Improving Distributional Similarity with Lessons Learned from Word Embeddings

Omer Levy, Yoav Goldberg, Ido Dagan

Presented by: **Ajay Sohmshetty**

Word Representation Methods

Count-based distributional models

Neural network-based models

Word Representation Methods

Count-based distributional models

- SVD (Singular Value Decomposition)
- PPMI (Positive Pointwise Mutual Information)

Neural network-based models

- SGNS (Skip-Gram Negative Sampling) / CBOW
- GloVe

Word Representation Methods

Count-based distributional models

- SVD (Singular Value Decomposition)
- PPMI (Positive Pointwise Mutual Information)

Neural network-based models

- SGNS (Skip-Gram Negative Sampling) / CBOW
- GloVe

Conventional wisdom:
Neural-network based models > Count-based models

Word Representation Methods

Count-based distributional models

- SVD (Singular Value Decomposition)
- PPMI (Positive Pointwise Mutual Information)

Neural network-based models

- SGNS (Skip-Gram Negative Sampling) / CBOW
- GloVe

Levy et. al.:
Hyperparameters and system design choices more important, not the embedding algorithms themselves.

Hyperparameters in Skip-Gram

\# of negative samples

$$J_t(\theta) = \log \sigma\left(u_o^T v_c\right) + \sum_{i=1}^{k} \mathbb{E}_{j \sim P(w)}\left[\log \sigma\left(-u_j^T v_c\right)\right]$$

$$P(w) = U(w)^{3/4} / Z$$

Unigram distribution smoothing exponent

➔ These can be transferred over to the count-based variants.

Context Distribution Smoothing

$$PMI_\alpha(w,c) = \log \frac{\hat{P}(w,c)}{\hat{P}(w)\hat{P}_\alpha(c)}$$

$$\hat{P}_\alpha(c) = \frac{\#(c)^\alpha}{\sum_c \#(c)^\alpha}$$

Shifted PMI

$$SPPMI(w,c) = \max\left(PMI\left(w,c\right) - \log k, 0\right)$$

All Transferable Hyperparameters

	Hyperparameter	**Explored Values**	**Applicable Methods**
	Window	2, 5, 10	All
Preprocessing	Dynamic Context Window	None, with	All
	Subsampling	None, dirty, clean	All
	Deleting Rare Words	None, with	All
Association Metric	Shifted PMI	1, 5, 15	PPMI, SVD, SGNS
	Context Distribution Smoothing	1, 0.75	PPMI, SVD, SGNS
Postprocessing	Adding Context Vectors	Only w, w+c	SVD, SGNS, GloVe
	Eigenvalue Weighting	0, 0.5, 1	SVD
	Vector Normalization	None, row, col, both	All

Results

win	Method	Word Similarity Tasks: WordSim Similarity	WordSim Relatedness	Bruni et al. MEN	Radinsky et al. M. Turk	Luong et al. Rare Words	Hill et al. SimLex	Analogy Tasks: Google Add / Mul	MSR Add / Mul
2	PPMI	.732	**.699**	.744	.654	.457	.382	.552 / .677	.306 / .535
	SVD	.772	.671	**.777**	.647	**.508**	.425	.554 / .591	.408 / .468
	SGNS	**.789**	.675	.773	**.661**	.449	**.433**	.676 / **.689**	.617 / **.644**
	GloVe	.720	.605	.728	.606	.389	.388	.649 / .666	.540 / .591
5	PPMI	.732	**.706**	.738	**.668**	.442	.360	.518 / .649	.277 / .467
	SVD	.764	.679	**.776**	.639	**.499**	**.416**	.532 / .569	.369 / .424
	SGNS	**.772**	.690	.772	.663	.454	.403	.692 / **.714**	.605 / **.645**
	GloVe	.745	.617	.746	.631	.416	.389	.700 / .712	.541 / .599
10	PPMI	.735	**.701**	.741	.663	.235	.336	.532 / .605	.249 / .353
	SVD	.766	.681	.770	.628	**.312**	.419	.526 / .562	.356 / .406
	SGNS	**.794**	.700	**.775**	**.678**	.281	**.422**	.694 / .710	.520 / **.557**
	GloVe	.746	.643	.754	.616	.266	.375	.702 / **.712**	.463 / .519

Key Takeaways

- This paper challenges the conventional wisdom that neural network-based models are superior to count-based models.

- While model design is important, hyperparameters are also KEY for achieving reasonable results. Don’t discount their importance!

- Challenge the status quo!