CS546: GloVe
Global Vectors for Word Representation
Aming Ni
Overview

- One Hot Encoding
- Global Matrix Factorization
- Local Context Window
- Short Intro to Skip-Gram
- GloVe
- GloVe V.S. Skip-Gram
- Results on GloVe
One Hot Encoding

- Sparsity: High OOV rate, huge # of parameters.
- Language models such as n-gram?
- We want:
  - Reduce # of parameters
  - Utilize both local and global information
  - Generalization
- Distribution hypothesis: words appear in similar contexts should be similar.
Global Matrix Factorization

- Utilize low-rank approximations to decompose large matrices that capture statistical information about a corpus.
- Latent Semantic Analysis (LSA)
  - The matrices are of “term-document” type
    - The rows correspond to words, and the columns correspond to different documents
  - Use a rank-k SVD to preserve the similarity structure among columns.
- PMI Matrix: perform a rank-k SVD on the matrix.
Local Context Window

- Learn the word representations in full context, rather than just the preceding context as is the case with language models.

- Continuous Bag of Words (CBOW)
  - Objective is to predict a word given its context

- Word2vec/Skip-Gram
  - Objective is to predict a context given a word.
CBOW and Skip-Gram Models
Short Intro Skip-Gram

- Maximize the average log probability: \( \frac{1}{N} \sum_{t=1}^{N} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} | w_t) \)

- One possible model softmax: \( p(w_j | w_i) = \frac{\exp(w_i^T w_j)}{\sum_{k=1}^{V} \exp(w_i^T w_k)} \)

- One problem: vocabulary size can be huge, each \( w_j \) in \( p(w_j | w_i) \) takes \( O(|V|) \) to compute
  - Solution: Hierarchical Softmax, Negative Sampling.
Why GloVe?

- Try to use both global statistics and local window context.
- Train only on the *nonzero elements* in a word-word co-occurrence matrix.
- Propose a specific *weighted least squares model* that trains on global word-word co-occurrence counts.
Some notations

- $X$: the matrix of word-word co-occurrence
- $X_{ij}$: number of times word $j$ occurs in the context of word $i$
- $X_i = \sum_k X_{ik}$: the number of times any word appears in the context of the word $i$
- $P_{ij} = P(j|i) = X_{ij} / X_i$: the probability that word $j$ appears in the context of word $i$
- $w_i$: the representation of word $i$ (if in vector form, $w_i \in \mathbb{R}^d$)
Simple Example for Co-occurrence Probabilities

- Co-occurrence probabilities for target words *ice* and *stream* with selected context words
- Noise words like *water* and *fashion* cancel out (close to zero)
- Intuitively, the score for solid/gas given context ice/stream should be high.
- This suggests that we should look at the ratios (relatively normalized) of co-occurrence probabilities rather than the pure co-occurrence probabilities: \( F(w_i,w_j,w_k) = \frac{P_{ik}}{P_{jk}} \)

<table>
<thead>
<tr>
<th>Probability and Ratio</th>
<th>k = solid</th>
<th>k = gas</th>
<th>k = water</th>
<th>k = fashion</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(k</td>
<td>ice)</td>
<td>1.9 x 10^{-4}</td>
<td>6.6 x 10^{-5}</td>
<td>3.0 x 10^{-3}</td>
</tr>
<tr>
<td>P(k</td>
<td>stream)</td>
<td>2.2 x 10^{-5}</td>
<td>7.8 x 10^{-4}</td>
<td>2.2 x 10^{-3}</td>
</tr>
<tr>
<td>P(k</td>
<td>ice) / P(k</td>
<td>stream)</td>
<td>8.9</td>
<td>8.5 x 10^{-2}</td>
</tr>
</tbody>
</table>
Adding Assumptions & Derivations

✧ We want the *ratio*, now start with this equation: \( F(w_i, w_j, w_k) = \frac{p_{ik}}{p_{jk}} \)

✧ Let’s enforce linear structures in vector space, we could use the difference. This assumption restricts us to only those functions of: \( F(w_i - w_j, w_k) = \frac{p_{ik}}{p_{jk}} \)

✧ Since \( \frac{p_{ik}}{p_{jk}} \) is a scalar, we could further restrict \( F \) to be: \( F((w_i - w_j)^T w_k) = \frac{p_{ik}}{p_{jk}} \)

✧ We can restrict \( F \) to be a homomorphism function \( \text{exp}(\text{structure preserving mapping}) \)
Adding Assumptions & Derivations Cont...

- If \( F \) is an exponent, then:
  \[
  F((w_i - w_j)^T w_k) = \frac{p_{ik}}{p_{jk}} = \frac{F(w_i^T w_k)}{F(w_j^T w_k)}
  \]

- This means: \( F(w_i^T w_k) = \exp(w_i^T w_k) = p_{ik} = \frac{x_{ik}}{x_i} \). If we solve for \( w_i^T w_k = \log(p_{ik}) = \log(x_{ik}) - \log(x_i) \)

- (1) Since \( \log(x_i) \) is independent of \( k \), we can set it as a bias term \( b_i \). (2) We can also add another bias term \( b_k \) for \( w_k \): \( w_i^T w_k + b_i + b_k = \log(x_{ik}) \)

- Log is ill defined when \( x_{ik} = 0 \) (a simple fix is to change \( \log(x_{ik}) \) to \( \log(x_{ik} + 1) \))

- One problem with the above objective function: it weights co-occurrence equally.
weighted least squares

From previous, we have $w_i^T w_k + b_i + b_k = \log(X_{ik})$

The author proposes the following objective function:

$J = \sum_{i,j=1}^{V} f(X_{ij}) \left( w_i^T w_k + b_i + b_k - \log(X_{ik}) \right)^2$

- We want $f(0) = 0$: 0 weight for zero elements in the matrix.
- $f(x)$ to be non-decreasing: more weight for high co-occurrence
- $f(x)$ to be relatively small for large values of $x$: frequent co-occurrence are not over-weighted.

A lot functions can satisfy above properties for $f(x)$, in the paper they used:

$f(x) = \begin{cases} 
(x/x_{\text{max}})^\alpha & \text{if } x < x_{\text{max}} \\
1 & \text{otherwise} 
\end{cases}$

For their experiments, they use $x_{\text{max}} = 100$, and $\alpha = 3/4$
Relationship to Skip-Gram

GloVe Objective: \( J = \sum_{i,j=1}^{V} f(X_{ij}) \left( w_i^T w_k + b_i + b_k - \log(X_{ik}) \right)^2 \)

- Let \( Q_{ij} \) in Skip-Gram be a softmax function: \( Q_{ij} = \frac{\exp(w_i^T w_j)}{\sum_k \exp(w_i^T w_k)} \)
- The objective function is to maximize the log probability: \( J = -\sum_{i \in \text{corpus}} \sum_{j \in \text{context}(i)} \log Q_{ij} \)
- We can group terms that have the same values: \( J = -\sum_{i=1}^{V} \sum_{j=1}^{V} X_{ij} \log Q_{ij} \)
  - Again: \( X_{ij} \) is an element in the co-occurrence matrix \( X \), \( X_i = \sum_k X_{ik} \), \( P_{ij} = P(j|i) = X_{ij} / X_i \)
  - Rewrite as: \( J = -\sum_{i=1}^{V} X_i \sum_{j=1}^{V} P_{ij} \log Q_{ij} = \sum_{i=1}^{V} X_i H(P_i, Q_i) \), where \( H(P_i, Q_i) \) is the cross entropy of the distributions \( P_i \) and \( Q_i \)
  - Rewrite again with least square measure: \( J = \sum_{i=1}^{V} X_i H(P_i, Q_i) \approx \sum_{i,j} X_i (P_{ij} - Q_{ij})^2 \approx \sum_{i,j} X_i (\log P_{ij} - \log Q_{ij})^2 = \sum_{i,j} X_i (w_i^T w_j - \log X_{ij})^2 \)
  - Replace \( X_i \) with a weight function \( f(X_{ij}) \): \( J = \sum_{i,j} f(X_{ij}) (w_i^T w_j - \log X_{ij})^2 \)
Complexity of the Model

GloVe Objective: \( J = \sum_{i,j=1}^{V} f(X_{ij}) \left( w_i^T w_k + b_i + b_k - \log(X_{ik}) \right)^2 \)

- GloVe Computational Complexity: \( \text{nnz}(X) \), or No worse than \( O(|V|^2) \).
  - \( V \) could be huge!

- Assume the number of co-occurrence of \( X_{ij} \) can be modeled as a power-law function of the frequency work pair rank \( r_{ij} \): \( X_{ij} = \frac{k}{(r_{ij})^\alpha} \)

- For the corpora they used in the paper, the frequencies can be modeled with \( \alpha = 1.25 \). This is roughly \( O(|C|^{0.8}) \).
  - Window Based model: scales with the corpus size \( O(|C|) \).
Results on Word Analogy

- Word analogy task: \(a\) is to \(b\) as \(c\) is to ? \(->\) \textbf{man} is to \textbf{king} as \textbf{woman} is to ?
- Model the problem as: which word \(d\) \(w_d\) is closest to \(w_b - w_a + w_c\) by similarity metric (cosine).
- Underlined scores are best within groups of similarly-sized models.
- Bold scores are best overall.
- Size Scalability: can be trained on 42 billion token corpus.
- Performance Scalability: increasing corpus size improves GloVe
  - Not necessary true for other corpus. Example: SVD-L decreases.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ivLBL</td>
<td>100</td>
<td>1.5B</td>
<td>55.9</td>
<td>50.1</td>
<td>53.2</td>
</tr>
<tr>
<td>HPCA</td>
<td>100</td>
<td>1.6B</td>
<td>4.2</td>
<td>16.4</td>
<td>10.8</td>
</tr>
<tr>
<td>GloVe</td>
<td>100</td>
<td>1.6B</td>
<td>67.5</td>
<td>54.3</td>
<td>60.3</td>
</tr>
<tr>
<td>SG</td>
<td>300</td>
<td>1B</td>
<td>61</td>
<td>61</td>
<td>61</td>
</tr>
<tr>
<td>CBOW</td>
<td>300</td>
<td>1.6B</td>
<td>16.1</td>
<td>52.6</td>
<td>36.1</td>
</tr>
<tr>
<td>viLBL</td>
<td>300</td>
<td>1.5B</td>
<td>54.2</td>
<td>64.8</td>
<td>60.0</td>
</tr>
<tr>
<td>ivLBL</td>
<td>300</td>
<td>1.5B</td>
<td>65.2</td>
<td>63.0</td>
<td>64.0</td>
</tr>
<tr>
<td>GloVe</td>
<td>300</td>
<td>1.6B</td>
<td>80.8</td>
<td>61.5</td>
<td>70.3</td>
</tr>
<tr>
<td>SVD</td>
<td>300</td>
<td>6B</td>
<td>6.3</td>
<td>8.1</td>
<td>7.3</td>
</tr>
<tr>
<td>SVD-S</td>
<td>300</td>
<td>6B</td>
<td>36.7</td>
<td>46.6</td>
<td>42.1</td>
</tr>
<tr>
<td>SVD-L</td>
<td>300</td>
<td>6B</td>
<td>56.6</td>
<td>63.0</td>
<td>60.1</td>
</tr>
<tr>
<td>CBOW†</td>
<td>300</td>
<td>6B</td>
<td>63.6</td>
<td>67.4</td>
<td>65.7</td>
</tr>
<tr>
<td>SG†</td>
<td>300</td>
<td>6B</td>
<td>73.0</td>
<td>66.0</td>
<td>69.1</td>
</tr>
<tr>
<td>GloVe</td>
<td>300</td>
<td>6B</td>
<td>77.4</td>
<td>67.0</td>
<td>71.7</td>
</tr>
<tr>
<td>CBOW</td>
<td>1000</td>
<td>6B</td>
<td>57.3</td>
<td>68.9</td>
<td>63.7</td>
</tr>
<tr>
<td>SG</td>
<td>1000</td>
<td>6B</td>
<td>66.1</td>
<td>65.1</td>
<td>65.6</td>
</tr>
<tr>
<td>SVD-L</td>
<td>300</td>
<td>42B</td>
<td>38.4</td>
<td>58.2</td>
<td>49.2</td>
</tr>
<tr>
<td>GloVe</td>
<td>300</td>
<td>42B</td>
<td>81.9</td>
<td>69.3</td>
<td>75.0</td>
</tr>
</tbody>
</table>
# Results on Word Similarity

- All vectors are 300 dimension
- Compute Cosine Similarity, and Use Spearman’s rank correlation coefficient between this score and human judgment.
- GloVe outperforms CBOW* while using 42B tokens.

<table>
<thead>
<tr>
<th>Model</th>
<th>Size</th>
<th>WS353</th>
<th>MC</th>
<th>RG</th>
<th>SCWS</th>
<th>RW</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVD</td>
<td>6B</td>
<td>35.3</td>
<td>35.1</td>
<td>42.5</td>
<td>38.3</td>
<td>25.6</td>
</tr>
<tr>
<td>SVD-S</td>
<td>6B</td>
<td>56.5</td>
<td>71.5</td>
<td>71.0</td>
<td>53.6</td>
<td>34.7</td>
</tr>
<tr>
<td>SVD-L</td>
<td>6B</td>
<td>65.7</td>
<td>72.7</td>
<td>75.1</td>
<td>56.5</td>
<td>37.0</td>
</tr>
<tr>
<td>CBOW†</td>
<td>6B</td>
<td>57.2</td>
<td>65.6</td>
<td>68.2</td>
<td>57.0</td>
<td>32.5</td>
</tr>
<tr>
<td>SG†</td>
<td>6B</td>
<td>62.8</td>
<td>65.2</td>
<td>69.7</td>
<td>58.1</td>
<td>37.2</td>
</tr>
<tr>
<td>GloVe</td>
<td>6B</td>
<td>65.8</td>
<td>72.7</td>
<td>77.8</td>
<td>53.9</td>
<td>38.1</td>
</tr>
<tr>
<td>SVD-L</td>
<td>42B</td>
<td>74.0</td>
<td>76.4</td>
<td>74.1</td>
<td>58.3</td>
<td>39.9</td>
</tr>
<tr>
<td>GloVe</td>
<td>42B</td>
<td><strong>75.9</strong></td>
<td><strong>83.6</strong></td>
<td><strong>82.9</strong></td>
<td><strong>59.6</strong></td>
<td><strong>47.8</strong></td>
</tr>
<tr>
<td>CBOW*</td>
<td>100B</td>
<td>68.4</td>
<td>79.6</td>
<td>75.4</td>
<td>59.4</td>
<td>45.5</td>
</tr>
</tbody>
</table>
Results on NER Task

- Used as features to CRF-based model.
- GloVe model outperforms all other methods except for the CoNLL test set.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dev</th>
<th>Test</th>
<th>ACE</th>
<th>MUC7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete</td>
<td>91.0</td>
<td>85.4</td>
<td>77.4</td>
<td>73.4</td>
</tr>
<tr>
<td>SVD</td>
<td>90.8</td>
<td>85.7</td>
<td>77.3</td>
<td>73.7</td>
</tr>
<tr>
<td>SVD-S</td>
<td>91.0</td>
<td>85.5</td>
<td>77.6</td>
<td>74.3</td>
</tr>
<tr>
<td>SVD-L</td>
<td>90.5</td>
<td>84.8</td>
<td>73.6</td>
<td>71.5</td>
</tr>
<tr>
<td>HPCA</td>
<td>92.6</td>
<td><strong>88.7</strong></td>
<td>81.7</td>
<td>80.7</td>
</tr>
<tr>
<td>HSMN</td>
<td>90.5</td>
<td>85.7</td>
<td>78.7</td>
<td>74.7</td>
</tr>
<tr>
<td>CW</td>
<td>92.2</td>
<td>87.4</td>
<td>81.7</td>
<td>80.2</td>
</tr>
<tr>
<td>CBOW</td>
<td>93.1</td>
<td>88.2</td>
<td>82.2</td>
<td>81.1</td>
</tr>
<tr>
<td>GloVe</td>
<td><strong>93.2</strong></td>
<td><strong>88.3</strong></td>
<td><strong>82.9</strong></td>
<td><strong>82.2</strong></td>
</tr>
</tbody>
</table>
Results on Vector Dim and Context Size

- Trained on 6 billion token corpus.
- (a) the window size is 10. (b) and (c) the vector size is 100
- Symmetric: context window left + right. Asymmetric: only left.
- Small window size: syntactic is better. Long window size: semantic is better.

(a) Symmetric context  (b) Symmetric context  (c) Asymmetric context
Results on Corpus Size

- Vector dimension 300.
- Syntactic subtask: monotonic increases in performance as the corpus size increases
  - Large corpus produces better statistics.

![Chart showing accuracy for different corpus sizes](chart.png)
Results on Runtime (Iterations)

- Vector dimension is 300, 6B token corpus, vocabulary size 400,000, and window size 10
- Learning cuves:

![Training Time (hrs) and Accuracy (%) graphs for GloVe vs CBOw and GloVe vs Skip-Gram](image_url)
Conclusion

◊ The paper shows that GloVe outperforms other methods on different experiments.

◊ However, as the math shown previously: all these models share some commonalities and only differ in weight functions, loss functions, and training time.

◊ There is many parameters that can have an impact on word2vec.

  ◊ As the author points out that it’s possible that parameters in word2vec is not tuned to be optimal since there is so many parameters while GloVe is almost optimal in parameters.

  ◊ For example, word2vec code they used is only designed for a single epoch for its study while GloVe is trained over many iterations for the LS problem.
Reference
