On SkipGram Word Embedding Models with Negative Sampling: Unified Framework and Impact of Noise Distributions

Ziqiao Wang[1]
joint work with Yongyi Mao[1], Hongyu Guo[2], and Richong Zhang[3]

[1]University of Ottawa [2]National Research Council Canada [3]Beihang University

November 27, 2019

Outline

Outline

Introduction

Learning the representations of words is an important task in natural language processing (NLP).

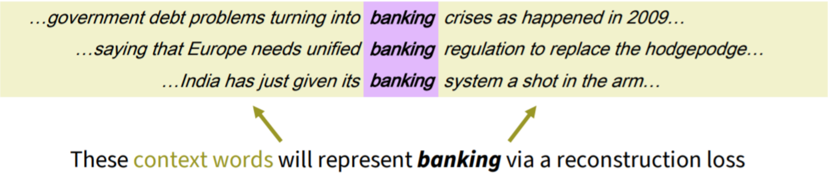
- The advances of word embedding models have enabled numerous successes in NLP applications.
- In the past years, when building a machine learning model for NLP, starting from a pretrained word embedding dictionary has become a nearly standard practice.
- e.g., Word2Vec(SkipGram & CBOW), Glove, ELMo, GPT, BERT, GPT-2, XL-Net, ERNIE, RoBERTa, ...

Brief Introduction to SkipGram

The SkipGram models are among the first word-embedding models and have been widely used since their introduction.

1. Distributional semantics: A word's meaning is given by the words that frequently appear close-by
   - "You shall know a word by the company it keeps" (J. R. Firth 1957: 11)
   - One of the most successful ideas of modern statistical NLP
2. When a word w appears in a text, its context is the set of words that appear nearby (within a fixed-size window).
3. Use the many contexts of w to build up a representation of w

Brief Introduction to SkipGram



Brief Introduction to SkipGram with Negative Sampling (SGN)

Learning a SkipGram model may incur significant training complexity when the word vocabulary is large.
An elegant approach to by-pass this complexity is through "negative sampling".

1. In this approach, a set of word-context pairs are drawn from a "noise" distribution, under which the context is independent of the center word.
2. These "noise pairs", or "negative examples", together with the word-context pairs from the corpus, or the "positive examples", are then used to train a binary classifier that is parameterized by the word embeddings.

Unanswered questions for SGN

In this work, we ask the following questions.

- Beyond that particular distribution, if one chooses a different noise distribution, is SGN still theoretically justified?
- Is there a general principle underlying SGN that allows us to build new embedding models?
- If so, how does the noise distribution impact the training of such models and their achievable performances?

Outline

Notations

1. $\mathcal{X}$: a vocabulary of words
   $\mathcal{Y}$: a set of contexts
   - A given training corpus may be parsed into a collection $\mathcal{D}^+$ of word-context pairs (x, y) from $\mathcal{X} \times \mathcal{Y}$ (using a running window of length $2L + 1$)
2. $\mathbb{P}$: an unknown distribution on $\mathcal{X} \times \mathcal{Y}$
   $\mathbb{P}_{\mathcal{X}}$: the marginal of $\mathbb{P}$ on $\mathcal{X}$
   $\mathbb{P}_{\mathcal{Y}|x}$: the conditional distribution of Y given $X = x$ under $\mathbb{P}$.
   $\mathbb{Q}_{\mathcal{Y}|x}$: a distribution on $\mathcal{Y}$
   $\mathbb{Q}$: the *noise distribution* on $\mathcal{X} \times \mathcal{Y}$
   - Given $\mathbb{Q}$, we draw word-context pairs i.i.d. from $\mathbb{Q}$ to form a *noise sample* or *negative sample* $\mathcal{D}^-$.
3. N^+: the number of pairs in $\mathcal{D}^+$
   N^-: the number of pairs in $\mathcal{D}^-$

The Classifier-Learning Problem

A binary classification problem on samples $\mathcal{D}^+$ and $\mathcal{D}^-$:

- Objective: distinguish the word-context pairs drawn from $\mathbb{P}$ from those drawn from $\mathbb{Q}$
- U: the binary class label associated with each word-context pair ($\mathcal{D}^+$: $U = 1$, $\mathcal{D}^-$: $U = 0$)
- The classification problem is equivalent to learning the conditional distribution $p_{U|XY}(\cdot|x, y)$ from $\mathcal{D}^+$ and $\mathcal{D}^-$:

$$p_{U|XY}(1|x, y) := \sigma\left(s(x, y)\right) \tag{1}$$

where $\sigma(\cdot)$ is the logistic function and $s(\cdot)$ is the score function.

The WCC framework

Let $\overline{\mathcal{X}}$ and $\overline{\mathcal{Y}}$ be two vector spaces. Let $f : \mathcal{X} \to \overline{\mathcal{X}}$ and $g : \mathcal{Y} \to \overline{\mathcal{Y}}$ be two functions representing the embedding maps for words and contexts respectively. Let $s(x, y)$ take the form

$$s(x, y) := \textbf{score}\left(f(x), g(y)\right), \tag{2}$$

the standard cross-entropy loss for this classification problem is

$$\ell = -\sum_{(x,y)\in\mathcal{D}^+} \log\sigma\left(s(x, y)\right) - \sum_{(x,y)\in\mathcal{D}^-} \log\sigma\left(-s(x, y)\right). \tag{3}$$

and the solution is

$$(f^*, g^*) := \arg\min_{f,g} \ell(f, g) \tag{4}$$

Theoretical Properties of WCC

Let $\widetilde{\mathbb{P}}$ and $\widetilde{\mathbb{Q}}$ be the empirical word-context distributions observed in $\mathcal{D}^+$ and $\mathcal{D}^-$ respectively

- $\widetilde{\mathbb{P}}(x,y) = \frac{\#(x,y)}{N^+}$ where $\#(x,y)$ is the number of times the word-context pair (x,y) appears in $\mathcal{D}^+$, and $\widetilde{\mathbb{Q}}(x,y)$ is defined similarly.
- the distribution $\widetilde{\mathbb{Q}}$ *covers* the distribution $\widetilde{\mathbb{P}}$ if the support $\mathrm{Supp}\left(\widetilde{\mathbb{P}}\right)$ of $\widetilde{\mathbb{P}}$ is a subset of the support $\mathrm{Supp}\left(\widetilde{\mathbb{Q}}\right)$ of $\widetilde{\mathbb{Q}}$.

Theorem 1

Suppose that $\widetilde{\mathbb{Q}}$ covers $\widetilde{\mathbb{P}}$. Then the following holds.

1. *The loss ℓ, as a function of s, is convex in s.*
2. *If f and g are sufficiently expressive, then there is a unique configuration s^* of s that minimizes $\ell(s)$, and the global minimizer s^* of $\ell(s)$ is given by*

$$s^*(x,y) = \log \frac{\widetilde{\mathbb{P}}(x,y)}{\widetilde{\mathbb{Q}}(x,y)} + \log \frac{N^+}{N^-}$$

for every $(x,y) \in \mathcal{X} \times \mathcal{Y}$.

Proof sketch.

$$p_U(U=1) = \frac{N^+\widetilde{\mathbb{P}}(x,y)}{N^+\widetilde{\mathbb{P}}(x,y) + N^-\widetilde{\mathbb{Q}}(x,y)} \tag{5}$$

Recall that $\ell - H(p_U) = KL(p_U||p_{U|XY})$, where $H(p_U)$ is the entropy of p_U and $KL(p_U||p_{U|XY})$ is the Kullback-Leibler divergence between p_U and $p_{U|XY}$. To make $KL(p_U||p_{U|XY}) = 0$, we have

$$p_U(U=1) = \frac{N^+\widetilde{\mathbb{P}}(x,y)}{N^+\widetilde{\mathbb{P}}(x,y) + N^-\widetilde{\mathbb{Q}}(x,y)} = \sigma\left(s^*(x,y)\right) = \frac{1}{1+\exp(s^*(x,y))} \tag{6}$$

which indicates $s^*(x,y) = \log\frac{\widetilde{\mathbb{P}}(x,y)}{\widetilde{\mathbb{Q}}(x,y)} + \log\frac{N^+}{N^-}$. □

Corollary 1

Let $N^+ = n$ and $N^- = kn$. Suppose that $\mathbb{Q}$ covers $\mathbb{P}$, and that f and g are sufficiently expressive. Then it is possible to construct a distribution $\widehat{\mathbb{P}}$ on $\mathcal{X} \times \mathcal{Y}$ using f^*, g^*, k, and $\mathbb{Q}$ such that for every $(x, y) \in \mathcal{X} \times \mathcal{Y}$, $\widehat{\mathbb{P}}(x, y)$ converges to $\mathbb{P}(x, y)$ in probability as $n \to \infty$.

Proof sketch.

Suppose we already have f^*, g^*, k, and $\mathbb{Q}$, recall that $s^*(x,y) = \log \frac{\widetilde{\mathbb{P}}(x,y)}{\widetilde{\mathbb{Q}}(x,y)} - \log k$ and $s^*(x,y) = \langle f^*(x), g^*(y) \rangle$. We can construct $\widehat{\mathbb{P}}$ as

$$\begin{aligned}
\widehat{\mathbb{P}}(x,y) &= \underbrace{\exp\{\langle f^*(x), g^*(y)\rangle + \log k\}}_{A(x,y)} \cdot \mathbb{Q}(x,y) \\
&= A(x,y) \cdot \widetilde{\mathbb{Q}}(x,y) \cdot \frac{\mathbb{Q}(x,y)}{\widetilde{\mathbb{Q}}(x,y)} \\
&= \widetilde{\mathbb{P}}(x,y) \cdot \frac{\mathbb{Q}(x,y)}{\widetilde{\mathbb{Q}}(x,y)} \\
&= \mathbb{P}(x,y) \cdot \frac{\widetilde{\mathbb{P}}(x,y)}{\mathbb{P}(x,y)} \cdot \frac{\mathbb{Q}(x,y)}{\widetilde{\mathbb{Q}}(x,y)}
\end{aligned} \tag{7}$$

□

Lemma 1

The derivative of the loss function ℓ with respect to $s(x, y)$ is

$$\frac{\partial \ell}{\partial s(x, y)} = \sigma\left(s(x, y)\right)\left(N^{-}\widetilde{\mathbb{Q}}(x, y) - e^{-s(x,y)} N^{+}\widetilde{\mathbb{P}}(x, y)\right)$$

Outline

SGN Model

Let $\mathbb{Q}$ factorize in the following form

$$\mathbb{Q}(x,y) = \widetilde{\mathbb{P}}_{\mathcal{X}}(x)\mathbb{Q}_{\mathcal{Y}}(y) \tag{8}$$

The following result follows from Theorem 1.

Corollary 2

In an unconditional SGN model, suppose that f and g are sufficiently expressive. Let $N^+ = n$ and $N^- = kn$. Then the global minimizer of loss function (3) is given by

$$s^*(x,y) = \overline{x} \cdot \overline{y} = \log \frac{\widetilde{\mathbb{P}}(x,y)}{\widetilde{\mathbb{P}}_{\mathcal{X}}(x)\widetilde{\mathbb{Q}}_{\mathcal{Y}}(y)} - \log k \tag{9}$$

As a special case when $\widetilde{\mathbb{Q}}_{\mathcal{Y}} = \widetilde{\mathbb{P}}_{\mathcal{Y}}$, the term $\log \frac{\widetilde{\mathbb{P}}(x,y)}{\widetilde{\mathbb{P}}_{\mathcal{X}}(x)\widetilde{\mathbb{Q}}_{\mathcal{Y}}(y)}$ is the well-known "pointwise mutual information" (PMI).

SGN Model

It is natural to consider the following forms of $\mathbb{Q}_{\mathcal{Y}}$ in unconditional SGN.

1. **"uniform SGN" (ufSGN)**: Let $\mathbb{Q}_{\mathcal{Y}}$ be the discrete uniform distribution over $\mathcal{Y}$, that is, $\mathbb{Q}_{\mathcal{Y}}(y) = 1/|\mathcal{Y}|$.
2. **"unigram SGN" (ugSGN)**: Let $\mathbb{Q}_{\mathcal{Y}}$ be empirical distribution $\mathbb{P}_{\mathcal{Y}}$ of context word in the corpus, that is, $\mathbb{Q}_{\mathcal{Y}}(y) = f_y / \sum_{y \in \mathcal{Y}} f_y$, where f_y is the frequency at which the context word y has occurred in the corpus.
3. **"3/4-unigram SGN" (3/4-ugSGN)**: Let $\mathbb{Q}_{\mathcal{Y}}$ be defined by $\mathbb{Q}_{\mathcal{Y}}(y) = f_y^{3/4} / \sum_{y \in \mathcal{Y}} f_y^{3/4}$. This is precisely the noise distribution used in vanilla SGN.

Conditional SGN Model

In this case, we factorize $\mathbb{Q}$ as

$$\mathbb{Q}(x, y) = \widetilde{\mathbb{P}}_{\mathcal{X}}(x)\mathbb{Q}_{\mathcal{Y}|x}(y)$$

where $\mathbb{Q}_{\mathcal{Y}|x}(\cdot)$ varies with x. Note that such form of $\mathbb{Q}$ includes all possible distributions $\mathbb{Q}$ whose marginals $\mathbb{Q}_{\mathcal{X}}$ on the center word are the same as $\widetilde{\mathbb{P}}_{\mathcal{X}}$.

Conditional SGN Model

Remark 1

In Theorem 1 and Corollary 1, the WCC framework is justified for any choice of empirical noise distribution $\widetilde{\mathbb{Q}}$ *that covers* $\widetilde{\mathbb{P}}$. *Consider some* $(x, y) \in \mathrm{Supp}\left(\widetilde{\mathbb{Q}}\right) \setminus \mathrm{Supp}\left(\widetilde{\mathbb{P}}\right)$, *namely,* (x, y) *is "covered" by* $\widetilde{\mathbb{Q}}$ *but not by* $\widetilde{\mathbb{P}}$. *By Lemma 1, the gradient is*

$$\frac{\partial \ell}{\partial s(x,y)} = \sigma\left(s(x,y)\right) \cdot N^{-}\widetilde{\mathbb{Q}}(x,y)$$

This may result in slow training.

Conditional SGN Model

Hypothesis 1

The best $\widetilde{\mathbb{Q}}$ is the one that barely covers $\widetilde{\mathbb{P}}$, namely, equal to $\widetilde{\mathbb{P}}$.

Under this hypothesis, we wish to choose $\mathbb{Q}_{\mathcal{Y}|x}$ to be equal to, or at least to closely resemble, $\widetilde{\mathbb{P}}_{\mathcal{Y}|x}$.

Conditional Adaptive SGN (caSGN) Model

Consider a version of $\left\{\widetilde{\mathbb{Q}}^t_{\mathcal{Y}|x} : x \in \mathcal{X}\right\}$ that varies with training iteration t:

- Suppose training is such that the loss computed for a batch converges and that $\widetilde{\mathbb{Q}}^t_{\mathcal{Y}|x}$ converges to $\widetilde{\mathbb{P}}_{\mathcal{Y}|x}$ for each $x \in \mathcal{X}$.
- The empirical distribution of the noise word-context pair seen during the entire training process is then $\widehat{\mathbb{Q}}^T(x,y) = \sum_{t=1}^{T} \widetilde{\mathbb{Q}}^t_{\mathcal{Y}|x}(y)\widetilde{\mathbb{P}}_{\mathcal{X}}(x)/T$.

Under the above stated assumptions, it is easy to see that $\widehat{\mathbb{Q}}^T$ must converge to $\widetilde{\mathbb{P}}$ with increasing T.

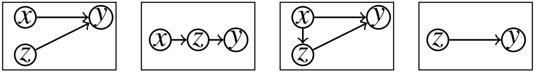
Conditional Adaptive SGN (caSGN) Model

In this case, when T is large enough, we can regard training as a version of mini-batched SGD with the noise distribution $\mathbb{Q}$ chosen as a distribution arbitrarily close to $\widetilde{\mathbb{P}}$, or a conditional SGN with $\mathbb{Q}_{\mathcal{Y}|x}$ arbitrarily close to $\widetilde{\mathbb{P}}_{\mathcal{Y}|x}$.

- This observation motivates us to design the “Conditional Adaptive SGN” (caSGN) model. The idea is to parameterize $\widetilde{\mathbb{Q}}_{\mathcal{Y}|x}$ using a neural network and force learning with mini-batched SGD to make $\widetilde{\mathbb{Q}}_{\mathcal{Y}|x}$ converge to $\widetilde{\mathbb{P}}_{\mathcal{Y}|x}$.

Conditional Adaptive SGN (caSGN) Model

Inspired by GAN, we parametrize $\widetilde{\mathbb{Q}}_{\mathcal{Y}|x}$ using an additional latent variable Z that takes value from a vector space $\mathcal{Z}$, and model Y as being *generated* from (X, Z) or simply Z:



(a) caSGN1 (b) caSGN2 (c) caSGN3 (d) aSGN

Figure 1: Generators of the adaptive SkipGram model

Since in every language the context always depends on the center word, using such Figure 1(d), $\widetilde{\mathbb{Q}}$ tend not to converge to $\widetilde{\mathbb{P}}$ by construction, except for very small training sample, to which model over-fits.

Conditional Adaptive SGN (caSGN) Model

Each of these generators can be implemented as a probabilistic neural network G (namely that the output of G is a random variable depending on its input). Then one can formulate the loss function in a way similar to GAN, e.g. in caSGN3 (Figure 1(c)),

$$\begin{aligned}\ell_{\text{caSGN3}} = -\mathbb{E}_{x\sim\widetilde{\mathbb{P}}_{\mathcal{X}}} \Big\{ & \mathbb{E}_{y\sim\widetilde{\mathbb{P}}_{\mathcal{Y}|x}} \log\sigma(s(x,y)) \\ & + \mathbb{E}_{z\sim G_{X|Z}(x), y\sim G_{Y|XZ}(x,z)} \log(-\sigma(s(x,y))) \Big\}\end{aligned} \tag{10}$$

The min-max optimization problem can be defined as

$$(f^*, g^*, G^*) := \arg\min_{f,g}\max_{G} \ell_{\text{caSGN3}}(f, g, G) \tag{11}$$

Outline

WordSim Similarity

Table 1: Spearman's ρ $(*100)$ on the word similarity tasks (text8).

Models	WS-353	WS-SIM	WS-REL	MTurk-287	MTurk-771	RW	MEN	MC	RG	SimLex
SGN	70.58	74.54	68.10	64.29	55.59	36.63	62.16	60.82	60.17	29.69
ACE	71.49	74.61	69.50	65.52	56.63	**37.85**	62.75	62.65	62.39	30.37
aSGN	71.12	74.76	68.82	**65.67**	56.47	37.58	62.63	62.36	62.36	30.49
caSGN1	71.72	**75.11**	**69.77**	65.63	56.63	37.63	**63.40**	62.54	64.18	30.36
caSGN2	**72.02**	75.05	69.64	65.44	**57.02**	37.61	63.36	**62.86**	**64.63**	**30.79**
caSGN3	71.74	74.61	69.63	65.57	56.56	37.78	62.69	62.61	62.52	30.31

Table 2: Spearman's ρ $(*100)$ on the word similarity tasks (wiki).

Models	WS-353	WS-SIM	WS-REL	MTurk-287	MTurk-771	RW	MEN	MC	RG	SimLex
SGN	67.49	74.61	61.51	63.00	59.24	39.99	68.73	64.47	69.59	31.37
ACE	71.03	**76.23**	**66.24**	63.05	60.27	40.02	67.55	76.60	70.01	31.10
aSGN	70.66	75.69	65.69	65.14	61.22	39.81	68.99	77.38	73.67	**31.58**
caSGN1	**71.56**	76.05	65.37	63.05	61.63	40.74	69.72	**80.07**	**77.58**	31.27
caSGN2	70.57	73.84	65.83	65.26	**62.28**	41.24	**70.93**	71.57	73.05	30.45
caSGN3	70.27	74.93	65.26	**65.98**	59.52	**41.55**	70.05	75.95	73.52	31.40

Word Analogy

Table 3: Accuracy on the word analogy task (text8).

Model	Semantic	Syntactic	Total
SGN	20.50	26.77	24.16
ACE	20.43	28.25	25.00
aSGN	20.84	27.86	24.94
caSGN1	21.25	**28.30**	**25.36**
caSGN2	**21.56**	27.79	25.20
caSGN3	20.43	27.76	24.71

Table 4: Accuracy on the word analogy task (wiki).

Model	Semantic	Syntactic	Total
SGN	27.28	35.52	31.77
ACE	27.62	35.30	31.81
aSGN	35.24	38.66	37.10
caSGN1	31.71	38.32	35.31
caSGN2	37.00	**39.96**	38.61
caSGN3	**41.21**	39.24	**40.14**

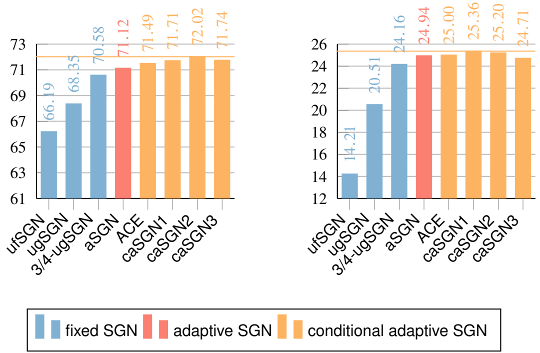


Figure 2: Left figure is Spearman's ρ $(*100)$ on WS-353 and Right figure is the total accuracy on Google Analogy

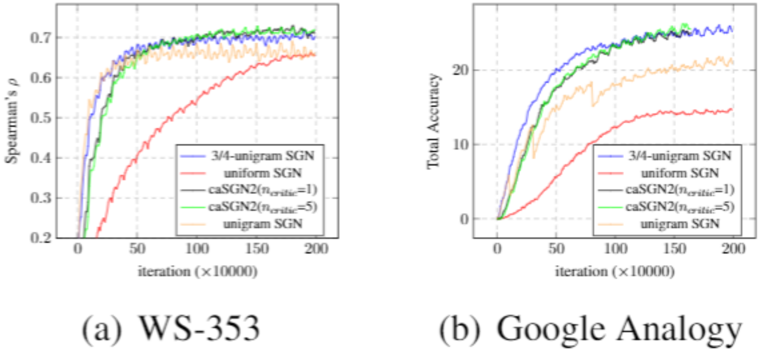


Figure 3: Curve of Spearman's ρ and the total accuracy. Notation n_{critic} is the number of iterations apply to the discriminator before per generator iteration