Methods Memo

1. Executive Summary

This memo outlines the methodological approach I developed to quantify differences between two news stories (o-articles & e-articles).

The goal was to move beyond simple "diffs" and capture deeper shifts in framing, style, and semantic integrity. The resulting prototype implements a 12-dimension analysis suite, ranging from surface-level lexical metrics to heavy transformer-based evaluations.

Because we are often comparing single document pairs (N=1), the statistical backbone of this tool relies on sentence-level bootstrapping to generate confidence intervals, ensuring we don't over-interpret random noise.

2. Methodology & Metric Selection

I selected one representative metric per dimension to balance computational cost with interpretability. The analysis pipeline is built in Python using spaCy, sentence-transformers, and transformers.

Surface and Structural Features

For the "fast" features, I stuck to established industry standards that don't require GPU acceleration:

- **Lexical Diversity:** I used **Type-Token Ratio (TTR)** to measure vocabulary richness. Lower TTR in rewrites often indicates simplification.
- **Readability:** I implemented the **Flesch-Kincaid Grade Level (FKGL)** via textstat. This is the standard proxy for text complexity.
- **Stylometry:** To measure stylistic distance, I calculated **Burrows' Delta**. This compares the z-scores of the most frequent function words, effectively measuring how "far apart" the two writing styles are, regardless of content.

Semantic and Content Integrity

To ensure the rewrite didn't drift from the original facts:

- **Semantic Similarity:** I used **SBERT (all-MiniLM-L6-v2)** to compute cosine similarity between aligned sentences. This confirms if the *meaning* was preserved even if words changed.
- **Topic/Entity Shift:** I used a Jaccard similarity index on Named Entities (extracted via spaCy). This serves as a proxy for "topic drift"—if the entities change significantly, the subject has likely changed.
- **Factuality (Proxy):** I built a lightweight **Numeric Jaccard** metric using regex. While not full fact-checking, it flags if specific numbers (dates, dollars, stats) were dropped or altered.

Tone, Framing, and Risk

These dimensions required more specialized models:

- **Sentiment:** I used **VADER Compound scores**. It's lexicon-based and specifically tuned for social media and news contexts, handling intensity well.
- **Framing:** I mapped tokens to the **Moral Foundations Dictionary (eMFD)**. This scores the text based on moral axes (care, fairness, loyalty, etc.), which is crucial for detecting subtle ideological shifts.
- **Sensationalism:** I trained a simple **Naive Bayes Clickbait classifier** on a headline dataset to detect "clickbaity" patterns.
- **Toxicity:** I integrated the **Detoxify** (BERT) model. This detects hostile or toxic language that might be introduced (or scrubbed) during rewriting.

AI Artifacts ("LLM-ness")

To detect if the rewrite feels "machine-generated":

- **Perplexity & GLTR:** I used a small **GPT-2** model to calculate perplexity and rank histograms. Low perplexity and high usage of "top-10" predicted tokens are strong signals of machine-generated text.

3. Statistical Approach: Handling N=1

A major challenge in comparing just two articles is statistical validity. We cannot run a t-test on two single numbers.

To solve this, I implemented **sentence-level bootstrapping**. By resampling the sentences 1,000 times, we generate a distribution of means for each metric. This allows us to:

1. Calculate 95% Confidence Intervals (CI) for the difference.

2. Compute **Cohen's d** effect sizes to see if the difference is actually meaningful or just trivial variation.

4. Extended Metric Set

If I were to take this project further, I found a few other metrics in my research that would be useful for deeper analysis, though they were too complex for this initial prototype:

1. **Yule's K (Lexical):** A more stable version of TTR for comparing files of very different lengths.
2. **MFRC Classifiers (Framing):** A context-aware model for moral framing, which would be more accurate than the keyword dictionary I used.
3. **BERTScore (Semantics):** This would allow me to pinpoint exactly *which* word substitutions changed the meaning, rather than just getting a similarity score for the whole sentence.
4. **LWBOW (Structure):** A method to track style shifts continuously across the document, rather than just averaging them.

5. Risks and Threats to Validity

1. **Metric Validity (Fact-Checking):** My "Factuality" metric matches numbers, not truth. If an article changes "$10 million profit" to "$10 million loss," my metric sees 100% overlap and assumes it's accurate. This is a blind spot.
2. **Model Bias:** The Detoxify and Sentiment models are trained on internet data. They may flag discussions of marginalized groups as "toxic" simply because those keywords often appear in toxic contexts online.
3. **Length Sensitivity:** Metrics like TTR are sensitive to text length. If the rewrite is significantly shorter, TTR naturally inflates, potentially creating a false signal of "richer vocabulary."

6. Conclusion

The prototype successfully implements a multi-dimensional view of article transformation. The combination of lightweight statistical proxies and deep learning transformers gives us a comprehensive dashboard. Moving forward, I recommend prioritizing the Minimal Set for high-volume scanning and reserving the Extended Set for detailed auditing of flagged articles.