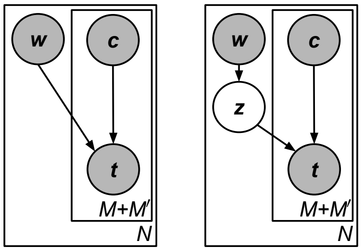
Embedding Words as Distributions with a Bayesian Skip-gram Model

Arthur Bražinskas **Serhii Havrylov** **Ivan Titov**
University of Amsterdam
`a.brazinskas@stud.uva.nl, {s.havrylov, titov}@uva.nl`

1 Introduction

Distributed representations induced from large unlabeled text collections have had a large impact on many natural language processing (NLP) applications, providing an effective and simple way of dealing with data sparsity. Word embedding methods [1, 2, 3, 4] typically represent words as vectors in a low-dimensional space. In contrast, we encode them as probability densities. Intuitively, the densities will represent the distributions over possible ‘meanings’ of the word. Representing a word as a distribution has many attractive properties. For example, this lets us encode generality of terms (e.g., ‘animal’ is a hypernym of ‘dog’), characterize uncertainty about their meaning (e.g., a proper noun, such as ’John’, encodes little about the person it refers to) or represent polysemy (e.g., ’tip’ may refer to a gratuity or a sharp edge of an object). Capturing entailment (e.g., ‘run’ entails ’move’) is especially important as it needs to be explicitly or implicitly accounted for in many NLP applications (e.g., question answering or summarization). Intuitively, distributions provide a natural way of encoding entailment: the entailment decision can be made by testing the level sets of the distributions for ‘soft inclusion’(e.g., using the KL divergence [5]).

2 Model and Inference



(a) (b)

for each position n in the collection:
  $z^n \sim \mathcal{N}(z; \mu_{w^n}, \sigma^2_{w^n}\mathrm{I})$ [draw a vector for word w^n]
  for $k = 1$ to $M + M'$ [go over context words]
    $t^n_k \sim Bernoulli(\sigma(u^T_{c^n_k} z^n))$ [decide if a true context or not]

(c)

Figure 1: Graphical models for skip-gram (a) and BSG (b); the generative story for BSG (c)

We extend the negative sampling version of the skip-gram model (SG) [3]. As with SG, for each word in the corpus w^n, we are given a vector of context words $c^n = (c^n_1, \ldots, c^n_{M+M'})$, where M words originate from the sliding window around w^n and the remaining M' words are sampled randomly (i.e. negative samples). The model is trained to discriminate between true and negative samples, with the origin of each context word recorded in the binary vector $t^n = (t^n_1, \ldots, t^n_{M+M'})$, $t^n_i = -1$ and $+1$ for negative and true samples, respectively. The graphical model for the standard SG model is shown in Figure 1a.

Unlike SG, our proposed model (Bayesian skip-gram, BSG) represents words as probability distributions by introducing a latent variable $z^n \in \mathbb{R}^d$ for each context window. The vector z^n is drawn from a parameterized word-specific prior distribution $p(z^n|w^n) = \mathcal{N}(z; \mu_{w^n}, \sigma^2_{w^n}\mathrm{I})$. This data-dependent prior then serves as an embedding of that word. Gaussian priors are not very suitable for modeling polysemy, we leave more sophisticated priors for future work. Our generative story and

Workshop on Bayesian Deep Learning, NIPS 2016, Barcelona, Spain.

the corresponding graphical model are shown in Figure 1c and 1b ; $u_{c_k^n} \in \mathbb{R}^d$ is an embedding of context word c_k^n and $\sigma(\cdot)$ stands for the logistic sigmoid function.

Though our model is similar to the standard Bayesian matrix factorization (BMF) model [6], there is a significant difference. Unlike BSG, BMF assumes that the same latent vector is used across all contexts of the word in a text collection. So the posteriors will encode uncertainty about the representation (e.g., the variance will be high if the word is infrequent) rather than the generality and degree of polysemy of the term (i.e. whether it is used in diverging contexts). In contract, by drawing a word vector for each sliding window in a collection, we encode the intuition that word meaning varies across contexts. Our model can be regarded as a form of 'multiple' Bayesian matrix factorization [7], where each sliding window is factorized individually but priors for words are shared across all the windows.

Despite the fact that proposed model can be formulated in a 'classic' variational inference settings, likelihood potentials have Bernoulli distributions and are no longer conjugate to Gaussian distribution. We can use neither closed-form solutions [8] nor simple VBMF schemas [9]. Instead, we rely on the variational autoencoding framework [10], resulting in an efficient embedding method which is only marginally slower than a comparable implementation of the standard Skip-gram model [3] (see Table 1). The variational lower bound on the marginal likelihood of datapoint n can be written as:

$$\log p_\theta(t^n|w^n, c^n) \geq \mathbb{E}_{q_\phi(z^n|w^n,c^n,t^n)}\left[p_\theta(t^n|z^n, c^n)\right] - D_{KL}\left(q_\phi(z^n|w^n, c^n, t^n) \| p_\theta(z^n|w^n)\right) \quad (1)$$

The KL-divergence can be integrated analytically, and we use the reparameterization trick [10] to obtain a low-variance gradient estimator for the encoder parameters. The approximate posterior distribution is a multivariate Gaussian with a diagonal covariance $q_\phi(z^n|w^n, c^n, t^n) = \mathcal{N}(z; \mu, \sigma^2\mathrm{I})$ where $\mu = W_1 h + b_1$, $\log \sigma^2 = W_2 h + b_2$, $h = f(v_{w^n} + \frac{1}{M+M'} \sum_{k=1}^{M+M'} t_k^n v_{c_k^n})$. $v_w \in \mathbb{R}^{d'}$ are (yet another) word embeddings, $W_1, W_2 \in \mathbb{R}^{d \times d'}$, $b_1, b_2 \in \mathbb{R}^d$ are parameters of the affine transformation, and f is an activation function.

3 Experiments and Discussion

There has been very little work on embedding words as distributions, or, in general, on unsupervised learning of embeddings specifically suited for entailment. One notable exception is Gaussian embeddings [5]. Similarly to us, they do not assume having one latent vector for each word. Unlike our generative modeling approach, they directly minimize similarities between densities corresponding to co-occurring words. Prior work on Bayesian modeling of embeddings has been limited to using the BMF formulation [11, 12, 13]. In order to assess the quality of induced word representations, we compare them to Gaussian embeddings on the standard textual entailment benchmark [14]. Though our focus is on learning representations suited for capturing lexical entailment, we also test them on similarity [15] and analogical reasoning benchmarks [3]. As we do not need covariance information for these two benchmarks, we use here SG as another baseline.

The Gaussian embeddings were estimated with diagonal covariance and KL as the energy function. Sub-tests of the similarity benchmark were combined into one file and the Spearman's rank correlation coefficient has been computed for all pairs at once. Finally, entailment evaluation has been performed in the same way as in Vilnis and McCallum [5]. We used the same hyperparameter selection procedure for all three models. For training, we used the one billion words corpus [16]. Larger and more varied collections are known to yield better results, we leave this for future work.

Model	Entailment (F1)	Similarity(ρ)	Semantic acc.(%)	Syntactic acc.(%)	Hours
Gauss [5]	0.69	**0.366**	32.2	38.0	**3.5**
SG [3]	-	0.352	**53.5**	**51.5**	4.5
BSG	**0.73**	0.357	52.0	41.1	5.3

Table 1: Evaluation results with 100 dimensional word embeddings.

The results of training different word-embeddings are presented in Table 1. As one can see, BSG outperforms Gaussian embeddings on the entailment benchmark. In general, BSG embeddings perform competitively on all benchmarks, including significant improvements over Gaussian embeddings on the semantic analogical reasoning tasks. However, BSG embeddings slightly under-performs on the similarity benchmark in comparison to Gaussian embeddings, and on analogical reasoning in comparison to SG. It is worth mentioning that words are embedded as probability distributions in

Euclidean space. Meanwhile, a probability of being a word from the sliding window is defined by a dot product. In this setup variance of the probability of being a word from the true context can be inconsistent with word representation variance. Hence model could be improved by removing this mismatch. The results are promising, and we expect further improvements with more powerful inference networks and priors.

Acknowledgments

This project is supported by SAP and ERC Starting Grant BroadSem (678254).

References

[1] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[2] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*, 2008.

[3] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *ICLR*, 2013.

[4] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[5] Luke Vilnis and Andrew McCallum. Word representations via Gaussian embedding. *ICLR*, 2015.

[6] Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *ICML*, 2008.

[7] Koh Takeuchi, Ryota Tomioka, Katsuhiko Ishiguro, Akisato Kimura, and Hiroshi Sawada. Non-negative multiple tensor factorization. In *ICDM*, 2013.

[8] Shinichi Nakajima and Masashi Sugiyama. Theorectical analysis of Bayesian matrix factorization. *JMLR*, 2011.

[9] Yew Jin Lim and Yee Whye Teh. Variational Bayesian approach to movie rating prediction. In *Proceedings of KDD cup and workshop*, volume 7, 2007.

[10] Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *ICLR*, 2014.

[11] Jingwei Zhang, Jeremy Salwen, Michael R Glass, and Alfio Massimiliano Gliozzo. Word semantic representations using Bayesian probabilistic tensor factorization. In *EMNLP*, 2014.

[12] Joseph H. Sakaya. Scalable Bayesian induction of word embeddings. *Thesis, University of Helsinki*, 2015.

[13] Oren Barkan. Bayesian neural word embedding. *arXiv preprint arXiv:1603.06571*, 2016.

[14] Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. Entailment above the word level in distributional semantics. In *EACL*, 2012.

[15] Manaal Faruqui and Chris Dyer. Community evaluation and exchange of word vectors at wordvectors.org. In *ACL*, 2014.

[16] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013.