**1. Exponential Decay Approach**

Mathematical Function:

decay\_factor = np.exp(-decay\_rate \* batches\_ago)

**Key Features:**

- Models topic novelty using exponential decay

- Recent topics have higher impact that decays exponentially over time

- Combines with topic frequency penalty (1/√count)

- 70% weight to position ratio, 30% to novelty

**Pros:**

- Simple to implement

- Matches intuition that recent topics matter more

- Smooth decay pattern

**Cons:**

- May decay too quickly for longer speeches

- Fixed decay rate may not fit all scenarios

**2. Power Law Approach**

Power law provides slower decay than exponential, better for long-form content where topics might resurface after extended periods

Mathematical Function:

decay\_factor = (batches\_ago + 1) \*\* (-alpha)

**Key Features:**

- Uses power law (polynomial) decay instead of exponential

- Slower initial decay but heavier long-term tail

- Similar frequency penalty as exponential approach

- Same 70/30 position-novelty weighting

**Pros:**

- Better for modeling long-range dependencies

- Matches patterns seen in natural discourse

- More flexible with alpha parameter

Cons:

- May give too much weight to very old topics

- Requires tuning of alpha parameter

**3. MACD(Moving Average Convergence Divergence) Staleness Approach**

Adapted from financial analysis, captures trend changes more sensitively than linear regression

Mathematical Function:

ema\_fast = alpha\_fast \* novelty + (1 - alpha\_fast) \* ema\_fast

ema\_slow = alpha\_slow \* novelty + (1 - alpha\_slow) \* ema\_slow

staleness = max(0, (ema\_slow - ema\_fast) \* 10

**Key Features:**

- Inspired by Moving Average Convergence Divergence (MACD) from finance

- Uses two exponential moving averages (fast and slow)

- Measures "staleness" as divergence between averages

- 60% position ratio, 40% staleness

**Pros:**

- Good at detecting trend changes

- Adapts to different speech paces

- Combines short-term and long-term views

**Cons:**

- More complex to understand

- Requires tuning two parameters (α\_fast, α\_slow)

**4. Gaussian Memory Approach**

Creates a "memory window" where recent topics are heavily penalized, but very old topics regain novelty more gracefully

Mathematical Function:

decay\_factor = np.exp(-(batches\_ago \*\* 2) / (2 \* sigma \*\* 2))

**Key Features:**

- Uses Gaussian (normal distribution) shaped memory window

- Topics have strongest impact when recently seen

- Smooth symmetric decay in both directions

- Same 70/30 weighting as other approaches

**Pros:**

- Matches psychological memory models

- Smooth transition between recent and old topics

- Flexible with sigma parameter

**Cons:**

- Computationally slightly more intensive

- May not fit scenarios needing asymmetric decay

**Key Mathematical Insights**

1. Decay Patterns:

- Exponential: e^(-kx)

- Power Law: x^(-α)

- Gaussian: e^(-x²/2σ²)

2. Position vs Novelty:

All approaches combine positional information (how far along the speech is) with topic novelty/staleness measures, but with different weightings.

3. Memory Modeling:

The approaches differ in how they model "memory" of past topics - from rapid forgetting (exponential) to long retention (power law).

4. Parameter Sensitivity:

- Exponential and Power Law have single parameters controlling decay speed

- MACD has two parameters for fast/slow adaptation

- Gaussian has σ controlling memory window width

The best approach depends on the speech characteristics - for example, MACD works well for presentations with clear phases, while Gaussian memory may better model natural conversations.