Ứng dụng học sâu trong bài toán Xây dựng mô hình đối thoại

Nguyễn Minh Quân

Trường Đại học Bách khoa Hà Nội

nguyenminhquan20163374@gmail.com

Ngày 27 tháng 1 năm 2021

Nội dung

Hệ thống đối thoại người máy (Dialogue Systems)

- Trợ lý ảo tương tác hội thoại, thuật ngữ: **Chatbot**
- Tương tác với người dùng bằng ngôn ngữ tự nhiên.
- Ứng dụng trên nhiều các lĩnh vực khác nhau: Thương mại, y tế, tự động hoá,...
- Siri (2010), Google Now (2012), Alexa (2015), Google Assistant (2016),...

Mô hình truy xuất thông tin (Retrieval-based)

- Đưa ra những phản hồi được chuẩn bị trước, tuân theo những kịch bản nhất định
- Sử dụng các thuật toán heuristic, thuật toán học máy phân loại câu hỏi và đưa ra đáp án từ tập dữ liệu.

Mô hình sinh hội thoại (Generation-based)

- Không dựa trên tập dữ liệu được định nghĩa từ trước
- Học các hội thoại có sẵn bằng các thuật toán học máy
- Với câu hỏi đầu vào, mô hình sinh ra câu trả lời tương ứng.

Ưu điểm-Nhược điểm

Mô hình truy xuất thông tin

- Câu trả lời mạch lạc, đưa ra các thông tin hữu ích
- Vấn đề gán nhãn dữ liệu
- Trong nhiều lĩnh vực đặc thù, cần các chuyên gia.

Mô hình sinh hội thoại

- Không cần gán nhãn dữ liệu
- Câu trả lời không mang đầy đủ thông tin, có thể mắc lỗi ngữ pháp
- Yêu cầu số lượng hội thoại đủ lớn.

Hướng tiếp cận quá khứ và hiện tại: mô hình truy xuất thông tin.
Hướng tiếp cận tương lai: kết hợp hai mô hình trên [3].

Mạng neuron hồi tiếp (Recurrent Neural Network - RNN)

Mạng RNN nhận đầu vào là một dãy các vec-tơ $x_{1:n} = x_1, x_2, ..., x_n$, được định nghĩa đệ quy như sau:

$$\text{RNN}^*(x_{1:n}; s_0) = y_{1:n}, y_i = O(s_i), s_i = R(s_{i-1}, x_i) \quad (1)$$

trong đó

$$x_i \in \mathbb{R}^{d_{in}}, y_n \in \mathbb{R}^{d_{out}}, s_i \in \mathbb{R}^{f(d_{out})}.$$

Với các mô hình cụ thể, ta sẽ định nghĩa hàm R và hàm O.

Các biến thể của RNN

Mạng neuron hồi tiếp đơn giản (Simple Recurrent Network S-RNN)

$$s_i = R_{\text{SRNN}}(x_i, s_{i-1}) = g(s_{i-1}W^s + x_i W^s + b) \tag{2}$$

$$y_i = O_{\text{SRNN}}(s_i) = s_i$$

$$s_i, y_i \in \mathbb{R}^{d_s}, x_i \in \mathbb{R}^{d_x}, W^x \in \mathbb{R}^{d_x \times d_s}, W^s \in \mathbb{R}^{d_s \times d_s}, b \in \mathbb{R}^{d_s}.$$

trong đó hàm g là hàm kích hoạt phi tuyến, thường là hàm **ReLU** hoặc hàm **tanh**.

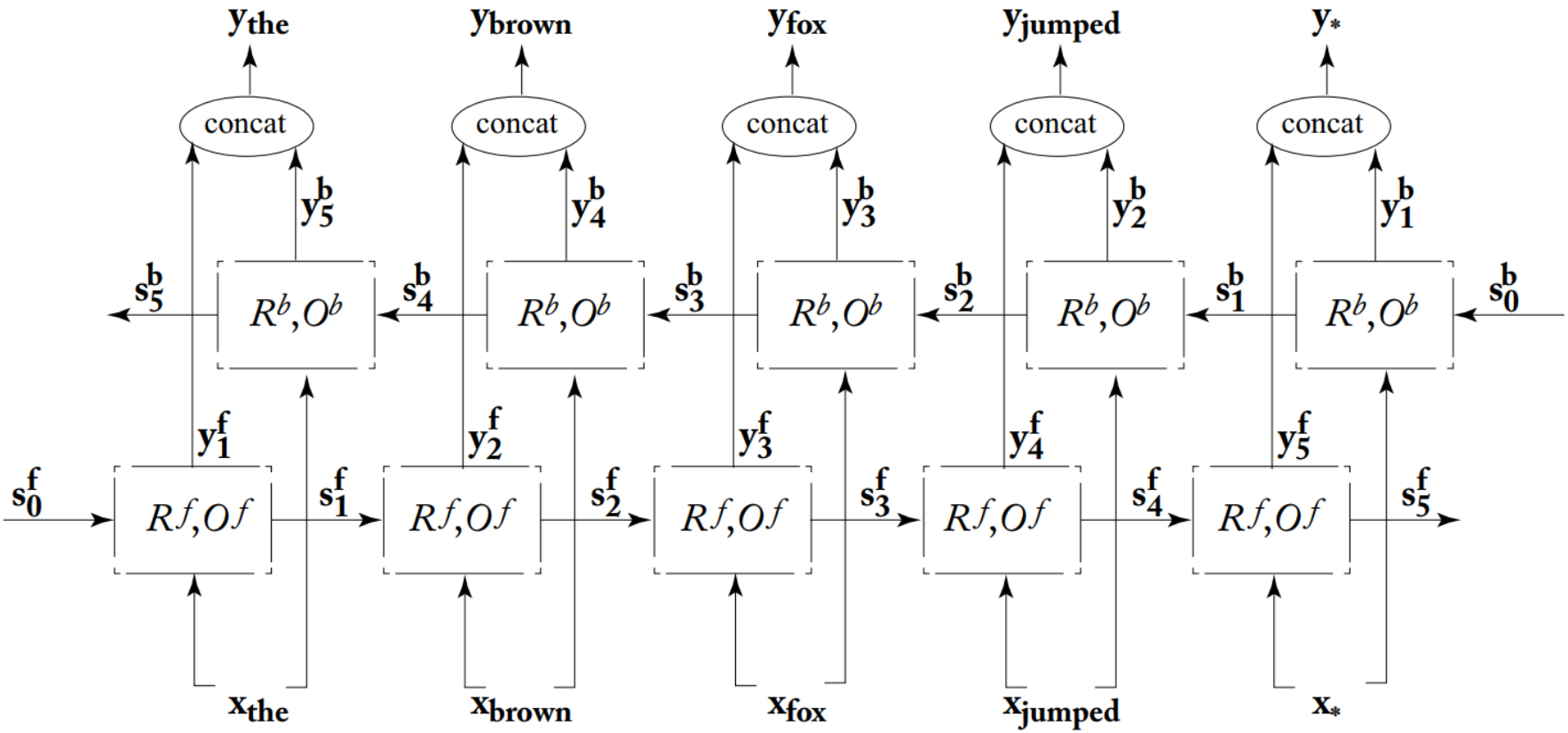
Các biến thể của RNN

Mạng neuron hồi tiếp hai chiều (Bidirectional RNN - biRNN)
Mạng biRNN bao gồm hai mạng RNN: mạng RNN thứ nhất (R^f, O^f) cung cấp cho chuỗi $x_{1:n}$, mạng RNN thứ hai (R^b, O^b) cung cấp cho chuỗi $x_{n:1}$. Vec-tơ đầu ra ở vị trí thứ i là sự kết hợp của hai vec-tơ đầu ra của hai mạng RNN tại vị trí thứ i:

$$\text{biRNN}(x_{1:n}, i) = y_i = [\text{RNN}^f(x_{1:i}), \text{RNN}^b(x_{n:i})] \tag{3}$$

$$y_i = [y_i^f; y_i^b] = [O^f(s_i^f); O^b(s_i^b)]$$

Mạng neuron hồi tiếp hai chiều (Bidirectional RNN)



Hình: Nguồn: [1]

Mạng Long Short-Term Memory (LSTM)

$$s_j = R_{\text{LSTM}}(s_{j-1}, x_j) = [c_j, h_j], y_j = O(s_j) = h_j. \tag{4}$$

trong đó c_j và h_j được tính toán dựa trên các phương trình sau:

$$\begin{aligned} c_j &= f * c_{j-1} + i * z \\ h_j &= o * \tanh(c_j) \\ i &= \sigma(x_j W^{xi} + h_{j-1} W^{hi}) \\ f &= \sigma(x_j W^{xf} + h_{j-1} W^{hf}) \\ o &= \sigma(x_j W^{xo} + h_{j-1} W^{ho}) \\ z &= \tanh(x_j W^{xz} + h_{j-1} W^{hz}) \end{aligned}$$

$$s_j \in \mathbb{R}^{2.d_h}, x_i \in \mathbb{R}^{d_x}, c_j, h_j, i, f, o, z \in \mathbb{R}^{d_h}, W^{xo} \in \mathbb{R}^{d_x \times d_h}, W^{ho} \in \mathbb{R}^{d_h \times d_h}.$$

Mô hình Sequence-to-Sequence (seq2seq)

- Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 14 Dec 2014. "Sequence to Sequence Learning with Neural Networks"
- Phát triển với bài toán dịch máy.
- Bao gồm hai mạng neuron: thành phần mã hoá (Encode) và thành phần giải mã (Decode).

Thành phần mã hoá (Encode)

Thành phần Encode nhận đầu vào là một dãy các vec-tơ $\mathbf{x} = (x_1, ..., x_{T_x})$ thành một vec-tơ c.

$$h_t = f(x_t, h_{t-1}) \tag{5}$$

và

$$c = q(\{h_1, ..., h_{T_x}\}),$$

trong đó $h_t \in \mathbb{R}^n$ là trạng thái ẩn thứ t và c là vec-tơ được tạo ra từ dãy các trạng thái ẩn. f và q là các hàm phi tuyến.

Thành phần giải mã (Decode)

Thành phần Decode được huấn luyện để dự đoán từ tiếp theo $y_{t'}$ từ vec-tơ ngữ cảnh c và tất cả các từ trước đó $\{y_1, ..., y_{t'-1}\}$.

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t | \{y_1, ..., y_{t-1}\}, c) \tag{6}$$

với $\mathbf{y} = (y_1, ..., y_{T_y})$. Với RNN, mỗi xác suất có điều kiện được mô hình hoá như sau

$$p(y_t | \{y_1, ..., y_{t-1}\}, c) = g(y_{t-1}, s_t, c), \tag{7}$$

trong đó g là hàm phi tuyến, hàm đưa ra xác suất của y_t và s_t là trạng thái ẩn của RNN.

Cơ chế Attention

- Bài báo đầu tiên: Dzmitry Bahdanau et al., 1 Sep 2014. "Neural Machine Translation by Jointly Learning to Align and Translate".
- Xác suất có điều kiện

$$p(y_i|y_1, ..., y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i), \tag{8}$$

trong đó s_i là trạng thái ẩn thứ i, được tính:

$$s_i = f(s_{i-1}, y_{i-1}, c_i).$$

Cơ chế Attention (tt)

Vec-tơ ngữ cảnh c_i phụ thuộc vào mỗi chuỗi các $(h_1, ..., h_{T_x})$ trong bộ mã hoá từ đầu vào. Vec-tơ c_i là tổng trọng số của các vec-tơ h_i:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j. \tag{9}$$

Các trọng số α_{ij} của mỗi h_j được tính toán:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}, \tag{10}$$

trong đó

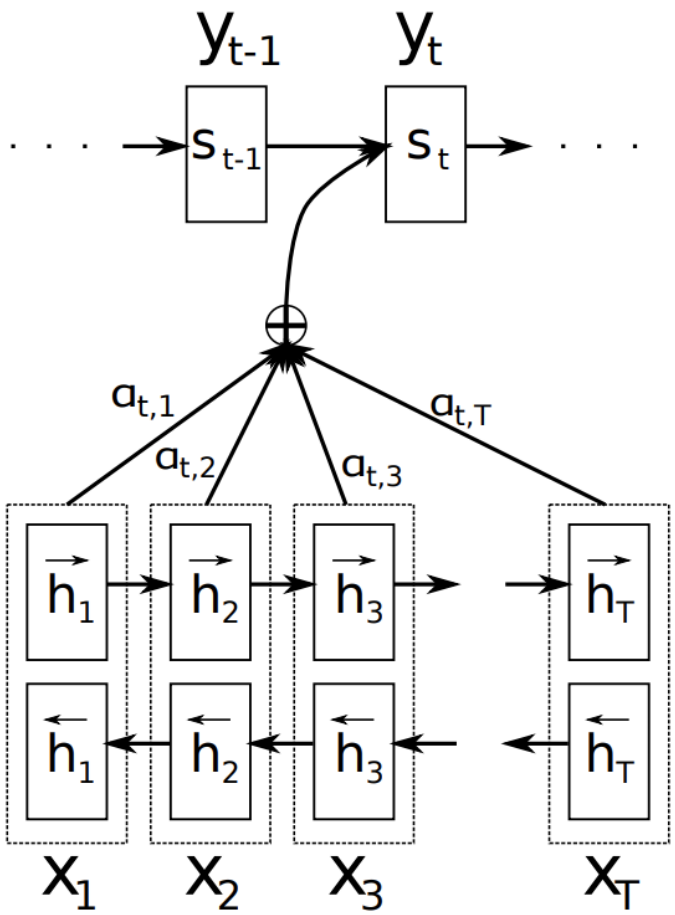
$$e_{ij} = a(s_{i-1}, h_j)$$

Trong bài báo Bahdanau et al. các tác giả đề xuất **alignment model** như sau:

$$\text{score}(s_{i-1}, h_j) = a(s_{i-1}, h_j) = v_a^T \tanh(W_a s_{i-1} + U_a h_j),$$

là một mạng neuron với $W_a \in \mathbb{R}^{n \times n}$, $U_a \in \mathbb{R}^n$ và $v_a \in \mathbb{R}^n$ là các ma trận trọng số.

Cơ chế Attention (tt)



Hình: Nguồn: [4]

Cài đặt mô hình

Mô hình sử dụng	Mô hình Sequence to Sequence
Pha mã hoá	biLSTM
Pha giải mã	LSTM
Cơ chế Attetion	Soft Attention
Thuật toán tối ưu	Thuật toán Adam
Hàm mất mát	Hàm Cross Entropy
Core	TensorFlow version 1.4.0
Xây dựng ứng dụng web	HTLM, CSS, Javascript, Python Flask

Bảng: Kiến trúc của ứng dụng Chatbot.

Thực hiện bài toán

Quy trình thực hiện bài toán:

- Hiển thị dữ liệu, phân tích dữ liệu và tiền xử lý dữ liệu.
- Cài đặt mô hình.
- Lựa chọn các tham số cho mô hình.
- Huấn luyện mô hình.
- Hiệu chỉnh các tham số.
- Cài đặt chương trình, cài đặt giao diện.

Các bộ tham số	Thứ nhất	Thứ hai	Thứ ba
batch size	128	512	128
embedding size	128	512	128
rnn size	128	512	128
learning rate	0.001	0.001	0.001
epochs	500	100	200
keep_probability	0.75	0.75	0.75
max conversation length	5	5	6
min conversation length	2	2	2

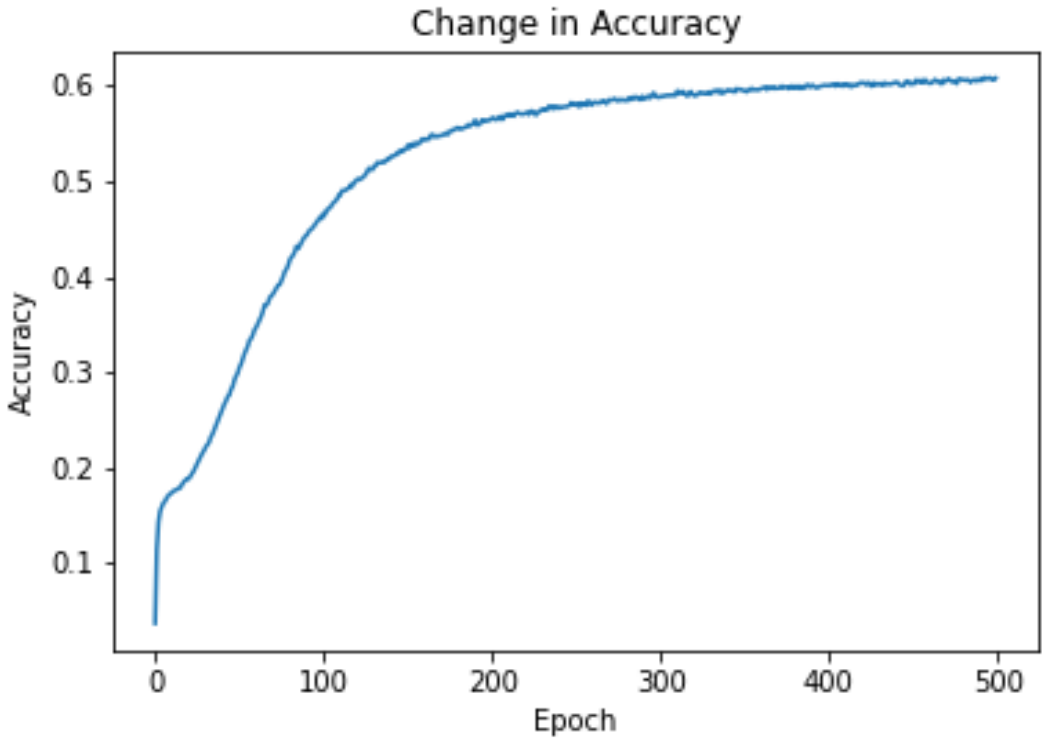
Bảng: Các tham số đầu vào của mô hình.

Kết quả

Các bộ tham số	Sai số	Hàm mất mát	Thời gian
Bộ thứ nhất	0.635	0.170	11.5
Bộ thứ hai	0.611	0.165	12.5
Bộ thứ ba	0.478	0.250	10

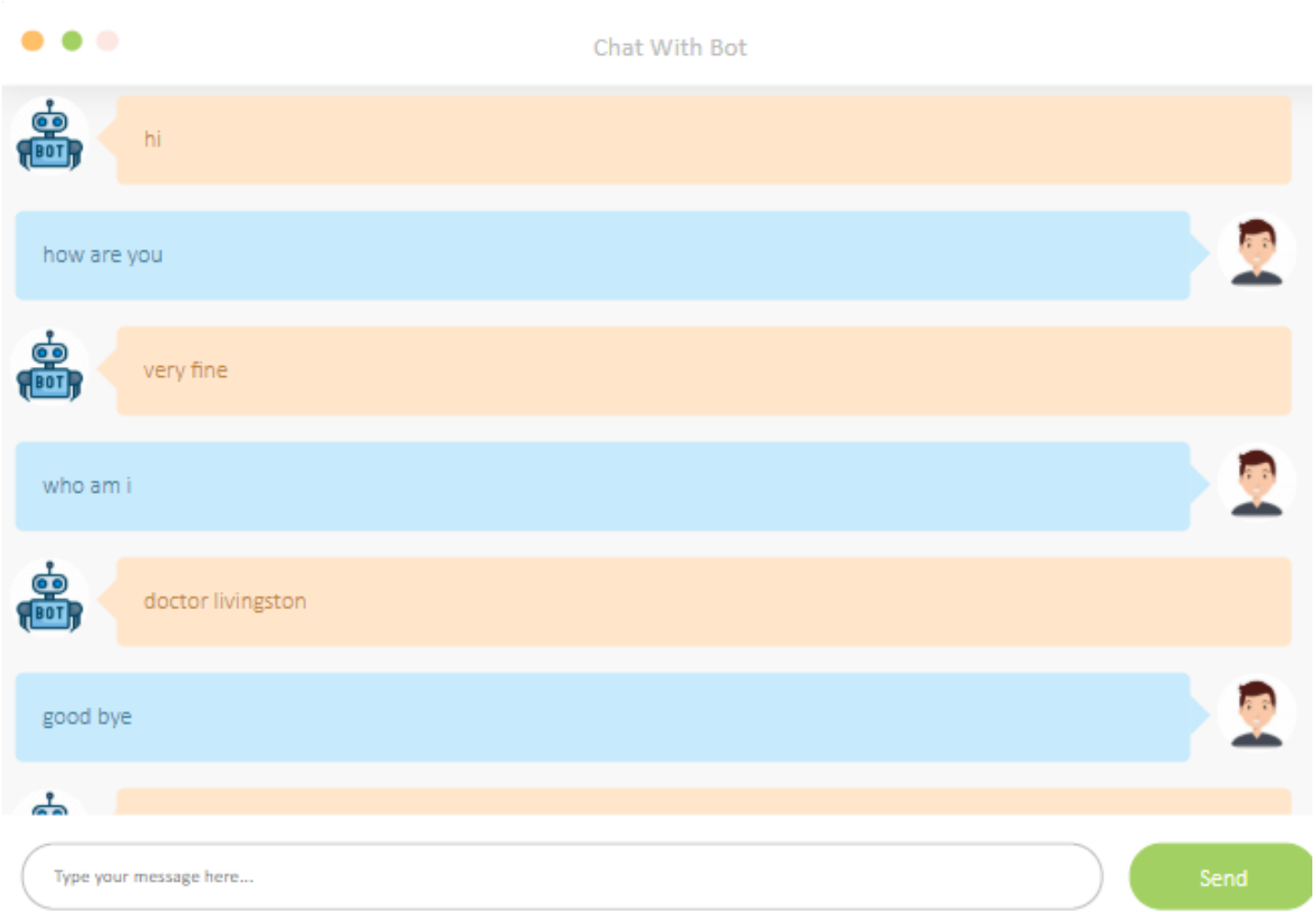
Bảng: Kết quả thực nghiệm với các bộ tham số.

Kết quả (tt)



Hình: Độ chính xác của mô hình.

Chatbot



Hình: Giao diện web người dùng với Chatbot.

Tài liệu tham khảo

- Yoav Goldberg, *Neural Network Methods for Natural Language Processing*, Morgan & Claypool Publishers, 2017.
- Ilya Sutskever, Oriol Vinyals, Quoc V. Le, 14 Dec 2014. "Sequence to Sequence Learning with Neural Networks".
- Liu Yang et al., 25 Aug 2019. "A Hybrid Retrieval-Generation Neural Conversation Model".
- Dzmitry Bahdanau et al., 1 Sep 2014. "Neural Machine Translation by Jointly Learning to Align and Translate".
- T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. "Recurrent neural network based language model". In INTERSPEECH, pages 1045–1048, 2010.

The End