Adversarial Contrastive Estimation

Avishek Bose* **Yanshuai Cao***
Borealis AI

Abstract

Learning by contrasting positive and negative samples is a general strategy adopted by many methods such as noise contrastive estimation, max margin estimation, contrastive divergence, etc. It has shown to be an effective way of solving difficult problems with intractable computational requirements. In this paper, we view contrastive learning as an abstraction of all such methods, and propose to adversarially learn the negative example sampling distribution, resulting in adaptive and harder negative examples, which forces the model to learn much better representation of the data. We evaluate our proposal on learning word embeddings under different models and observe both significant performance gains in rare word embeddings and improved semantic relevance in larger neighborhoods of common words.

1 Introduction

Many models learn by comparing losses on positive examples with those on negative examples. In unsupervised learning, positive examples are observed data and negative examples could be from anywhere else. In supervised learning, positive examples consist of observed input-output pairs, while negative examples could be any paired configurations that are not observed in real data. With only easy negative examples, learning does not force the model to find critical characteristics of "good configurations" [14], leading to lack of discriminativeness in supervised setting and lack of sharpness in unsupervised learning. Generative adversarial networks (GANs) learn by having two neural networks playing a minimax game where a discriminator, D, tries to tell apart samples generated by a generator, G, from real data [4]. In this work, we propose to apply the adversarial principle to supervised contrastive learning scenarios. Instead of generating negative examples from noise, we adversarially learn a conditional sampling distribution over observed output data to form negative (unobserved) input-output pairs.

Contrastive learning approaches include max margin estimation used for SVM [15], structural and latent structural svm [13], [18], as well as triplet loss based supervised metric learning [6]; contrastive divergence used for learning undirected graphical models such as RBM; Noise Contrastive Estimation (NCE) [5] and Negative Sampling [3] used as computationally efficient replacements for full softmax. From the perspective of energy based models, learning needs to push energy down on positive samples, while pushing up at any other points in space [8]. In all of these cases, positive examples are observed and hence losses and gradient of losses can be evaluated easily on them, but negative examples are not. Different estimation methods make different choices about where to evaluate negative example losses.

We propose to adversarially learn an adaptive negative sampling distribution, to be used inside a contrastive learning framework, henceforth referred to as Adversarial Contrastive Estimation (ACE). In the max margin estimation framework, our method can be viewed as an alternative to the typical loss augmented MAP inference, especially applicable in nonconvex settings when deep neural networks are used as scoring functions. In Noise Contrastive Estimation and Negative Sampling, we show that augmenting the original fixed sampling noise to a mixture distribution with the additional component being adversarially learned greatly improves word embedding models [16].

*{avishek.bose,yanshuai.cao}@rbc.com

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

2 Learning by contrasting positive and negative examples

We now describe the adversarial contrastive estimation (ACE) method. In the most general form, we are interested in the following value function:

$$V(\omega, \theta) = \Big\langle < e(f_\omega(x), y) >_{P^+(y|x)} - < \tilde{e}(f_\omega(x), y) >_{P^-_\theta(y|x)} \Big\rangle_{P^+(x)} + R(w, \theta) \tag{1}$$

where $f_\omega(x)$ is our discriminative model parametrized by w that takes input x and produces output y; e and $\tilde{e}$ are the energy/losses penalizing deviation of prediction $f_\omega(x)$ against some target value of y; $< . >$ denotes expectation and P^+ denotes positive sampling distribution (marginal or conditional); $P^-_\theta(y|x)$ is the conditional negative sampling distribution, parametrized by θ; and $R(w, \theta)$ is an additional optional regularization term. Note that e and $\tilde{e}$ might or might not be the same, but are related for all useful learning schemes that we will discuss later.

To learn by ACE, we perform alternating updates to simulate a minimax game $\min_\omega \max_\theta V(w, \theta)$, like in generative adversarial nets (GANs). The important distinction is that instead of having a generator creating negative samples from noise, as in typical GANs, our $P^-_\theta(y|x)$ defines a sampling distribution over a set of values for y, with the set being either the universe of possible y values or all y values observed in a training set. The discrete sampling space limits application of end-to-end gradient learning as in GANs. Instead, one can use the gumbel-softmax trick [7]. In case of large number of discrete choices, like the word embedding problem that we will demonstrate, due to computational scalability issues, one can use REINFORCE gradient estimator [17] or some variance reduction version thereof. An advantage of discrete sampling space is that we can let $P^-_\theta(y|x)$ be a mixture distribution of a fixed distribution and an adversarially learned adaptive one, leading to much more stale and easy optimization. Furthermore, the adaptive distribution can still benefit from data sampled by the fixed distribution via importance reweighting.

As examples, we will demonstrate next how max margin estimation and noise contrastive estimation can be interpreted in this framework. Supervised metric learning via triplet ranking loss and contrastive divergence can be similarly framed, but we will omit due to space limitation.

2.1 Max margin estimation

Classical methods for structural prediction tasks using the structural SVM, as well as similarly motivated supervised metric learning method with triplet ranking loss fall into the category of max margin estimation. We will show that the setup of max magin estimation can be viewed as a special case of ACE, although learning is done in completely different ways as typical max margin estimation with shallow models can be framed as convex optimizations.

Typical max margin estimation optimizes the following objective:

$$min_\omega \|\omega\|^2 + C \sum_i^n \max_{\hat{y}} \{\Delta(y_i, \hat{y}) + w^\top \Phi(x_i, \hat{y}) - w^\top \Phi(x_i, y_i)\} \tag{2}$$

where Δ defines penalty on structured output differences; Φ is a joint feature extractor on x and y.

Let $e(f_\omega(x_i), y_i)$ be $-w^\top \Phi(x_i, y_i)$, and $\tilde{e}(f_\omega(x_i), \hat{y})$ to be $-\Delta(y_i, \hat{y}) - w^\top \Phi(x_i, \hat{y})$. Then by grouping all (x_i, y_i) pairs based on x_i, and dividing by the number of samples, the summation in 2 can be turned into expectation with respect to empirical marginal distribution of x_i, $\hat{P}^+(x)$ and empirical conditional distribution of y_i given x_i, $\hat{P}^+(y|x)$. And the overall Eq. 2 can be expressed as

$$min_\omega \|\omega\|^2 + \tilde{C} \Big\langle < e(f_\omega(x), y) >_{\hat{P}^+(y|x)} - < \tilde{e}(f_\omega(x_i), \hat{y}) >_{P^*(y|x)} \Big\rangle_{\hat{P}^+(x)} \tag{3}$$

where $P^*(y|x_i) = 1$ for $y = \text{argmax}_{\hat{y}}\ \tilde{e}(f_\omega(x_i), \hat{y})$, 0 otherwise.

Then minimax optimization of ACE value function can be recovered by letting $P^-_\theta(y|x)$ be a learnable sampling distribution, with optimal θ yielding $P^*(y|x)$ for every x. In case such an optimal $P^-_\theta(y|x)$ does not exist, minimax learning with ACE value function can be viewed as an approximation to max margin estimation, with adversarial negative sampling used as approximate inference instead of the typical loss augmented MAP inference in structural prediction. Obviously, in typical structural predicition problems loss augmented MAP inference can be carried out efficiently, there is no benefit for the ACE approach. But if Φ is a deep neural networks, ACE would potentially be better suited.

2.2 Noise contrastive estimation

In noise contrastive estimation (NCE), a binary classification decides if an input-output pair (x, y) is from real data. NCE tries to raise log likelihood of positive pairs, and lower the log likelihood of negative pairs, where negative pairs are formed by sampling $\tilde{y}$ in some fixed way and pair $\tilde{y}$ with x. NCE minimizes the following objective:

$$L = -\sum_{(x,y)\in\text{train}} \log P_\omega(z=1|x,y) + \sum_{x\in\text{train}} \sum_{\tilde{y}} \log P_\omega(z=0|x,\tilde{y}) \tag{4}$$

Let $-\log P_\omega(z=1|.,.)$ and $\log P_\omega(z=0|.,.)$ be $e(f_\omega(.),.)$ and $\tilde{e}(f_\omega(.),.)$ respectively, Eq. 4 can be rewritten in the form of Eq. 1, with the only difference being a fixed negative sampling distribution instead of an adaptive one with parameter θ.

This difference motivates us to augment the fixed NCE noise distribution to a mixture distribution $P^-(y|x) = \lambda P^-_{nce} + (1-\lambda)P^-_\theta$. In the next section, we discuss how this augmentation of NCE to ACE can be applied to improve word embedding learning.

3 Adversarial Contrastive Estimation for Word Embeddings

Word embeddings learn vector representation of words from co-occurrences in a text corpus. (See Appendix A for more details on word embeddings.) NCE casts this learning problem as a binary classification where the model tries to distinguish positive word and context pairs, from negative noise samples composed of word and false context pairs. The NCE objective in Skip-Gram ([10]) for word embeddings is then defined as follows:

$$L = -\sum_{w_t\in V} [\log P(y=1|w_t,w_c) + \sum_{c=1}^{K} \log P(y=0|w_t,\tilde{w_c})], \tag{5}$$

Here, w_c is sampled from the true set of contexts and $\tilde{w_c} \sim Q$ is sampled k times from a fixed noise distribution. [10] introduced a further simplification of NCE, called "Negative Sampling". With respect to our ACE framework, the difference between NCE and Negative Sampling is inconsequential, so we continue the discussion using NCE. The choice of noise distribution is a free parameter in NCE word embedding models but is usually chosen as a variation of the uniform distribution. A drawback of this sampling scheme is that it favors more common word as context; another drawback is that the negative context words are sampled in the same way, rather than tailored toward the actual target word. These issues cause the negative samples to be too easy. To fix this problem, we apply ACE using a mixture of distribution consisting of fixed NCE noise distribution, and an adaptive conditional distribution, trained to draw samples of context words given target words. Let (w_t, w_c) denote a positive example where both input and context words are drawn from the real data distribution and $(w_t, \tilde{w_c})$ denote a negative example where the context word is from the mixture distribution. Our ACE value function, $V(D,G)$ is then defined as:

$$< \log D(w_c|w_t) >_{P_r(w_c)} + < \log(1-D(\tilde{w_c}|w_t)) >_{P_g(\tilde{w_c}|w_t)} + < \log(1-D(\tilde{w_c}|w_t)) >_{P_{nce}(\tilde{w_c})} \tag{6}$$

where $\tilde{w_c} = G(\tilde{w_c}|z,w_t)$ with $z \sim P(z)$, and $\tilde{w_c} = P_{nce}(\tilde{w_c})$ for the NCE noise distribution. The conditional generator network requires special treatment in the case of discrete data as one cannot backpropagate through the sampling process. We update the conditional generator network using REINFORCE ([17]) where the reward is taken to be the negative of the discriminator's output. Thus the gradient update for the conditional generator, parametrized by θ, is defined as:

$$\nabla_\theta L_G = < -\nabla_\theta \log G(\tilde{w_c}|z,w_t) D(G(z|w_t)|w_t) >_{P(z)} \tag{7}$$

To stabilize learning, we first normalize the reward by subtracting the mean and dividing the standard deviation, then pass through sigmoid nonlinearity (with temperature .5) to restrict the range.

4 Word Embedding Experiments

Training from scratch

In our first experiment we test the efficacy of our approach by training 300-dimensional word embeddings, the training curves for which may be found in Appendix B, and then qualitatively evaluate them by inspecting the nearest neighbors of five selected words. We first present the five nearest neighbors to each word to show that both NCE and ACE models learn sensible embeddings. We then show that ACE embeddings have much better semantic relevance in larger neighborhood (nearest neighbor 45-50).

Table 1: Top 5 Nearest Neighbors of Words followed by Neighbors 45-50 for different Models.

	Queen	King	Computer	Man	Woman
Skip-Gram NCE Top 5	princess	prince	computers	woman	girl
	king	queen	computing	boy	man
	empress	kings	software	girl	prostitute
	pxqueen	emperor	microcomputer	stranger	person
	monarch	monarch	mainframe	person	divorcee
Skip-Gram NCE Top 45-50	sambiria	eraric	hypercard	angiomata	suitor
	phongsri	mumbere	neurotechnology	someone	nymphomaniac
	safrit	empress	lgp	bespectacled	barmaid
	mcelvoy	saxonvm	pcs	hero	redheaded
	tsarina	pretender	keystroke	clown	jew
Skip-Gram ACE Top 5	princess	prince	software	woman	girl
	prince	vi	computers	girl	herself
	elizabeth	kings	applications	tells	man
	duke	duke	computing	dead	lover
	consort	iii	hardware	boy	tells
Skip-Gram ACE Top 45-50	baron	earl	files	kid	aunt
	abbey	holy	information	told	maid
	throne	cardinal	device	revenge	wife
	marie	aragon	design	magic	lady
	victoria	princes	compatible	angry	bride

Finetuning

In this experiment we test the hypothesis of whether ACE is capable of learning better word embeddings for rare words than NCE. We take pretrained 50-dimensional Glove embeddings [12] and embeddings from [2], and finetune them using our algorithm (Refer to Appendix C and D for training curves and examples). We evaluate our performance on the Stanford Rare Word Dataset, the baseline results for C&W are taken from [9]. As can be seen from our results in Table. 2. ACE sampling does indeed improve the performance of both models suggesting that it is indeed a general strategy.

Table 2: Spearman score ($\rho * 100$) for finetuned models on Rare Word Dataset.

	Spearman Score
C&W baseline (Number taken from [9])	26.75
C&W ACE	32.86
C&W NCE + ACE	**37.69**
Glove baseline (Recomputed based on[12])	33.94
Glove NCE + ACE	**36.37**

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[3] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[5] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

[6] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer, 2015.

[7] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[8] Yann LeCun and Fu Jie Huang. Loss functions for discriminative training of energy-based models. In *AIStats*, 2005.

[9] Thang Luong, Richard Socher, and Christopher D Manning. Better word representations with recursive neural networks for morphology. In *CoNLL*, pages 104–113, 2013.

[10] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[11] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[12] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[13] Sunita Sarawagi and Rahul Gupta. Accurate max-margin training for structured output spaces. In *Proceedings of the 25th international conference on Machine learning*, pages 888–895. ACM, 2008.

[14] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–769, 2016.

[15] Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. Learning the curriculum with bayesian optimization for task-specific word representation learning. *arXiv preprint arXiv:1605.03852*, 2016.

[16] Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.

[17] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[18] Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *Proceedings of the 26th annual international conference on machine learning*, pages 1169–1176. ACM, 2009.

A Background Information

A.1 Generative Adversarial Networks

Generative Adversarial Networks is an unsupervised learning technique that has gained prominence due to its success in training Deep Generative Models. The core of GAN training involves jointly training two networks, a generator network, tasked with producing samples from some distribution that ideally mimics examples from the true data distribution, and a discriminator network, which attempts to differentiate between samples from the true data distribution and the one produced by the generator. In its most vanilla form the GAN objective can be written as a min max optimization problem with the following form:

$$\min_G \max_D < \log(D(x)) >_{P_r(x)} + < \log(1 - D(\tilde{x})) >_{P_g(\tilde{x})}, \tag{8}$$

where $\mathbb{P}_r$ represents the true data distribution and $\mathbb{P}_g$ is the model distribution which is implicitly defined by the generator. Thus a generated sample is then $\tilde{x} = G(z), z \sim P(z)$ where z is sampled from some noise distribution such as a Gaussian. It has been shown that a Discriminator trained to optimality minimizes the Jensen Shannon Divergence (JSD) but this leads to vanishing gradients as the Discriminator saturates. In practice it is common for the generator to maximize $< \log(D(\tilde{x})) >_{P_g(\tilde{x})}$ [4]. If both the Discriminator and Generator are conditioned on additional information such the label y then it has been shown that the quality of generated samples significantly improves. It is important to note that there are other iterations of GANs that do not in fact minimize the JSD but other metrics such as the Wasserstein distance [1]. In this work, we only consider the JSD GAN and its conditional variant [11].

A.2 Word Embeddings

Continuous Vector Space representation of Words, also known as Word Embeddings, learned from Large Datasets have been shown to capture syntactic and semantic meaning between words and improve performance on downstream tasks. For example, with well trained Word Embeddings one can compute the following relationship of $king - man + woman \approx queen$. Consequently, there have been many algorithms proposed to efficiently learn Word Embeddings, in this work we focus on the Skip-Gram variant which is arguably the most popular Word Embedding model [10]. At a high level given an input center word the Skip-Gram model tries to maximize the log probability of predicting all of its surrounding words within a fixed context window. Formally, given a sequence of training words $w_1, w_2, w_3, ..., w_T$, the objective of the Skip-gram model is to maximize the average log probability

$$\frac{1}{T} \sum_{t=1}^{T} \sum_{c=1}^{C} \log P(w_c|w_t), \tag{9}$$

where C is the context window size of the input center word w_t and w_c is a context word taken from this window. The probability over surrounding words is computed by first computing a score between the input word and a context word from the vocabulary then normalizing with a softmax.

B Supplementary Experimental details and results for Skip-Gram ACE trained from Scratch

B.1 Implementation Details

For our experiments we train all our models on the May 2017 dump of the English Wikipedia considering only unigrams and performing only a single pass through the dataset. Furthermore, we use the same positive subsampling scheme proposed by [10] in their seminal word2vec paper. Both our conditional discriminator and generator are modeled using the Skip-Gram architecture which is a two layer neural network with a linear mapping between the layers. The first layer is equivalent to a lookup table for the chosen center word while the second layer is a lookup table for the sampled context word. The only difference between the discriminator and generator for us is that we use a sigmoid nonlinearity after the second layer in the discriminator as the goal is to differentiate real word context pairs from fake ones. While in the conditional generator we aim to choose a word from our vocabulary and consequently we use the softmax nonlinearity which is used to categorically sample our negative context word. The Word Embeddings are taken to be the first dense matrix in the conditional discriminator. Finally, we use cosine similarity as our metric of choice as is common in the word embedding literature.

B.2 Training Plots for Skip-Gram Model Trained from Scratch

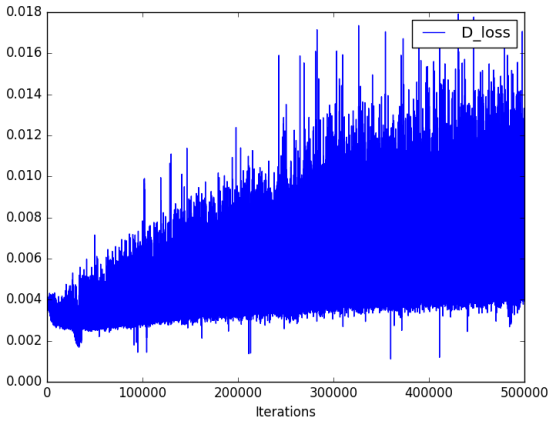

Figure 1: Discriminator Loss under ACE

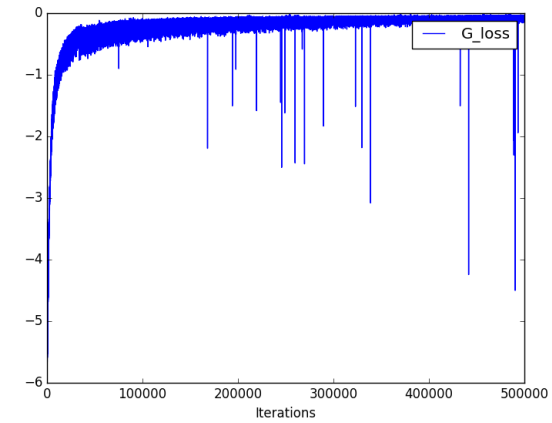

Figure 2: Generator Loss under ACE

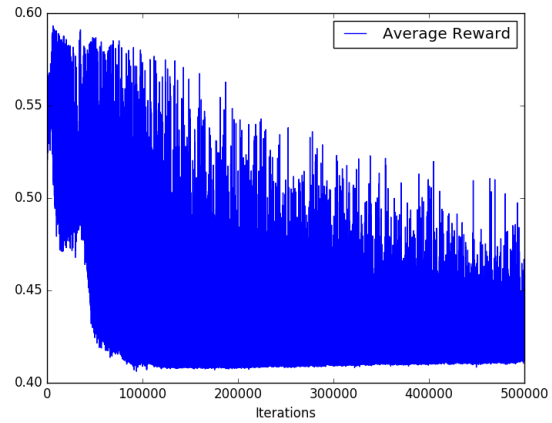

Figure 3: Average Normalized Penalty for Generator per iteration

C Training Plots for Finetuned Models

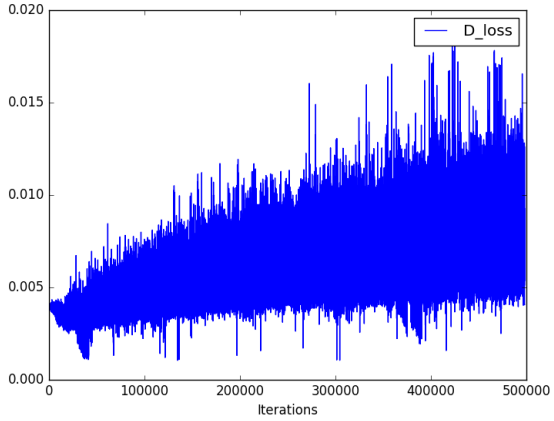

Figure 4: C&W Discriminator Loss (NCE+ACE sampling)

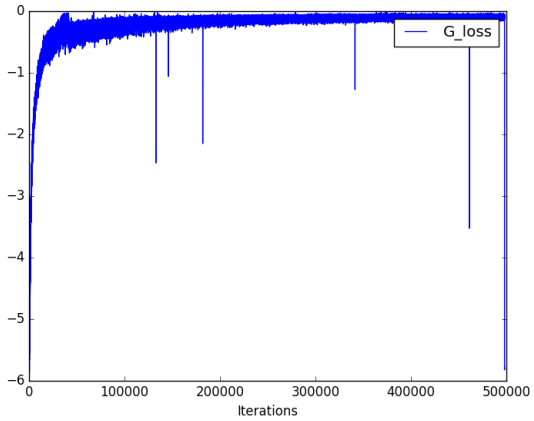

Figure 5: C &W Generator Loss

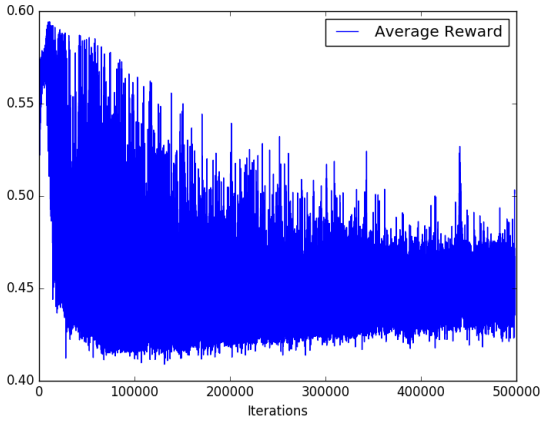

Figure 6: C&W Average Normalized Penalty for Generator per iteration

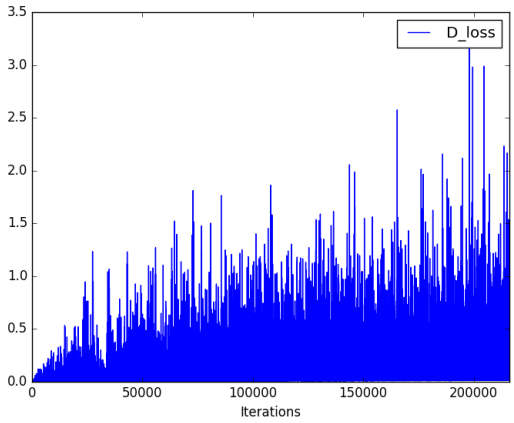

Figure 7: 50-d Glove Discriminator Loss (NCE+ACE sampling)

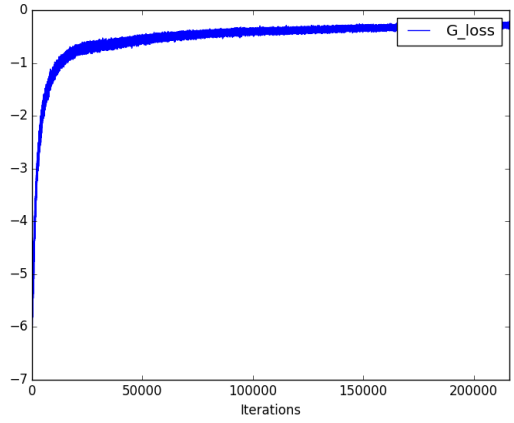

Figure 8: 50-d Glove Generator Loss

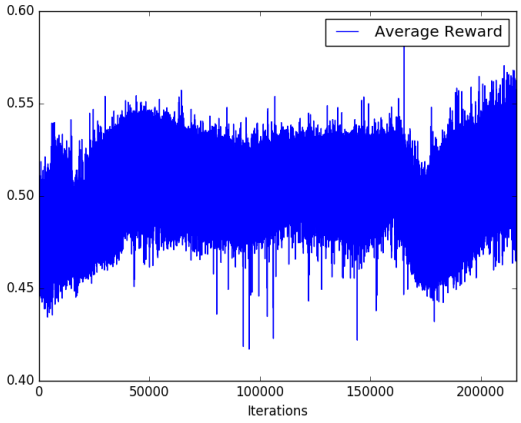

Figure 9: 50-d Glove Average Normalized Penalty for Generator per iteration

D Selected Generated Negative Samples through ACE

Here we present selected results of the negative words with the highest probability that were sampled under our conditional generator. In our setup the generator takes a center word as input and produces a corresponding context word categorically sampled based on the output probabilities of the network. The results are taken from our finetuned C&W embeddings which was our best performing model on the Rare Word dataset. The probability of generated samples were in the range of 10^{-3} to 10^{-4} for a vocabulary size of 130 thousand words, suggesting that the generator is not choosing its favorite negative sample but instead a diverse set. To emphasize this point, we present two sets of generated samples; the first set which could conceivably be seen as a positive examples taken from the real data distribution. The second set contains more random words which makes for easier negative examples.

Table 3: Top word with the highest probability as a hard negative example vs. easy negative examples also sampled by our Generator for the same word

Input Word	Hard Generated Sample	Easy Generated Samples
revolutionaries	biplanes	decorum
french	post-independence	superbike
anarchist	arsonists	jobs
secular	disgusted	codec
government	decertify	one-days
rules	participation	subcellular
federalist	collectivists	miscalculation
islamic	legalizing	taxonomists
religious	evangelical	matchday
movements	leftism	iowa
autistic	medicorp	chefs
superior	right-wingers	neoclassical
body	caress	sanctuaries
sword	maneuver	record-high
tv	technicolor	habitations