End-to-End quantum language model with Application to Question Answering

Peng Zhang[1], JiaBin Niu[1], Zhan Su[1], Bengyou Wang[2], Liqun Ma[1], Dawei Song[1]
Tianjin University[1]
Tencent[2]
Contents

- QA System
- Statistical Language Model
- Quantum Language Model
- NN-based Quantum Language Model
- Quantum AI
QA system in Tencent

➢ Community QA
  ● FAQs

➢ KBQA
  ● Knowledge Base

➢ Passage QA
  ● Only unstructured documents
Two-step Architecture in Community QA
Textual Matching

- **Unsupervised Models**
  - TFIDF/BM25
  - language model

- **Neural Network Models**
  - DSSM
  - CNN/RNN variants
Contents

- QA System
- **Statistical Language Model**
- Quantum Language Model
- NN-based Quantum Language Model
- Quantum AI
• For a sequence of terms in the document $d=w_1w_2...w_n$, SLM calculates the probability $P(w_1w_2...w_n)$. Based on Bayes’ rule, we have:

$$p(w_1w_2\cdots w_n) = p(w_1)p(w_2\cdots w_n | w_1)$$

$$= p(w_1) \prod_{i=2}^{n} p(w_i | w_{i-1}\cdots w_1)$$
SLM-based IR model (SLMIR)

- **Query likelihood model**: define the relevance as the generative probability of the current query w.r.t. each document.

- **Translation model**: define the relevance as the probability that the query would have been generated as a translation of the document, and factor in the user’s general preferences in the form of a prior distribution over documents.

- **KL-divergence model**: query and document are correspond to two different languages. And the relevance is defined as the KL-divergence between the two language models.

- We focus on KL-divergence model in this talk.
Contents

- QA System
- Statistical Language Model
- **Quantum Language Model**
- NN-based Quantum Language Model
- Quantum AI
Quantum Concept

- Simple Example:

A unit vector $\hat{u} \in \mathbb{R}^n$, $\|\hat{u}\|_2 = 1$ is written as $|\hat{u}\rangle$ (ket)

The transpose $\hat{u}^\top$ is written as $\langle \hat{u} |$ (bra)

The projector onto the direction $\hat{u}$ writes as $|\hat{u}\rangle\langle \hat{u} |$ (dyad), corresponding to the pure state

The inner product between two vectors writes as $\langle \hat{u} | \hat{v} \rangle$

The elements of the standard basis in $\mathbb{R}^n$ are denoted as $|e_i\rangle = \delta_{1i}, \ldots, \delta_{ni}$, where $\delta_{ij} = 1$, iff $i = j$

Generally, any ket $|\psi\rangle = \sum_i v_i |u_i\rangle$ is called a superposition of the $|u_i\rangle$, where $\{|u_1\rangle, \ldots, |u_n\rangle\}$ form an orthonormal basis

- $|e_1\rangle = (1,0)^\top, |e_2\rangle = (0,1)^\top$

  $s_1 = |e_1\rangle\langle e_1 | = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$

  $s_2 = |e_2\rangle\langle e_2 | = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$

  $s_3 = \frac{1}{\sqrt{2}} (s_1 + s_2)$

  $\hat{v} = \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right)^\top$

  Projection_1 $= s_1 \cdot \hat{v} = \left( \frac{1}{\sqrt{2}}, 0 \right)^\top$

  Projection_2 $= s_2 \cdot \hat{v} = \left( 0, \frac{1}{\sqrt{2}} \right)^\top$

  Projection_3 $= s_3 \cdot \hat{v} = \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right)^\top$
Density Matrix

A density matrix corresponds to the discrete probability distribution in classical probability theory. It assigns a quantum probability to each one of the infinite dyads (an elementary event in quantum probability). For a density matrix $\rho$.

$$\rho = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$$

$$\mu_\rho(|e\rangle\langle e|) = tr(\rho |e\rangle\langle e|) = 0.5, \mu_\rho(|f\rangle\langle f|) = tr(\rho |f\rangle\langle f|) = 1$$

where:

$$|e\rangle = (1,0)^T$$

$$|e\rangle\langle e| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$|f\rangle = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)^T$$

$$|f\rangle\langle f| = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$$

Quantum Language Models (QLM)

- For example:

\( V = \{ \text{computer, architecture, system} \}, \ W_d = \{ \text{computer, architecture} \} \)

\( P_d = \{ E_{\text{computer}}, E_{\text{architecture}} \} \)

\( E_{\text{computer}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \), \( E_{\text{computer}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \)

- If we observe the dependency of “computer” and “architecture”

\( k_{ca} = \sigma_c |e_c\rangle + \sigma_a |e_a\rangle, \) Set \( \sigma_c = \sqrt{2}/3, \sigma_a = \sqrt{1}/3 \)

\( K_{ca} = \begin{bmatrix} 2/3 & \sqrt{2}/3 & 0 \\ \sqrt{2}/3 & 2/3 & 0 \\ 0 & 0 & 0 \end{bmatrix} \)
N-gram in extended Vector Space

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>10</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Frequency</td>
<td>0.33</td>
<td>0.2</td>
<td>0.166</td>
<td>0.133</td>
<td>0.1</td>
<td>0.066</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ P(W | \theta_d ) = [0.33, 0.2, 0.166, 0.133, 0.1, 0.066, 0] \]

\[ |V| = C_n^1 + C_n^2 + \cdots + C_n^n = \sum_{i=0}^{n} C_n^i \]

The dimension of parameter in extended Vector Space: \( o(n!) \)
Term Dependency (N-gram) in QLM

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Projection</td>
<td>$e_c=[1,0,0]$</td>
<td>$e_a=[0,1,0]$</td>
<td>$e_s=[0,0,1]$</td>
<td>$k_{ac}=[\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2},0]$</td>
<td>$k_{cs}=[\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}]$</td>
<td>$k_{as}=[0, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]$</td>
<td>$k_{cas}=[\frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}, \frac{\sqrt{3}}{3}]$</td>
</tr>
<tr>
<td>Frequency</td>
<td>$\text{Tr}(\rho</td>
<td>e_c\rangle\langle e_c</td>
<td>)$</td>
<td>$\text{Tr}(\rho</td>
<td>e_a\rangle\langle e_a</td>
<td>)$</td>
<td>$\text{Tr}(\rho</td>
</tr>
</tbody>
</table>

\[
\rho = \begin{bmatrix}
\frac{2}{3} & \frac{\sqrt{2}}{3} & 0 \\
\frac{\sqrt{2}}{3} & \frac{2}{3} & 0 \\
\frac{3}{3} & \frac{3}{3} & 0
\end{bmatrix}
\]

\[
P(W|\theta_d) = 
\begin{bmatrix}
\text{Tr}(\rho|e_c\rangle\langle e_c|), \text{Tr}(\rho|e_a\rangle\langle e_a|), \text{Tr}(\rho|e_s\rangle\langle e_s|), \text{Tr}(\rho|k_{ac}\rangle\langle k_{ac}|), \text{Tr}(\rho|k_{cs}\rangle\langle k_{cs}|), \text{Tr}(\rho|k_{as}\rangle\langle k_{as}|), \text{Tr}(\rho|k_{cas}\rangle\langle k_{cas}|)
\end{bmatrix}
\]

The dimension of parameter in QLM: $o(n^2)$
Definition 1. (QE): Let $A$ be an $n$-qubit system in a state $|\phi_A\rangle$ and $\{A_1, A_2\}$ be a partition of $A$, where two disjoint parts $A_1$ and $A_2$ have $0 < k < n$ qubits and $n - k$ qubits, respectively. $A$ is entangled iff. there does NOT exist any tensor product decomposition of $|\phi_A\rangle$ such that $|\phi_A\rangle = |\phi_{A_1}\rangle \otimes |\phi_{A_2}\rangle$, where $|\phi_{A_1}\rangle$ and $|\phi_{A_2}\rangle$ are the states of $A_1$ and $A_2$, respectively.

Definition 2. (UPD): A pattern $A = \{W_1, W_2, \cdots, W_n\}$ forms the UPD pattern iff. the joint probability distribution over $A$ cannot be unconditionally factorized, i.e., there does NOT exist any $m$-partition $\{A_1, A_2, \cdots, A_m; m > 1\}$ of $A$, so that $p(a) = p(a_1) \cdot p(a_2) \cdots p(a_m)$, where $p(a_i)$, $i = 1, 2, \cdots, m$, is the joint distribution over $A_i$. 

- Modeling quantum entanglements in quantum language models. IJCAI 2015
• LM: a document $d$ is represented by a sequence of terms
• QLM: $d$ is represented by a sequence of quantum events (with dyads for a term or a dependency)

[Sordoni, Nie, Bengio, 2013]
Computing probabilities

\[
p(s) = \sum p(\varphi_i) |\langle s|\varphi_i\rangle|^2
\]

\[
= \sum p(\varphi_i) \langle \varphi_i | \hat{M}_s^\dagger \hat{M}_s | \varphi_i \rangle
\]

\[
= \sum p(\varphi_i) \text{tr}(\hat{M}_s | \varphi_i \rangle \langle \varphi_i |)
\]

\[
= \text{tr} \left( \hat{M}_s p(\varphi_i) \sum_{\rho} | \varphi_i \rangle \langle \varphi_i | \right)
\]

\[
= \text{tr}(\hat{M}_s \rho)
\]

\[
= \text{tr}(\rho \hat{M}_s)
\]

Where \( \rho \) is a density matrix
• A density matrix $\rho$ to represent sentence

• Given the observed projectors $P_d = \{\Pi_1, \ldots, \Pi_M\}$ for sentence $S$, the quantum language model $\rho$ is estimated through Maximum Likelihood Estimation, and the likelihood is represented as:

$$L_{P_d}(\rho) = \prod_{i=1}^{M} tr(\rho \Pi_i)$$
Maximum likelihood estimation for QLM

• Likelihood:

\[ \mathcal{L}_{\mathcal{P}_d}(\rho) = \prod_{i=1}^{M} \text{tr}(\rho \Pi_i). \]

• Estimation/Training of Density Matrix:

\[
\begin{align*}
\text{maximize} \quad & \log \mathcal{L}_{\mathcal{P}_d}(\rho) \\
\text{subject to} \quad & \rho \in \mathcal{S}^n.
\end{align*}
\]

• Matching:

\[
\begin{align*}
-\Delta_{VN}(\rho_q \| \rho_d) & \overset{\text{rank}}{=} - \text{tr}(\rho_q (\log \rho_q - \log \rho_d)) - \text{tr}(\rho_q \log \rho_d),
\end{align*}
\]
Limitation in QLM

- If the two documents do not share any words, especially in short text
  - Use embedding as a basic vector

- It is independent with the label.
  - Training in a end-2-end network

Neural Network based Quantum Language Model
Contents

- QA System
- Statistical Language Model
- Quantum Language Model
- NN-based Quantum Language Model
- Quantum AI
Simple version: NNQLM1

- Density matrix representation for sentences (q or a)

\[ \rho = \sum_i p_i S_i = \sum_i p_i |S_i\rangle \langle S_i| \]

\[ |S_i\rangle = \frac{S_i}{|S_i|} \]

\[ \rho = \Sigma p_i S_i = \Sigma p_i |S_i\rangle \langle S_i| \]

\[ |S_i\rangle = \frac{|S_i|}{|S_i|} \]
Using the product of the density matrixes as their joint representation, the combined representations show the similarity of their density matrices.
Inter-sentence Similarities

- Since the density matrix is semi-positive, it

\[ \rho_q = \sum_i \lambda_i |r_i\rangle \langle r_i| \]

\[ \rho_a = \sum_j \lambda_j |r_j\rangle \langle r_j| \]

\[ \rho_q \rho_a = \sum_{i,j} \lambda_i \lambda_j |r_i\rangle \langle r_i| r_j\rangle \langle r_j| \]

\[ = \sum_{i,j} \lambda_i \lambda_j \langle r_i| r_j\rangle \langle r_i| r_j\rangle \]

\[ \text{tr}(\rho_q \rho_a) = \sum_{i,j} \lambda_i \lambda_j \langle r_i| r_j\rangle^2 \]

the similarity between \( \rho_q \) and \( \rho_a \)
Future work in Quantum-inspired NN

- **Complex embedding**
  
  Richer input, higher performance

- **Interference in NN**, 
  
  Cross-modal fusion

- **Entanglement in NN**
  
  Connection and memory in NN

More works try to bridge the gap between Quantum Concept and Deep learning\[1\], It may open a new door to reveal the black-box inner mechanism of Neural Network

---

\[1\] Deep Learning and Quantum Entanglement: Fundamental Connections with Implications to Network Design. ICLR 2018
Contents

➢ QA System
➢ Statistical Language Model
➢ Quantum Language Model
➢ NN-based Quantum Language Model
➢ Quantum AI
Exploration in **Quantum AI**

- Machine learning algorithm in Quantum computer

- **✓** Quantum-inspired models and ideas, but not depends on Quantum Computer
Quantum on general AI

- Solving the quantum many-body problem with artificial neural networks[J]. *Science*, 2017
- Deep Learning and Quantum Entanglement: Fundamental Connections with Implications to Network Design. *ICLR 2018*
- Deep complex Network. *ICLR 2018*
- SchNet: A continuous-filter convolutional neural network for modeling quantum interactions, *NIPS 2017*
Quantum AI on Language

- Modeling multi-query retrieval tasks using density matrix transformation. *SIGIR 2015*
- Modeling quantum entanglements in quantum language models. *IJCAI 2015*
- Learning Concept Embeddings for Query Expansion by Quantum Entropy Minimization. *AAAI 2014*
- Modeling latent topic interactions using quantum interference for information retrieval. *CIKM 2013*
- Modeling term dependencies with quantum language models for IR. *SIGIR 2013*
- Pure high-order word dependence mining via information geometry, *ICTIR 2011 best paper.*