3 Context Clustering based on Context Pair Classification Results

Given n mentions $\{C_i\}$ of a keyword, we use the following context clustering scheme. The discovered context clusters correspond to distinct word senses.

For any given context pair, the context similarity features defined in Section 2 are computed. With n mentions of the same keyword, $\frac{n(n-1)}{2}$ context similarities $CS_{i,j}$ $(i \in [1,n], j \in [1,i))$ are computed. Using the context pair classification model, each pair is associated with two scores $sc_{i,j}^0 = \log(\Pr(S_i = S_j | CS_{i,j}))$ and $sc_{i,j}^1 = \log(\Pr(S_i = S_j | CS_{i,j}))$ which correspond to the probabilities of two situations: the pair refers to the same or different word senses.

Now we introduce the symbol $\{K, M\}$ which refers to the final context cluster configuration, where K refers to the number of distinct sense, and M represents the many-to-one mapping (from contexts to a sense) such that $\mathrm{M}(\mathrm{i}) = \mathrm{j}, \mathrm{i} \in [1, \mathrm{n}], \mathrm{j} \in [1, \mathrm{K}]$. Based on the pairwise scores $\{sc_{i,j}^0\}$ and $\{sc_{i,j}^1\}$, WSD is formulated as searching for $\{K, M\}$ which maximizes the following global scores:

$$\mathrm{sc}(\{\mathrm{K},\mathrm{M}\}) = \sum_{\substack{\mathrm{i} \in [1,\mathrm{n}], \\ j \in [1,i)}} sc_{i,j}^{k(i,j)} \quad (2)$$

$$\text{where } k(i,j) = \begin{cases} 0, if\ M(i) = M(j) \\ 1, \quad otherwise \end{cases}$$

Similar clustering scheme has been used successfully for the task of co-reference in (Luo etc. 2004), (Zelenko, Aone and Tibbetts, 2004a) and (Zelenko, Aone and Tibbetts, 2004b).

In this paper, statistical annealing-based optimization (Neal 1993) is used to search for $\{K, M\}$ which maximizes Expression (2).

The optimization process consists of two steps. First, an intermediate solution $\{K, M\}_0$ is computed by a greedy algorithm. Then by setting $\{K, M\}_0$ as the initial state, statistical annealing is applied to search for the global optimal solution. The optimization algorithm is as follows.

1. Set the initial state $\{K, M\}$ as $K = n$, and $M(i) = i,\ \mathrm{i} \in [1, \mathrm{n}]$;
2. Select a cluster pair for merging that maximally increases $\mathrm{sc}(\{\mathrm{K},\mathrm{M}\}) = \sum_{\substack{\mathrm{i} \in [1,\mathrm{n}], \\ j \in [1,i)}} sc_{i,j}^{k(i,j)}$
3. If no cluster pair can be merged to increase $\mathrm{sc}(\{\mathrm{K},\mathrm{M}\}) = \sum_{\substack{\mathrm{i} \in [1,\mathrm{n}], \\ j \in [1,i)}} sc_{i,j}^{k(i,j)}$, output $\{K, M\}$ as the intermediate solution; otherwise, update $\{K, M\}$ by the merge and go to step 2.

Using the intermediate solution $\{K, M\}_0$ of the greedy algorithm as the initial state, the statistical annealing is implemented using the following pseudo-code:

Set $\{K, M\} = \{K, M\}_0$;
for($\beta = \beta_0; \beta < \beta_{\mathrm{final}}; \beta^* = 1.01$)
{
  iterate pre-defined number of times
  {
    set $\{K, M\}_1 = \{K, M\}$;
    update $\{K, M\}_1$ by randomly changing cluster number and cluster contents;
    set $x = \frac{\mathrm{sc}(\{\mathrm{K},\mathrm{M}\}_1)}{\mathrm{sc}(\{\mathrm{K},\mathrm{M}\})}$
    if(x>=1)
    {
      set $\{K, M\} = \{K, M\}_1$
    }
    else
    {
      set $\{K, M\} = \{K, M\}_1$ with probability x^{β}.
    }
    if $\mathrm{sc}(\{\mathrm{K},\mathrm{M}\}) > \mathrm{sc}(\{\mathrm{K},\mathrm{M}\}_0)$
    then set $\{K, M\}_0 = \{K, M\}$
  }
}
output $\{K, M\}_0$ as the optimal state.