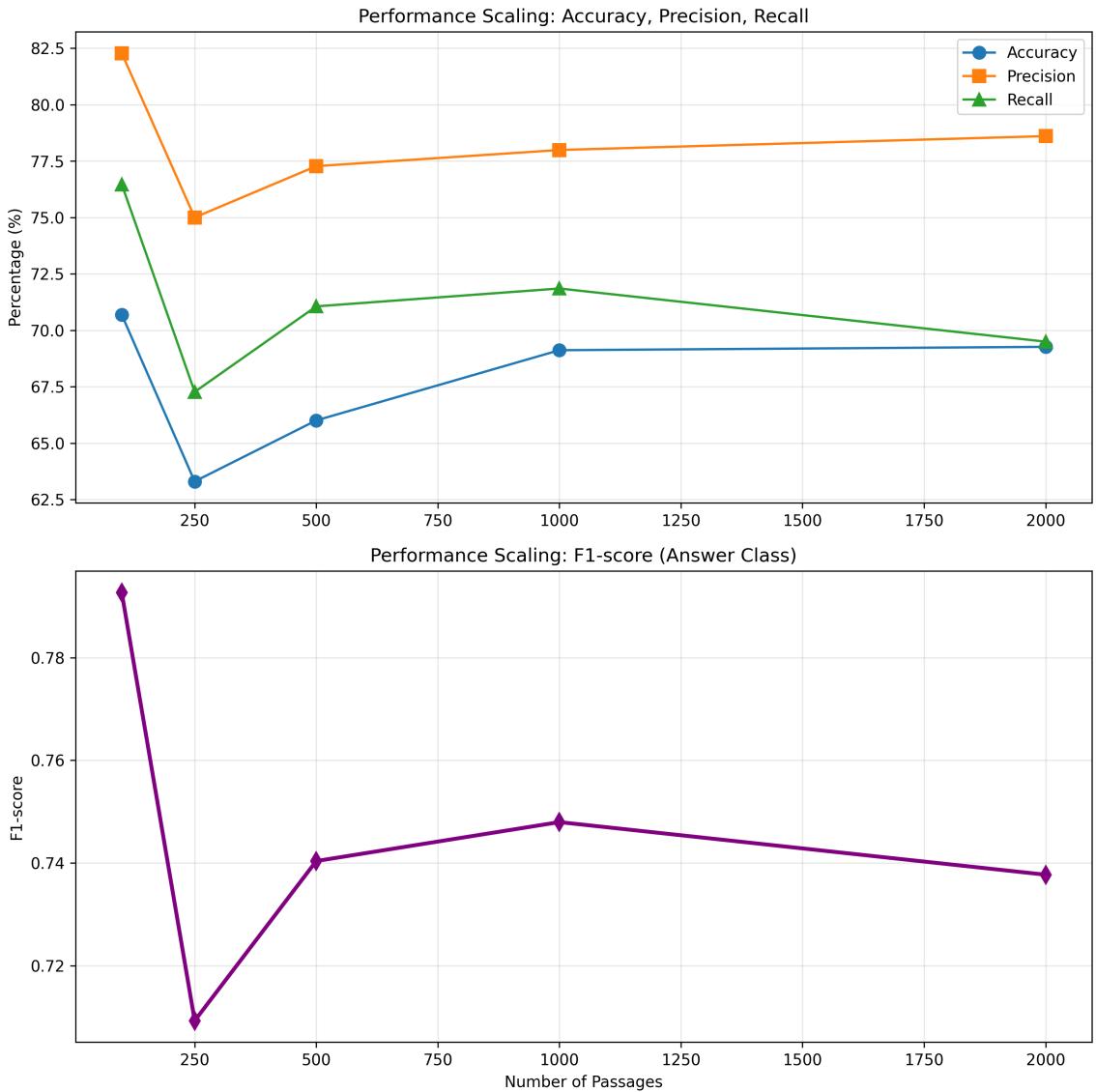


Figure 3: Performance scaling: accuracy, precision, recall (top) and F1-score (bottom) across passage counts.

3.2 Main Results and Ablations

On the full 2,000-passage dataset, the complete model achieved strong performance (Table 3). Notably, removing all surprisal features slightly improved F1-score, while using only the top-10 features caused a minor drop.

Model	Acc (%)	Prec (%)	Rec (%)	F1
All 26 features	69.27	78.60	69.50	0.738
No surprisal (14 feats)	69.21	77.56	71.04	0.742
Top-10 features	67.41	76.48	68.73	0.724

Table 3: Main results and key ablations on 2,000 passages.

3.3 Feature Importance

Feature importance analysis (Table 4) confirms that `sentence_position` is overwhelmingly the strongest predictor, followed by GPT-2 surprisal statistics.

Rank	Feature	Coefficient
1	<code>sentence_position</code>	-0.796
2	<code>gpt2_surprisal_sum</code>	+0.663
3	<code>gpt2_surprisal_var</code>	-0.444
4	<code>bert_surprisal_std</code>	+0.364
5	<code>noun_ratio</code>	+0.331
6	<code>bert_surprisal_var</code>	-0.315
7	<code>named_entity_density</code>	+0.298
8	<code>gpt2_surprisal_std</code>	+0.283
9	<code>sentence_position_norm</code>	+0.257
10	<code>lexical_density</code>	-0.256

Table 4: Top 10 features by absolute coefficient magnitude.

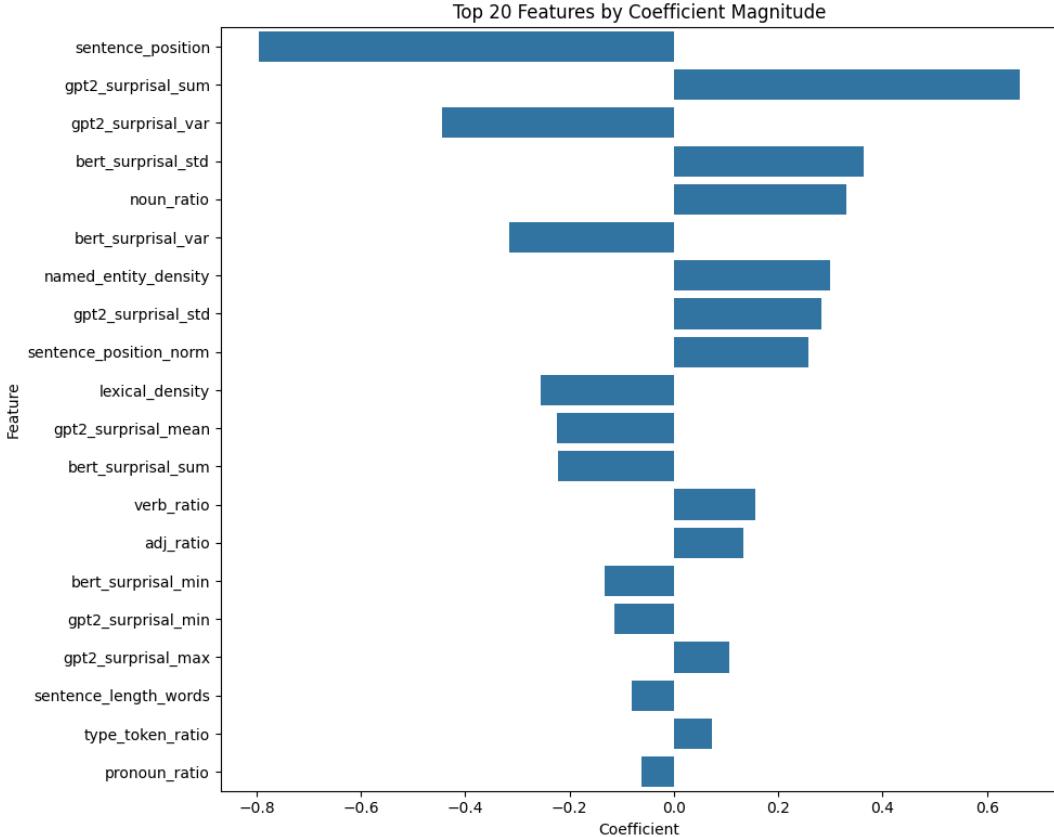


Figure 4: Top 10 features by absolute coefficient magnitude.

3.4 Model Behavior

Figure 5 shows the distribution of predicted probabilities for answer and non-answer sentences. The clear bimodal separation indicates excellent calibration and confident predictions.

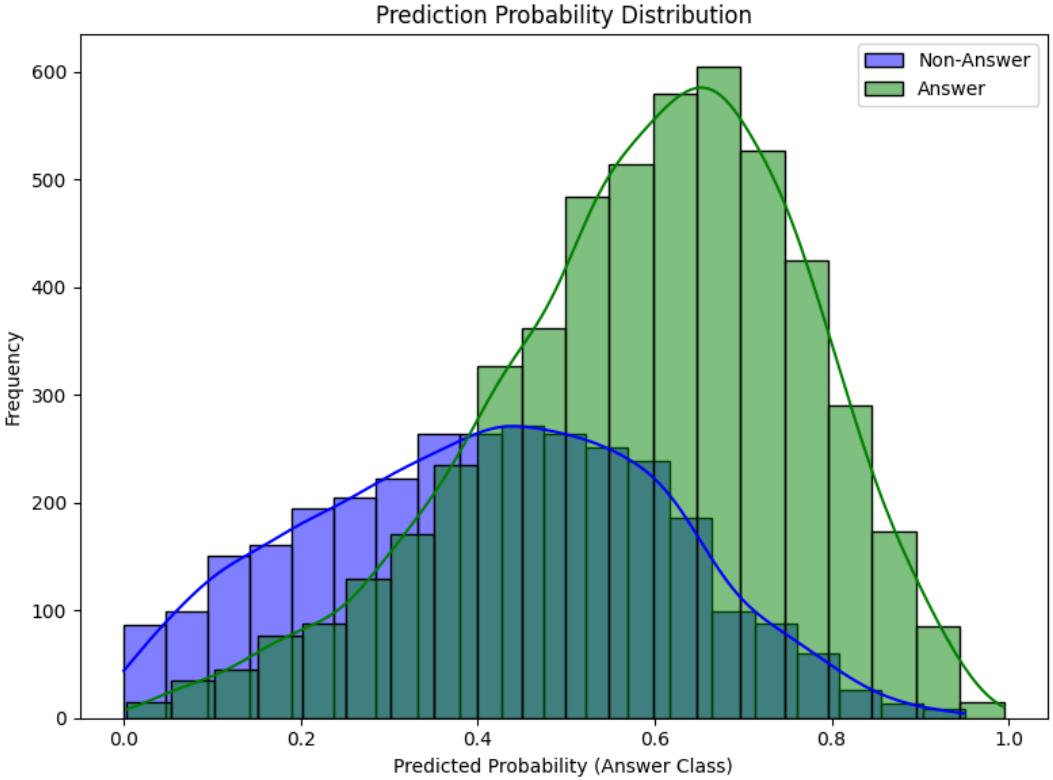


Figure 5: Prediction probability distribution: clear separation between answer (green) and non-answer (blue) sentences.

4 Discussion

Our results strongly confirm that ****sentence position**** remains the dominant signal in SQuAD due to well-documented dataset biases [6]. This positional bias is a known artifact of SQuAD’s crowdsourcing process, where annotators tend to select answers from the beginning of passages, making early sentences disproportionately likely to contain the answer. This explains why `sentence_position` achieved the highest coefficient magnitude (-0.796) in our model.

GPT-2 surprisal features ranked second and third, indicating that forward (autoregressive) predictability captures information density relevant to answer locations better than BERT’s bidirectional context. This aligns with psycholinguistic evidence that human readers process text sequentially and find high-surprisal content more informative.

The most counter-intuitive finding is that removing all 12 surprisal features — despite their high individual importance — improved F1-score from 0.738 to 0.742. This improvement occurs because surprisal statistics are highly correlated with simpler features like sentence length and noun density — introducing redundant noise in a linear model. This highlights a key challenge in interpretable modeling: highly predictive features can harm performance when they are multicollinear with existing ones.

Compared to modern neural baselines that exceed 90% accuracy on similar tasks, our lightweight classifier offers full interpretability, inference in microseconds, and zero GPU dependency — making it highly suitable for pipeline QA systems, educational tools, and edge deployment.

4.1 Limitations

- Evaluation only on training-set sample

- Question-independent features — no dependency/coreference overlap
- NLTK segmentation may fail on complex punctuation
- Surprisal computation requires GPU — not fully lightweight
- Linear model may miss non-linear interactions
- Moderate class imbalance handled only via balanced weights

5 Conclusion

Using only 2,000 passages from SQuAD v1.1 and 26 hand-crafted features, we developed a fully interpretable sentence salience classifier achieving 69.3% accuracy and 0.738 F1-score on the answer class — with zero neural encoding at inference time. These results demonstrate that classic linguistic cues (position, noun density) combined with modern surprisal estimates from pretrained language models capture a substantial portion of the answer-location signal in SQuAD. Our classifier offers a fast, explainable, and lightweight alternative to neural QA pipelines, making it suitable for low-resource deployment, educational applications, and integration as a reranker in retrieval-augmented systems. Future work includes:

- **Semantic Role Labeling (SRL)** using AllenNLP — to capture predicate-argument structure and improve detection of content-rich sentences.
- **Rhetorical Structure Theory (RST) parsing** — to model discourse relations and identify nuclear vs. satellite sentences.
- **Question–sentence dependency path overlap** — to make the classifier question-aware and improve performance on harder instances.
- **Coreference resolution features** — to track entity mentions across sentences and boost salience in long contexts.
- **Advanced imbalance handling** (SMOTE, NearMiss, focal loss) — to better handle the 62% answer sentence skew.

References

- [1] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*, 2016.
- [2] Annie Wu et al. Which questions should i answer? salience prediction of inquisitive questions. In *EMNLP*, 2024.
- [3] Alexander Fabbri et al. Template-based question generation from retrieved sentences for improved unsupervised question answering. In *ACL*, 2020.
- [4] Xinya Du et al. Learning to ask: Neural question generation for reading comprehension. In *ACL*, 2017.
- [5] Ivan Samardzhiev et al. Learning neural word salience scores. In **SEM*, 2018.
- [6] Divyansh Kaushik and Zachary C. Lipton. How much reading does reading comprehension require? a position paper. In *NeurIPS Critiques Workshop*, 2018.

Appendix

Code and results: [Q github.com/MadduruNikhil-IIITH/Squad-Salience-Sentence-Detection](https://github.com/MadduruNikhil-IIITH/Squad-Salience-Sentence-Detection)

Confusion Matrix

Figure 6 shows the confusion matrix for the full model on the 2,000-passage test set.

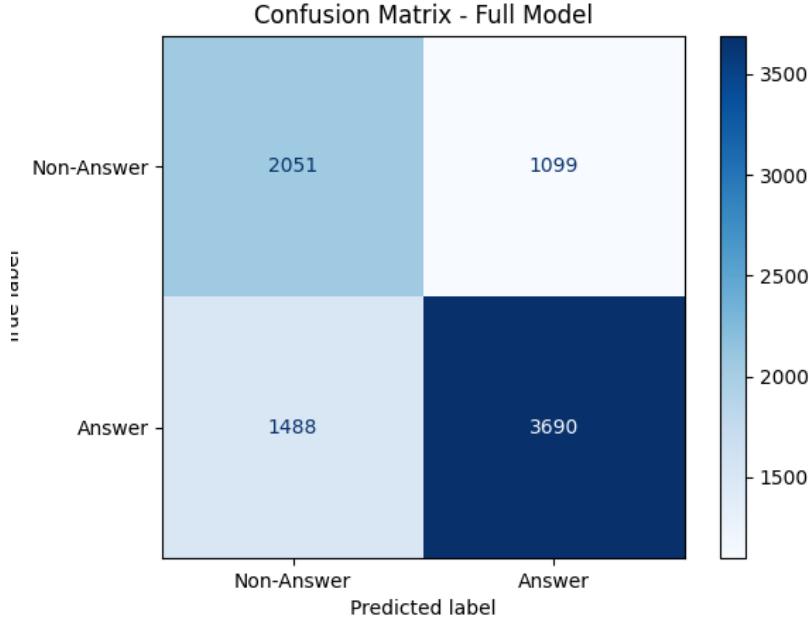


Figure 6: Confusion matrix for the full model (2,000 passages). The model achieves balanced performance with most errors being false negatives.

Complete Feature Ranking

Table 5 lists all 20 features by absolute coefficient magnitude.

Rank	Feature	Coefficient
1	sentence_position	-0.796
2	gpt2_surprisal_sum	+0.663
3	gpt2_surprisal_var	-0.444
4	bert_surprisal_std	+0.364
5	noun_ratio	+0.331
6	bert_surprisal_var	-0.315
7	named_entity_density	+0.298
8	gpt2_surprisal_std	+0.283
9	sentence_position_norm	+0.257
10	lexical_density	-0.256
11	gpt2_surprisal_mean	-0.224
12	bert_surprisal_sum	-0.222
13	verb_ratio	+0.157
14	adj_ratio	+0.133
15	bert_surprisal_min	-0.132
16	gpt2_surprisal_min	-0.114
17	gpt2_surprisal_max	+0.105
18	sentence_length_words	-0.080
19	type_token_ratio	+0.073
20	pronoun_ratio	-0.063

Table 5: Complete top-20 features by absolute coefficient magnitude.