Confidence-Weighted PMI: An Information-Theoretic Approach with Proper Error Propagation

[John Creighton (AKA s243a) in collaboration with Claude Sonet 3.5]

October 27, 2024

Abstract

We present a novel approach to calculating Pointwise Mutual Information (PMI) that combines information theory, Bayesian statistics, and error propagation analysis. Our method addresses traditional PMI calculation challenges through theoretically grounded smoothing techniques that properly account for uncertainty in both observed and expected probabilities. Key innovations include entropy-based degrees of freedom calculation, proper error propagation in probability products, and a unified treatment of statistical confidence. Results show improved handling of both rare and common n-grams while maintaining proper probabilistic interpretation.

1 Introduction

Pointwise Mutual Information (PMI) is a fundamental measure in computational linguistics that quantifies word associations. While powerful, traditional PMI calculations face several challenges:

- Undefined values for unseen combinations
- Unreliable estimates for rare events
- Lack of proper error propagation
- Mixing of count and probability spaces

1.1 Historical Context and Innovation

Our approach builds on established foundations while introducing several novel elements:

1.1.1 Established Components

- Laplace smoothing (Laplace, 1812) [8]
- Linguistic significance testing (Church & Hanks, 1990) [3]
- Dirichlet priors (MacKay & Peto, 1995) [9]

1.1.2 Novel Contributions

- Entropy-based degrees of freedom
- Proper error propagation in probability products
- Units-consistent smoothing formulation
- Confidence-weighted probability adjustment

2 PMI Framework

2.1 Traditional Definition

The classical PMI formula for words x and y is:

$$\text{PMI}(x, y) = \log \frac{P(x, y)}{P(x)P(y)} \tag{1}$$

For n-grams, this extends to:

$$\text{PMI}(w_1, \ldots, w_n) = \log \frac{P(w_1, \ldots, w_n)}{P(w_1) \cdots P(w_n)} \tag{2}$$

2.2 Challenges in Traditional Approaches

2.2.1 Zero Probabilities

When P(x,y) = 0, PMI is undefined. Traditional solutions include:

- Add-one smoothing
- Add-α smoothing

- Good-Turing estimation

These approaches often lack theoretical justification and can introduce biases.

2.2.2 Type Mismatches

Traditional smoothing often mixes different types of quantities:

$$P_{\text{smoothed}} = \frac{\text{count} + \alpha}{N + \alpha k} \tag{3}$$

where count is an integer (occurrence count) and α is a probability-like quantity (0 to 1). This mixing of types raises several concerns:

- Dimensional inconsistency: Adding counts to probabilities violates unit homogeneity principles [**?**]
- Scale sensitivity: The effect of α varies dramatically with corpus size
- Interpretation difficulty: The smoothed result lacks clear probabilistic interpretation

Dimensional Analysis Consider the dimensions in traditional smoothing:

- count: [occurrences]
- α: [dimensionless]
- N: [total occurrences]
- k: [possible outcomes]

The expression count + α combines [occurrences] with [dimensionless], violating the principle of dimensional homogeneity [**?**, 1]. This is analogous to adding meters to unitless numbers in physics [12].

Real-World Effects This type mismatch leads to practical issues:

1. Corpus Size Dependency: For a small corpus (N = 1000):

$$P_{\text{smoothed}} = \frac{5 + 0.5}{1000 + 0.5k} \approx 0.005 \tag{4}$$

For a large corpus (N = 1,000,000):

$$P_{\text{smoothed}} = \frac{5000 + 0.5}{1,000,000 + 0.5k} \approx 0.005 \tag{5}$$

The smoothing effect becomes negligible with corpus size.

2. Cross-Corpus Comparison: When comparing across corpora of different sizes, the same α value produces inconsistent smoothing effects [2].

Our formulation maintains type consistency by operating entirely in probability space:

$$P_{\text{smoothed}} = \frac{w \cdot P + \alpha(1 - \text{confidence})}{1 + w} \tag{6}$$

where all terms are dimensionless probabilities or probability ratios. This ensures:

- Scale invariance with corpus size
- Consistent interpretation across datasets
- Proper probabilistic semantics

This approach aligns with modern statistical practice [6] and measurement theory [7].

2.3 Information Theory Perspective

PMI relates to mutual information through expectation:

$$I(X;Y) = \mathbb{E}_{x,y}[\text{PMI}(x,y)] \tag{7}$$

Shannon entropy provides a natural framework for uncertainty:

$$H(X) = -\sum_{x} P(x) \log P(x) \tag{8}$$

- The entropy can be used to calculate the degrees of freedom (See appendix B.1)
- The mutual information can be used to calculate probabilities via the partition function (See Appendix C.2)

2.4 Statistical Significance

The t-statistic measures deviation from independence:

$$t = \frac{P_{\text{observed}} - P_{\text{expected}}}{\text{SE}} \tag{9}$$

For our standard error estimation, we use the form:

$$\text{SE} = \sqrt{\frac{p(1-p)}{\text{df}}} \tag{10}$$

where p represents the probability of the word or n-gram occurring at any position, and df is calculated using our entropy-based approach. This formulation treats each position in the text as a binary trial (presence or absence of the target sequence), with degrees of freedom accounting for the effective number of independent observations.

3 Enhanced PMI Formulation

3.1 Unit-Consistent Smoothing with Statistical Weighting

Our approach combines confidence-weighted smoothing with statistical significance:

$$P_{\text{smoothed}}^{\text{ngram}} = \frac{w \cdot P_{\text{observed}} + \alpha(1 - \text{confidence})}{1 + w} \tag{11}$$

$$P_{\text{smoothed}}^{\text{expected}} = \frac{w \cdot P_{\text{expected}} + \alpha(1 - \text{confidence})}{1 + w} \tag{12}$$

where:

- $w = 1/p$ is the statistical weight (p is two-tailed p-value) [1]
- $\alpha = \sqrt{\text{relative\_variance}}/\sqrt{\text{df}}$ is the smoothing parameter
- confidence is derived from the t-statistic

3.2 Statistical Significance

The t-statistic measures deviation from independence:

$$t = \frac{P_{\text{observed}} - P_{\text{expected}}}{\text{SE}} \tag{13}$$

The p-value for this statistic determines our weighting:

$$p = 2(1 - \text{CDF}_t(|t|, \text{df})) \tag{14}$$

This formulation provides:

[1] 1/p is referred to as surprise by some authors [11]. The term "binary surprise index" appears in [4], where it is attributed to Shannon's work [10].

- Strong weighting ($w \gg 1$) for statistically significant associations
- Balanced weighting ($w \approx 1$) for borderline cases
- Proper probability normalization through $(1 + w)$ denominator

3.3 Error Propagation

For products of probabilities, relative errors add in quadrature:

$$\text{rv\_num} = \sum_i \left(\frac{\text{se}_i}{\text{prob}_i} \right)^2 \tag{15}$$

Normalization ensures proper scaling:

$$\text{rv\_den} = \sum_i \left(\frac{1}{\text{prob}_i} \right)^2 \tag{16}$$

The relative variance is then:

$$\text{relative\_variance} = \frac{\text{rv\_num}}{\text{rv\_den}} \tag{17}$$

3.4 Entropy-Based Degrees of Freedom

Degrees of freedom calculation incorporates uncertainty:

$$\text{df} = \exp(H) \cdot \text{expected\_occurrences} - \text{ngram\_length} \tag{18}$$

This naturally handles:

- Rare events
- Multiple observations
- Structural constraints

4 Statistical Analysis of Error Propagation

4.1 Probability Space Smoothing

Our smoothing operates consistently in probability space:

$$P_{\text{smoothed}}^{\text{ngram}} = \frac{\text{ngram\_count}}{\text{total\_unigrams}} + \alpha(1 - \text{confidence}) \tag{19}$$

$$P_{\text{smoothed}}^{\text{expected}} = P_{\text{expected}} + \alpha(1 - \text{confidence}) \tag{20}$$

where α is derived from proper error propagation.

4.2 Error Propagation in Products

For the expected probability (product of unigram probabilities), the relative error follows from the product rule of differentiation:

$$\frac{\partial}{\partial x_i} \prod_j x_j = \prod_{j \neq i} x_j \tag{21}$$

Dividing by the product yields proportional errors:

$$\frac{1}{\prod_j x_j} \frac{\partial}{\partial x_i} \prod_j x_j = \frac{1}{x_i} \tag{22}$$

4.3 Variance Normalization

The relative variance components:

$$\text{rv\_num} = \sum_i \left(\frac{\text{se}_i}{p_i} \right)^2 \tag{23}$$

$$\text{rv\_den} = \sum_i \left(\frac{1}{p_i} \right)^2 \tag{24}$$

ensure that standard error per degree of freedom equals the expected unigram standard error.

5 Core Implementation

```
def calculate_pmi_with_t_score(self, ngram):
    # Calculate probabilities
    ngram_prob = ngram_count / total_unigrams

    # Calculate relative variance
    rv_num = sum((ws['se']/ws['prob'])**2
                 for ws in word_stats)
    rv_den = sum((1/ws['prob'])**2
                 for ws in word_stats)
    relative_variance = rv_num / rv_den

    ws = Calculate_WordStats(ngram)
```

```
    # Calculate expected probability and error
    expected_prob = np.prod([ws['prob'] for ws in word_stats])

    # Calculate degrees of freedom
    ngram_df = calculate_entropy_df(ngram)

    ngram_se = math.sqrt((ngram_prob * (1-ngram_prob)) / ngram_df)

    t_stat = abs((ngram_prob - expected_prob)/ngram_se)
    p_value = 2 * (1 - stats.t.cdf(t_stat, df=ngram_df))
# Two-tailed test
    w = 1/p_value  # Or some function of p_value that grows with surpr

    # Confidence-weighted smoothing
    confidence = 1 - stats.t.cdf(t_stat, df=ngram_df)

    # Smoothing parameters
    alpha = math.sqrt(relative_variance) / math.sqrt(ngram_df)

    # Apply smoothing
    smoothed_ngram_prob = ( w * ngram_prob +
        alpha * (1 - confidence)) / w
    smoothed_expected_prob = w* (expected_prob +
        alpha * (1 - confidence)) / w

    return math.log(smoothed_ngram_prob /
                    smoothed_expected_prob)
```

6 Generalizing the Significance Weighting (i.e. w)

In the previous section we use a factor w to weight higher the observed results higher when they are more statistically significance. Specifically we set:

$$w(statistics) := 1/p(x) \tag{25}$$

We are using the notation

$$" := "$$

as set denote one possibility. For instance, we could have used $(1/(p(x))^2$, in which case the PMI would converge fast to the value implied directly from

the data as the significance increases. It's worth noting that in the above section the standard error was inferred from the n-gram statistics but we could also have a standard error in the expected value statistics based on sub-n-grams such as uni grams.

So there are two natural measures of statistical significance, one is with respect to the n-gram statistics (the observed), and one is with respect to what would be expected given it's constituent components (e.g. uni grams).

In the later case multiple measures could be derived based on the components of sub-n-grams and assumptions about there statistics.

In the next subsections, we will look at both a recursive approach based on Bayesian statistics which we will reserve for future work, and a unigram approach that assumes statistical independence.

6.1 Error Propagation with Multinomial Statistics

For a sequence of independent unigrams, the multinomial standard error is:

$$\text{SE}_{\text{multinomial}} = \sqrt{\sum_i \frac{p_i(1-p_i)}{n} - \sum_{i \neq j} \frac{p_i p_j}{n}} \tag{26}$$

where p_i are the individual unigram probabilities and n is our degrees of freedom.

6.2 Future Work

6.2.1 Sequential Dependency in N-grams

A more sophisticated approach would leverage the hierarchical nature of n-grams:

$$P(w_1 ... w_n) = P(w_1 ... w_{n-1}) P(w_n | w_1 ... w_{n-1}) \tag{27}$$

This leads to a recursive standard error formulation:

$$\text{SE}_{n\text{-gram}} = \sqrt{\left(\frac{\text{SE}_{n-1\text{-gram}}}{P_{n-1\text{-gram}}}\right)^2 + \left(\frac{\text{SE}_{w_n}}{P_{w_n}}\right)^2} \tag{28}$$

This approach:

- Captures sequential dependencies
- Uses available statistics efficiently
- Maintains computational tractability
- Provides natural extension to higher-order n-grams

7 Confidence Intervals and Statistical Significance

7.1 Asymmetric Confidence Bounds

For PMI as a ratio of random variables:

$$\text{PMI} = \log\left(\frac{X + e_1}{Y + e_2}\right) \tag{29}$$

where X is observed frequency, Y is expected frequency, and e_1, e_2 are error terms.

7.2 95% Confidence Interval

The minimum PMI value for 95% confidence satisfies:

$$\int_{\text{PMI}_{\min}}^{\infty} p(\text{PMI})\, d\text{PMI} = 0.95 \tag{30}$$

7.3 Bayesian Interpretation

In the low-data regime, the posterior distribution incorporates prior information:

$$p(\text{PMI}|\text{data}) \propto p(\text{data}|\text{PMI})p(\text{PMI}) \tag{31}$$

8 Results and Validation

8.1 Common Bigrams

Example results showing proper handling of frequent combinations:

```
'do n't': PMI = 1.92
'has been': PMI = 0.98
'ca n't': PMI = 0.97
```

8.2 Rare Combinations

Demonstration of appropriate smoothing for rare events:

```
'jet propulsion': PMI = 0.18
'propulsion laboratory': PMI = 0.16
```

9 Discussion and Future Work

9.1 Theoretical Implications

The units-consistent approach provides several advantages:

- Proper error propagation
- Natural confidence weighting
- Theoretical connection to information theory

9.2 Future Directions

Areas for further research:

- Alternative error distribution models
- Extension to higher-order n-grams
- Domain-specific applications

A Derivation of Error Propagation

A.1 Product Rule Application

For a product of probabilities $P = \prod_i p_i$, the error propagation follows:

$$\Delta P = \sqrt{\sum_i \left(\frac{\partial P}{\partial p_i} \Delta p_i \right)^2} \tag{32}$$

After normalization:

$$\left(\frac{\Delta P}{P} \right)^2 = \sum_i \left(\frac{\Delta p_i}{p_i} \right)^2 \tag{33}$$

A.2 Confidence Interval Derivation

For asymmetric confidence bounds:

$$P(\text{PMI} > \text{PMI}_{\text{min}}) = \int_{\text{PMI}_{\text{min}}}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \tag{34}$$

B Implementation Details

B.1 Entropy Calculation

```
def calculate_entropy_df(self, ngram):
    words = ngram.split()
    probs = []
    for word in words:
        count = self.word_freq.get(word, 0)
        prob = count / self.total_unigrams
        probs.append(prob)

    H = -sum(p * math.log(p)
             for p in probs if p > 0)

    return math.exp(H) * expected_occurrences \
           - len(words)
```

B.2 Error Propagation Implementation

```
# Calculate relative variance components
rv_num = sum((ws['se']/ws['prob'])**2
             for ws in word_stats)
rv_den = sum((1/ws['prob'])**2
             for ws in word_stats)
relative_variance = rv_num / rv_den

# Confidence-weighted smoothing
alpha = math.sqrt(relative_variance) \
        / math.sqrt(df)
```

C Additional Theoretical Background

C.1 Information Theory Connection

The relationship between entropy and degrees of freedom:

$$\text{df} \approx e^H \tag{35}$$

provides a natural measure of effective sample size.

C.2 Statistical Mechanics Analogy

The entropy-based approach connects to partition functions in statistical mechanics:

$$Z = \sum_i e^{-\beta E_i} \tag{36}$$

where energy levels correspond to probability ratios.

the partition function can be used to calculate probabilities as follows:

$$P(E_i) = exp(-\beta E_i)/Z \tag{37}$$

References

[1] Bridgman, P. W. (1922). Dimensional analysis. Yale University Press. https://archive.org/details/dimensionalanaly00bridrich

[2] Chen, S. F., & Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. Computer Speech & Language, 13(4), 359-394. https://arxiv.org/abs/cmp-lg/9606011 [arXiv]

[3] Church, K. W., & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1), 22-29. https://aclanthology.org/J90-1003/ [OA]

[4] Cole, S. C. (2021). Surprise! American Journal of Epidemiology, Volume 190, Issue 2, February 2021, Pages 191–193, https://doi.org/10.1093/aje/kwaa136 [OA]

[5] Good, I. J. (1953). The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3-4), 237-264. https://doi.org/10.1093/biomet/40.3-4.237

[6] Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). Bayesian data analysis. Chapman and Hall/CRC. https://doi.org/10.1201/b16018

[7] Hand, D. J. (2004). Measurement theory and practice: The world through quantification. Arnold London.

[8] Laplace, P. S. (1812). Théorie analytique des probabilités. *Courcier*, Paris. https://archive.org/details/thorieanalytiqu00laplgoog [OA]

[9] MacKay, D. J. C. & Peto, L. C. (1995). A hierarchical Dirichlet language model. *Natural language engineering*, 1(3), 289-308. https://doi.org/10.1017/S1351324900000218

[10] Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27(3), 379-423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

[11] Stone, D. J. (2019) Information Theory: A Tutorial Introduction https://arxiv.org/abs/1802.05968 [arXiv]

[12] Taylor, J. R. (1997). An introduction to error analysis: The study of uncertainties in physical measurements. University Science Books. https://archive.org/details/introductiontoer00tayl