## 1 How to update the assignment of sense of each word:

Firstly, assign sense to each word in corpus randomly (assume each $w$ in corpus has n senses)

For each word $w$ in corpus:
$context(w) := u_1, u_2, ..., u_m$
each word $u_i (1 \leq i \leq m)$ of $context(w)$ already has sense $u_{i_s}$
(our goal is to select the best sense of $w$ from the information of $context(w)$)
$w_s$ = bestSense($w$)

bestSense($w$):
(assume $w$ has sense $w_1, w_2, ..., w_n$)
$score_j := p(w_j|u_{1_s}) * p(w_j|u_{2_s}) * ... * p(w_j|u_{m_s})$ $(1 \leq j \leq n)$
select biggest $score_j$ with $w_j$, and $w_s := w_j$

## 2 How to calculate $p(w_j|u_{i_s})$ $(1 \leq j \leq n, 1 \leq i \leq m)$:

Firstly, generate Negative-Sampling of word $w$: $NEG(w)$

$NEG(w) := z_1, z_2, ..., z_t$, each word $z_k (1 \leq k \leq t)$ of $NEG(w)$ is assigned sense $z_{k_s}$ randomly

$p(w_j|u_{i_s}) := \sigma(V(u_{i_s})^T \Theta(w_j)) * \prod_{1 \leq k \leq t} [1 - \sigma(V(u_{i_s})^T \Theta(z_{k_s}))]$

$\sigma(x) := \frac{1}{1+e^{-x}}$ , V := lookup-table for senses, $\Theta$ := predicting parameters for senses

## 3 How to update V and $\Theta$:

We need to update lookup-table of $w_s$ (the best sense of word w) and lookup-table of $z_{k_s}$ (random senses of $NEG(w)$)
We also need to update predicting parameters of $u_{i_s}$ (known senses of $context(w)$)

*loss function* for each pair of senses($w_j$ and $u_{i_s}$) := $-\log p(w_j|u_{i_s})$

And then we use $-\log p(w_j|u_{i_s})$ to calculate SGD (stochastic gradient descent), the detail about calculate gradient will be showed later in the code.
And then use this gradient to update V and $\Theta$.