Precog Recruitment Task

Language Representations — Part 1: Dense Representations

Nilanjan Sarkar
MS by Research, CSE
ID: 2025701014

Abstract

This report documents a complete count-based pipeline for learning dense word embeddings from a large English corpus ($>$ 300 K sentences), together with systematic evaluation and a comparison to a neural baseline (pre-trained fastText). Starting from a tokenized corpus, I construct word–word co-occurrence matrices for multiple context window sizes, transform counts to *Positive Pointwise Mutual Information* (PPMI), and obtain d-dimensional embeddings via *Truncated Singular Value Decomposition* (SVD). I select the window size and dimensionality using a blend of diagnostics: explained variance ratio, reconstruction *Root Mean Squared Error* (RMSE) on PPMI, neighborhood rank stability (median Spearman correlation), clustering silhouette, and downstream clustering of curated semantic seed sets (purity and *Normalized Mutual Information* (NMI)). The best-performing configuration in my runs is window W=8, dimension d=300. Count-based PPMI+SVD yields coherent neighborhoods and perfect seed clustering on my probe sets, while pre-trained fastText outperforms on SimLex-999 and WordSim-353 similarity.

1 Corpus and Preprocessing

Tokenization. I lowercased text and retained alphabetic tokens only. A minimum frequency threshold of `min_count=100` produced a vocabulary of $N = 5314$ types. The in-vocabulary token stream comprised roughly 5,219,269 tokens (from the news-like corpus supplied in the notebook).

Vocabulary mappings. I built `word2id` and `id2word` dictionaries to index the co-occurrence matrix rows/columns.

2 Co-occurrence Construction and Window Experiments

For each sentence, I accumulated symmetric co-occurrence counts with distance weighting $1/\Delta$ for token distance $\Delta \in \{1, \ldots, W\}$. I swept window sizes $W \in \{2, 5, 8, 10\}$. Each count matrix was stored as a Compressed Sparse Row (CSR) matrix.

Neighbor sanity checks (raw counts). For anchors such as *india*, *government*, *football*, *market*, top co-occurrence neighbors included plausible function words at small W and more topical terms as W grew. A Jaccard overlap analysis of top-10 neighbor sets showed high stability once $W \geq 5$ (e.g., for *government*, Jaccard = 1.00 for W=8 vs W=10). The main takeaway is that context $W \in [5, 10]$ captures similar lexical neighborhoods while avoiding overly local noise.

Table 1: Sparsity summary across window sizes ($N = 5314$).

Window W	Nonzeros (nnz)	Density	Notes
2	2,727,946	0.0966	tight context
5	5,292,484	0.1874	broader
8	6,759,486	0.2394	best later
10	7,416,895	0.2627	widest

3 From Counts to Positive PMI (PPMI)

Pointwise Mutual Information (PMI). For word w and context c, with probabilities estimated from counts,

$$\mathrm{PMI}(w, c) = \log \frac{p(w, c)}{p(w)\, p(c)}.$$

Positive PMI (PPMI). $\mathrm{PPMI}(w, c) = \max\{0, \mathrm{PMI}(w, c)\}$ zeroes negative associations, which empirically improves linear structure for embeddings. I applied log-smoothing and computed PPMI for each W.

4 Dimensionality Reduction via Truncated SVD

Singular Value Decomposition (SVD). For matrix $X \in \mathbb{R}^{N\times N}$, SVD factorizes $X \approx U\,\Sigma\,V^\top$. *Truncated* SVD keeps the top d singular components to yield X_d. I used scikit-learn's sparse-aware `TruncatedSVD` on PPMI and row-normalized the resulting embeddings. I evaluated $d \in \{50, 100, 200, 300\}$.

Explained Variance and Reconstruction RMSE

Explained variance ratio (EVR). EVR increased smoothly with d and with larger W. Example (W=8): EVR=0.139 ($d = 50$), 0.180 (100), 0.245 (200), 0.301 (300).

Reconstruction RMSE. I estimated *Root Mean Squared Error* on a random sample of PPMI entries using the low-rank reconstruction. Lower is better. RMSE decreased as W increased and was relatively flat in d (e.g., W=10: RMSE $\approx$ 1.155 at d=50 to 1.145 at d=200).

5 Choosing the Embedding Dimension d

Beyond EVR and RMSE, I used two additional diagnostics:

- **Neighborhood stability.** For a probe set, I computed the median *Spearman rank correlation* between similarity rankings at dimension d vs a high-capacity reference (d=300). Stability rose with d (e.g., W=8: $\rho = 0.882$, 0.917, 0.947, 1.000 for $d = 50$, 100, 200, 300).
- **Silhouette for unsupervised clusters.** Using MiniBatch KMeans on random subsets with cosine distance, average silhouette modestly decreased with d (e.g., W=8: 0.106 $\rightarrow$ 0.055 from d=50 to 300), which is common as neighborhoods become denser. Given downstream results (next section), I favored stability/EVR over raw silhouette.

Choice. Aggregating these signals, I selected d=300 for all W. The final window choice (§6) emerged from downstream evaluations, with W=8 best overall.

6 Intrinsic Evaluation and Window Selection

6.1 Seed-category clustering

I curated small, interpretable seed sets (e.g., countries, sports, professions) and clustered their embeddings with KMeans. I report **purity** (fraction of items assigned to their majority class per cluster, averaged) and **Normalized Mutual Information (NMI)** between predicted clusters and true categories.

Table 2: Seed clustering at d=300.

Window W	Purity	NMI
2	0.979	0.962
5	0.979	0.968
8	**1.000**	**1.000**
10	0.917	0.898

Outcome. W=8 achieved perfect separation on these probes and was chosen as the default window for subsequent analyses.

6.2 Lexical similarity benchmarks

Using cosine similarity on embeddings for word pairs present in the vocabulary, I computed **Spearman's rank correlation** (ρ) with human ratings.

- **SimLex-999** (concrete similarity): $\rho = 0.210$ (437 pairs covered).
- **WordSim-353** (relatedness): $\rho = 0.581$ (188 pairs covered).

6.3 Neighborhood and visualization sanity checks

Two-dimensional **Principal Component Analysis (PCA)** projections of seed words showed clear category clusters; **t-Distributed Stochastic Neighbor Embedding (t-SNE)** on 500 frequent content words yielded locally coherent topical groupings. For anchors, nearest neighbors were thematically sensible (e.g., *india* near *pakistan*, *china*, *asia*).

7 Comparison with a Neural Baseline (fastText)

I loaded pre-trained **fastText** English vectors (subword-aware neural embeddings) and evaluated them identically:

- Coverage: 5301/5314 vocabulary words matched.
- Seed clustering: purity 0.938, NMI 0.907 (lower than PPMI+SVD on these curated sets).
- Similarity: SimLex-999 $\rho = 0.379$, WordSim-353 $\rho = 0.740$ (higher than PPMI+SVD).

Interpretation. The subword and larger-corpus training of fastText favor lexical similarity and analogy structure, while the task-tailored PPMI+SVD with W=8 excelled at clustering my curated semantic seeds—consistent with different inductive biases.

8 Discussion and Takeaways

Window size. Empirically, W=8 offered the best downstream clustering and stable neighborhoods while avoiding the function-word dominance seen at very small windows and the noisier conflation at the widest window.
Dimension d. Larger d improved explained variance and neighbor stability; although silhouette decreased slightly, downstream clustering and similarity tasks were not harmed. I therefore fixed d=300.
Count vs. neural. PPMI+SVD produced highly interpretable structure on curated seeds; fastText excelled on lexical similarity and analogies. This complementarity suggests ensemble or retrofitting opportunities.