Natural Language Processing Report

Vincent Kenny

August 2025

Abstract

This report investigates methods for classifying related and unrelated headline-article pairs in the Fake News Challenge. I present a DeBERTa-based model which achieves 93.3% overall accuracy in the task, with an F1 score of 77.3, which is equally weighted across each class. I discuss strategies for addressing class imbalance in the dataset and examine the impact of various hyperparameters, including a modified sliding window tokenisation approach. Performance of Google FLAN-T5 is examined in the same task. This generative ”prompt-based” approach achieves 86.9% overall accuracy, with both zero-shot and chain of thought prompting. I conclude by summarising some of the ethical implications of each approach, calculating the carbon footprint of inference in both models.

1 Data Analysis

1.1 N-gram

There are 49,972 headlines referring to 1,683 articles in the training dataset, which means that each article has approximately 30 associated headlines on average. Therefore, I decided to include duplicate article bodies to preserve distribution details of related vs. unrelated headlines. Discarding duplicates yielded similar n-gram counts across groups, but retaining them revealed nuances (such as articles with many unrelated headlines) that allowed for a more detailed evaluation.

Related Headlines		Related Article Bodies	
N-gram	Count	N-gram	Count
isis	2194	said	39672
foley	1043	video	13812
james	979	one	11550
video	926	isis	11507
journalist	908	state	11362
michael	842	group	11172
says	790	told	9974
islamic	774	would	9965
state	753	also	9822
boko	747	foley	9446
haram	747	islamic	9436
boko haram	747	people	9296
american	723	man	8693
brown	691	new	8536
islamic state	689	syria	8433
audio	678	government	8375
man	670	news	8203
claims	640	could	7750
michael brown	640	islamic state	7699
us	631	iraq	7610

Unrelated Headlines		Unrelated Article Bodies	
N-gram	Count	N-gram	Count
isis	2055	said	34607
foley	873	apple	12504
james	797	one	11217
video	785	would	10206
michael	781	also	9584
journalist	752	people	9581
says	726	told	9330
apple	714	new	9082
man	697	state	8961
state	657	video	8738
islamic	657	isis	8461
brown	633	watch	8215
haram	620	could	7884
claims	620	news	7780
boko haram	620	man	7646
boko	620	time	6959
american	615	according	6925
kim	614	group	6769
audio	603	islamic	6520
islamic state	595	kim	6354

Table 1: Normalised top 20 N-grams for the related and unrelated headline-body pairs

The n-gram counts for the unrelated class have been normalised to enable direct comparison to the related class. Terms from major news stories, such as ”Isis”, ”James Foley”, and ”Boko Haram” appear commonly in both sets of N-grams. In contrast, ”Kim” appears much more frequently in the unrelated set, which reflects the frequent mention of Kim Jong-un in misleading or clickbait headlines. Words like ”state” and ”video” appear more often in related headlines, where they are likely used to provide evidence or context in accurate news coverage.

1.2 Latent Dirichlet Allocation

The number of topics is an important hyperparameter for Latent Dirichlet Allocation (LDA). To determine an appropriate value for this I used the method proposed in [1]. For a series of candidate topic numbers $t_1, t_2, \ldots, t_r$ and their respective perplexities $P_1, P_2, \ldots, P_r$, the rate of perplexity change (RPC) for topic number t_i is given by $RPC(i) = |\frac{P_i - P_{i-1}}{t_i - t_{i-1}}|$. For the first i that satisfies $RPC(i) < RPC(i+1)$, t_i is taken as the most appropriate number of topics. In this case the algorithm recommended 60 topics.

Figure 1: Topic frequency across related article bodies.

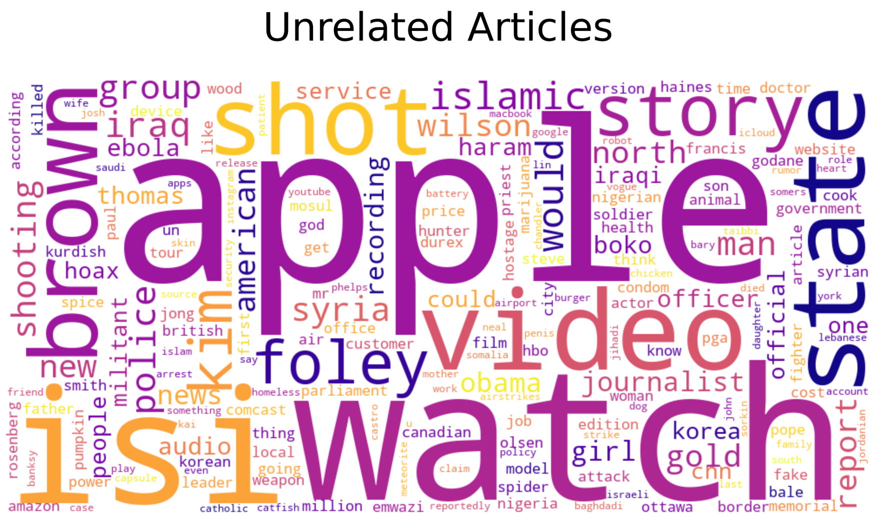

Figure 2: Topic frequency across unrelated article bodies.

These word clouds were constructed by extracting the top 10 tokens from each topic, and multiplying their token weights with the weighting of the topic that they were a member of. Lemmatization was applied in the word-based tokenisation process. This helps to identify relationships between words with common meanings, such as ”walking” and ”walked”. The word ”say” was removed from topic analysis, as it was by far the most common in both related and unrelated sets.

Figure 2 highlights the frequent reference to the Apple Watch in articles with unrelated headlines. This could suggest that technology, and especially new products, appear disproportionally in fake news articles. Sites may include trending terms to boost search engine visibility. In the related set of articles, the topic concerning James Foley had a weighting of 0.07, as opposed to 0.03 for the same story in the unrelated article set. Analysis of the n-gram distributions confirms that this was the most common story in the dataset. This data shows that the related articles are more concentrated on a few large news stories, as opposed to the unrelated articles, which are more evenly spread amongst a broader range of topics.

2 DeBERTa v3 Classification

DeBERTa [2] is a transformer based language model, which improves on BERT [3] and RoBERTa [4] with disentangled attention, separating content and position embeddings, and enhanced masking [5] for better bidirectional context. I trained the model on my personal computer, using an 8GB Nvidia 2070 super GPU.

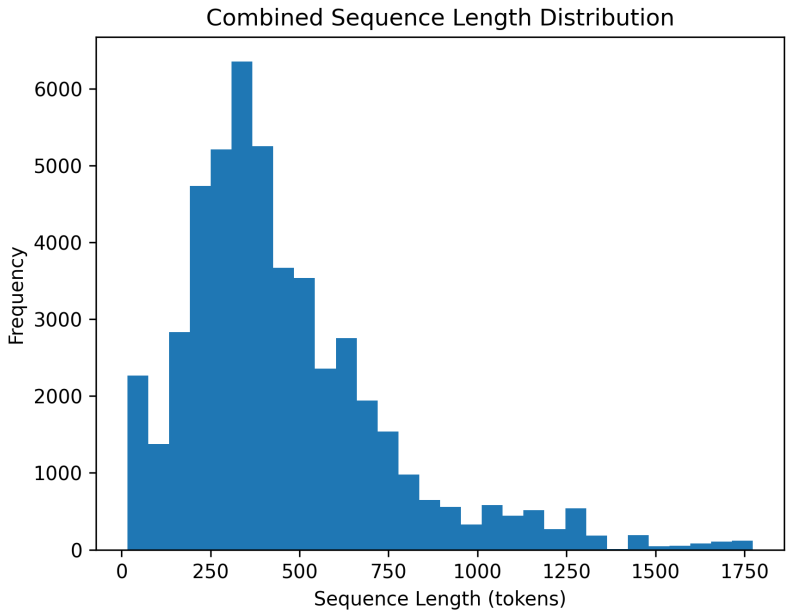

Figure 3: Distribution of the number of tokens in each headline-body pair in the training dataset, excluding the top 1%.

The average sequence length in the dataset is 474 tokens. I experimented with increasing maxSequenceLength above 256, but VRAM limits forced a smaller batch size, which slowed convergence and extended training time.

Given this maxSequenceLength limitation, there is significant loss of information in longer articles; using truncation, any tokens past the 256 cut-off point are lost. One common technique to address this, without increasing maxSequenceLength, is *sliding window tokenisation*, where the input sequence is split into overlapping chunks which are then fed into the transformer sequentially. However, in this use-case, the input sequence is a concatenation of the headline and the article body. Therefore, any chunk other than the first would contain only article body text, destroying the relationship we are looking to capture. To address this, I apply sliding window tokenisation only to the article body and prepend the headline to each chunk.

Basic truncation already achieves high accuracy on the *unrelated* class, but performance on the *agree*, *disagree*, and *discuss* classes is weaker due to dataset imbalance: *unrelated* makes up 73% of labels. To counter this, I apply sliding window tokenisation only to the underrepresented classes, increasing their dataset share and improving classification accuracy while keeping training time low by limiting *unrelated* inputs to one per data point.

All models were trained for 5 epochs, with a base learning rate of 5×10^{-5}.

Model Name	Batch Size	Dropout	Weight Decay	Max Input Length (Tokens)	Class Loss Weighting	Training Tokenizer	Learning Rate Scheduler
Model 1	8	0.1	0.00	256	Uniform	Truncate	Linear
Model 2	4	0.1	0.00	512	Uniform	Truncate	Linear
Model 3	8	0.1	0.00	256	Uniform	Sliding window	Linear
Model 4	8	0.1	0.00	256	Uniform	Sliding window	Cosine
Model 5	8	0.1	0.00	256	Inverse frequency	Sliding window	Cosine
Model 6	8	0.2	0.01	256	Inverse frequency	Sliding window	Linear

Table 2: Hyperparameter configurations for each model.

Model Name	Equal Weighted F1 Score	Overall Accuracy (%)	Class Accuracy (%)			
			Agree	Disagree	Discuss	Unrelated
Model 1	**0.773**	**93.3**	**73.9**	**51.6**	**83.7**	**99.2**
Model 2	0.745	92.6	65.2	46.3	84.8	99.0
Model 3	0.753	91.7	69.9	48.1	86.6	96.8
Model 4	0.739	91.8	73.9	38.0	86.5	97.0
Model 5	0.751	92.0	73.9	49.1	83.7	97.5
Model 6	0.732	91.8	66.9	45.2	85.9	97.6

Table 3: Comparison of model performance in the competition dataset.

Equal weighted F1 score is used to represent model performance across all classes, correcting for imbalance in the frequency of each stance. *Disagree* was the most difficult class for all models, and had the least data with just a 1.68% share of the training dataset. Table 3 shows that Model 1 achieved both the highest Overall accuracy and F1 score, with 93.3% and 0.773 respectively in the competition dataset.

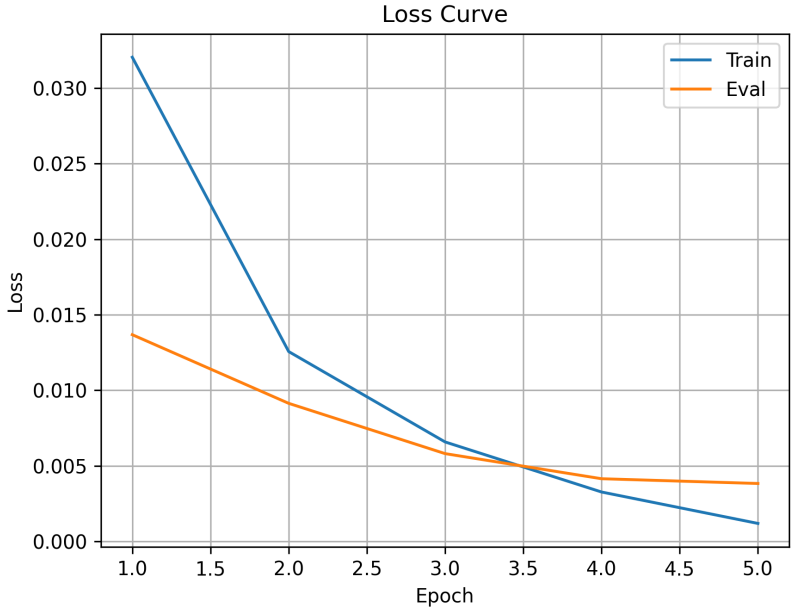

Figure 4: Model 1 loss during training.

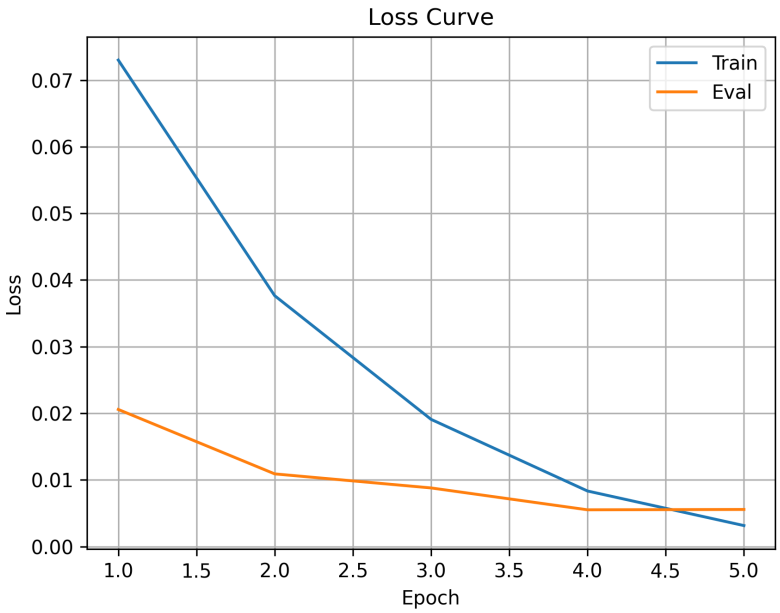

Figure 5: Model 6 loss during training.

In summary, I was unable to improve validation accuracy over Model 1 with any combination of hyperparameters I tested, as demonstrated in Table 3. Both training and test accuracy reached above 99% after 5 epochs in all six models. However, every model showed a notable drop in accuracy on the validation set compared to the training set, suggesting potential overfitting. To mitigate this, I introduced higher dropout rates and applied weight decay to the optimiser in Model 6, aiming to improve generalisation.

Despite these adjustments, Model 6 did not achieve higher validation accuracy. Its loss curve in Figure 5 indicates that these techniques primarily delayed convergence, as opposed to improving generalisability. This perhaps suggests that overfitting is not the main bottleneck for performance here. If the training data is simply not extensive enough to achieve close to 100% accuracy with a BERT-based model, then perhaps a generative approach could be more effective by leveraging pre-trained knowledge and reasoning capabilities.

3 Google FLAN-T5-XL: Classification Prompting

As FLAN is a sequence-to-sequence generative model, it does not natively support the Python Guidance package. Despite this, answer extraction is straightforward even without the SELECT operator. The last word of the response is stripped of punctuation, and converted to lowercase, which reliably yields a valid classification on all 25,413 items in the competition dataset.

For prompt tuning, I analysed performance across the first 1000 headline-body pairs in the competition dataset to keep the runtime manageable. I then ran

the strongest performing prompt from each category over the entire competition dataset for a comprehensive performance assessment.

The full text for each prompt is available in appendix A.

Prompt ID	Description	Accuracy (%)
P1	Baseline zero-shot	87.6
P2	Baseline zero-shot (512 truncation)	86.2
P3	Minimal zero-shot	87.9
P4	Minimal zero-shot variation	88.5
P5	Automatic chain of thought (short examples)	87.4
P6	Automatic chain of thought (real examples)	2.8
P7	Zero-shot chain of thought	88.3
P8	Zero-shot chain of thought variation	88.4
P9	Few-shot chain of thought	76.9
P10	Detailed few-shot chain of thought	86.1
P11	Detailed few-shot chain of thought variation	84.8
P12	Minimal few-shot chain of thought	87.4

Table 4: Performance comparison of different prompt formulations for the first 1000 items in the competition dataset.

I initially expected P6, which incorporated real headline-body pairs from the training set, to perform well. However, I believe the prompt was too long for the model to remember its initial instructions, as it classified every headline as *disagree*, leading to the 2.8% accuracy figure.

Prompt ID	Prompt Type	Overall Accuracy (%)	Class Accuracy (%)			
			Agree	Disagree	Discuss	Unrelated
P4	Zero-shot	86.8	26.5	46.1	68.6	99.0
P5	Automatic CoT	86.9	25.6	52.9	66.4	99.5
P12	Few-shot CoT	86.5	37.9	47.9	62.0	99.0

Table 5: Performance comparison of the top prompts over the entire competition dataset

The prompt with the strongest performance across all stances was P12, achieving 37% accuracy on the *disagree* class. As shown below, this prompt contains no examples, but breaks the classification problem down into two distinct steps. First classifying the pair as related or unrelated, then further classifying related pairs as either *agree*, *disagree* or *discuss*. Chain of Thought (CoT) prompting

is shown to improve model accuracy here in the *disagree* class by 43% over zero-shot prompting; however, overall accuracy saw no improvement.

The primary conclusion that I draw from my findings is that, with FLAN-T5, complex prompt design is not required to achieve high accuracy in this task. FLAN-T5 uses an encoder-decoder architecture which is trained on direct input to output mappings. The input is compressed into a fixed representation before being fed into the decoder, so the intermediate reasoning steps don't directly feed into the generative stage. Prompt engineering would likely yield greater performance gains on decoder only architectures, such as GPT or LLaMA. Ultimately, a concise CoT prompt such as P12 balances efficiency and accuracy, making it the best-performing approach for FLAN-T5 in this task.

Final Zero-Shot Prompt: P4

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

Final CoT Prompt: P12

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.
Think it through step by step, first consider if they are
'unrelated'. If they are related then further classify into 'agree',
'disagree' or 'discuss'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

4 Model Comparison

FLAN-T5 was not trained directly on the fake news dataset, therefore it is notable that its accuracy variation across each class closely mirrors that of DeBERTa. This would suggest that there is some innate higher difficulty in classifying the *agree*, *disagree* and *discuss* classes, which cannot be solely attributed to the imbalanced stance frequencies.

Generative models don't require task specific tuning, allowing them to generalise well to many tasks. FLAN-T5 achieved an accuracy of 86.9% on the competition dataset, where DeBERTa was able to achieve 93.3%. Despite marginally lower

performance, FLAN-T5 didn't require any exposure to the training dataset. Additionally, if the task was modified, adding a 5th class *strongly agree* for example, FLAN-T5 could adapt with just a small tweak to the prompt template, whereas DeBERTa would require complete retraining. This generalisability comes at a cost of incredibly long inference times, coupled with worse task-specific performance in this case.

5 Ethical Considerations

Carbon emissions are a growing concern for modern LLMs. Using the machine learning emissions calculator [6], my DeBERTa classification (5 minutes on an Nvidia 2070 super) emitted 0.01 kg CO_2, whereas FLAN-T5 (3 hours on an A100 in Google Colab) used 0.21 kg CO_2. The generative solution used significantly more energy, with worse accuracy in the task, making it the much less eco-friendly option. These figures only include inference. Training requires far more energy. For example, GPT-3 alone consumed 1287 MW h [7], highlighting the significant environmental impact of large-scale machine learning models.

Objections have also been raised about the use of data in training LLMs, specifically in the area of intellectual property. This paper [8] discusses legal cases brought against companies like OpenAI, who are currently being sued by the New York Times, and others, for unauthorised use of their articles.

These concerns have to be balanced against the considerable social goods that LLMs are capable of generating. To use the example of fake news classification, it would be infeasible for a human operator to sort through and identify inaccurate headlines at the scale of an LLM. Being able to quickly and efficiently classify text in this way is a useful tool for combatting misinformation, and can help to filter dangerous content online.

A Appendix: Full Prompts

P1

```
Classify the relationship between the headline and the article body
as 'agree', 'disagree', 'discuss', or 'unrelated'.
Respond with only one word (agree, disagree, discuss, or unrelated).

Now classify the following:
Headline: "{headline}"
Article Body: "{articleBody}"
```

P2 (Truncation = 512)

```
Classify the relationship between the headline and the article body
as 'agree', 'disagree', 'discuss', or 'unrelated'.
Respond with only one word (agree, disagree, discuss, or unrelated).

Now classify the following:
Headline: "{headline}"
Article Body: "{articleBody}"
```

P3

```
Classify the relationship between the headline and the article body
as 'agree', 'disagree', 'discuss', or 'unrelated'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

P4

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

P5

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.

Examples:
1.
Headline: "New study shows cats reduce stress levels"
Article Body: "A recent scientific study confirms that spending time
with cats can significantly reduce stress."
Answer: agree

2.
Headline: "Climate change is a hoax"
Article Body: "Scientists around the globe provide evidence that
climate change is real and caused by human activity."
Answer: disagree

3.
Headline: "Electric cars will replace petrol vehicles by 2035"
Article Body: "The shift to electric vehicles is gaining momentum,
but infrastructure and consumer adoption remain key challenges."
Answer: discuss

4.
Headline: "The Eiffel Tower was built in 1999"
Article Body: "A new study suggests that drinking coffee can improve
cognitive function and extend lifespan."
Answer: unrelated

Now classify the following:
Headline: "{headline}"
Article Body: "{articleBody}"

Answer with only one category: 'agree', 'disagree', 'discuss', or
'unrelated'.
```

P6

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.
Before providing the final classification, briefly explain your
reasoning.

### **Examples**

Example 1:
Headline: Kim Jong-un ‘is so fat from eating cheese that he has
broken his ankles’
Article Body: Turkey expressed dismay that the weapons intended for
the Kurds ended up with ISIS.
Response: unrelated

Example 2:
Headline: Pumpkin Spice Condoms, And 6 Other Flavors That Aren’t
Happening
Article Body: The Internet was excited to see a picture of a pumpkin
spice flavored condom make the rounds on social media just before
the fall season, but as Mara Montalbano (@maramontlabano) tells us,
it was all a tease.
Response: agree

Example 3:
Headline: We took a look at the photo said to show the body of
Islamic State leader Abu Bakr Al-Baghdadi (right) and found it was
really an ethnic Albanian militant killed in 2013
Article Body: Islamic State of Iraq and Syria leader Abu Bakr
al-Baghdadi probably isn't dead, but no one who's in a position to
know for sure is saying anything.
Response: discuss

Example 4:
Headline: Israeli Canadian fighting ISIS posts on Facebook: I'm safe
and secure
Article Body: Gill Rosenberg (31) from Tel Aviv former IDF soldier
was captured by ISIS after going to Iraq to join Kurds & fight ISIS
Response disagree

### **Now classify the following:**
Headline: "{headline}"
Article Body: "{articleBody}"
Response:
```

P7

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.
Let's think step by step.

Headline: "{headline}"
Article Body: "{articleBody}"
```

P8

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.
Think deeply about the task, consider nuance in the article.

Headline: "{headline}"
Article Body: "{articleBody}"
```

P9

```
Classify the relationship between the following headline and article
body as either 'related' or 'unrelated'.
If they are 'related', then further classify as either 'agree',
'disagree' or 'discuss'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

P10

```
Classify the relationship between the following headline and article
body step by step:

1. First, determine whether the headline and the article body are
'related' or 'unrelated'.
2. If they are 'related', further classify the relationship as one
of the following:
    - 'agree': The headline and article body express the same
    viewpoint.
    - 'disagree': The headline and article body express opposing
    viewpoints.
    - 'discuss': The article body provides additional information
    or discussion about the headline without explicitly agreeing or
    disagreeing.

Headline: "{headline}"
Article Body: "{articleBody}"

Provide your reasoning step-by-step and conclude with the final
classification.
```

P11

```
Classify the relationship between the following headline and article
body step by step:

1. First, determine whether the headline and the article body are
'related' or 'unrelated'.
    - 'related': The headline and article body discuss the same
    topic or content.
    - 'unrelated': The headline and article body are about
    completely different topics.

2. If they are 'related', further classify the relationship as one
of the following:
    - 'agree': The headline and article body express the same
    viewpoint.
    - 'disagree': The headline and article body express opposing
    viewpoints.
    - 'discuss': The article body provides additional information
    or discussion about the headline without explicitly agreeing or
    disagreeing.

Headline: "{headline}"
Article Body: "{articleBody}"

Provide your reasoning step by step and conclude with the final
classification as one of: 'agree', 'disagree', 'discuss', or 'unrelated'.
```

P12

```
Classify the relationship between the following headline and article
body into one of these categories: 'agree', 'disagree', 'discuss',
or 'unrelated'.
Think it through step by step, first consider if they are
'unrelated'. If they are related then further classify into 'agree',
'disagree' or 'discuss'.

Headline: "{headline}"
Article Body: "{articleBody}"
```

References

[1] Weizhong Zhao, James J Chen, Roger Perkins, Zhichao Liu, Weigong Ge, Yijun Ding, and Wen Zou. A heuristic approach to determine an appropriate number of topics in topic modeling. In *BMC bioinformatics*, volume 16, pages 1–10. Springer, 2015.

[2] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

[3] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364, 2019.

[5] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.

[6] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

[7] Alex de Vries. The growing energy footprint of artificial intelligence. *Joule*, 7(10):2191–2194, 2023.

[8] Deven R Desai and Mark Riedl. Between copyright and computer science: The law and ethics of generative ai. *arXiv preprint arXiv:2403.14653*, 2024.