Neural LMs
Feed forward

• Bengio et al. (2003, 2001) showed that NNLMs could be scaled reasonably well

• Feed forward based language model.

• \( p(w_t \mid w_{t-1} \ldots w_{t-n}) \). \( n \) was typically 4.

• Started with one-hot encodings of 4 words, \( w_{t-1} \ldots, w_{t-4} \)

• Projection operator \( C \) for lower dimension embedding of words
Procedure

\( w \in \mathbb{R}^{|V|}, C \in \mathbb{R}^d \times \mathbb{R}^{|V|} \quad C(w) = C.w \)

\( [w_{t-1}, w_{t-2} \ldots w_{t-n}] \rightarrow C.[w_{t-1}, w_{t-2} \ldots w_{t-n}] \)

\( f(x) = C.[w_{t-1}, w_{t-2} \ldots w_{t-n}] \in \mathbb{R}^{nd} \)

\( A \in \mathbb{R}^{nd \times h} \quad \sigma = \text{non-linearity} \quad B \in \mathbb{R}^{h \times |V|} \)

\( \hat{p}(y|x) \propto \exp (B^T.\sigma(A^T.f(x))) \)

error = Cross-entropy loss

\( CE = -\log \hat{p}(y^*|x) \)
RNNLM

- Motivation was to capture long distance dependencies via the hidden states at each time step.
- At each time step in the input, the prediction depends on the current word and the previous state. 
  \[ \hat{p}(y_t = w_{t+1} | w_t, s(t - 1)) = g(w_t, s(t - 1)) \]
Procedure

\[ w_t \in \mathbb{R}^d, \ s(t-1) \in \mathbb{R}^h, \ U \in \mathbb{R}^h \times \mathbb{R}^d, \ W \in \mathbb{R}^h \times \mathbb{R}^h \]

\[ s(t) = \sigma(U.w_t + W.s(t-1)) \]

\[ V \in \mathbb{R}^{|V|} \times \mathbb{R}^h \]

\[ \hat{p}(y_t = w_{t+1} | w_t, s(t-1)) \propto \exp(V.s(t)) \]
References
