End-to-End Quantum-like Language Models with Application to Question Answering

Peng Zhang[1,*], Jiabin Niu[1], Zhan Su[1], Benyou Wang[2], Liqun Ma[3], Dawei Song[1,4,*]
1. School of Computer Science and Technology, Tianjin University, Tianjin, China
2. Department of Social Network Operation, Social Network Group, Tencent, Shenzhen, China
3. School of Electrical and Information Engineering, Tianjin University, Tianjin, China
4. Computing and Communications Department, The Open University,United Kingdom
* Corrrespondence: {pzhang, dwsong}@tju.edu.cn

Abstract

Language Modeling (LM) is a fundamental research topic in a range of areas. Recently, inspired by quantum theory, a novel Quantum Language Model (QLM) has been proposed for Information Retrieval (IR). In this paper, we aim to broaden the theoretical and practical basis of QLM. We develop a Neural Network based Quantum-like Language Model (NNQLM) and apply it to Question Answering. Specifically, based on word embeddings, we design a new density matrix, which represents a sentence (e.g., a question or an answer) and encodes a mixture of semantic subspaces. Such a density matrix, together with a joint representation of the question and the answer, can be integrated into neural network architectures (e.g., 2-dimensional convolutional neural networks). Experiments on the TREC-QA and WIKIQA datasets have verified the effectiveness of our proposed models.

Introduction

Language Models (LM) play a fundamental role in Artificial Intelligence (AI) related areas, e.g., natural language processing, information retrieval, machine translation, speech recognition and other applications. The commonly used language models include statistical language models and neural language models. Generally speaking, statistical language models compute a joint probability distribution over a sequence of words (Manning, Raghavan, and Schütze 2008; Zhai 2008), while neural language models can obtain a distributed representation for each word (Bengio et al. 2003; Mikolov et al. 2013).

Recently, by using the mathematical formulations of quantum theory, a Quantum Language Model (QLM), has been proposed in Information Retrieval (IR). QLM encodes the probability uncertainties of both single and compound terms in a density matrix, without resorting to extend the vocabulary artificially (Sordoni, Nie, and Bengio 2013). The ranking of documents against a query is based on the von-Neumann divergence between the density matrices of the query and each document. QLM shows an effective performance on the ad-hoc retrieval task.

QLM is theoretically significant, as for the first time it generalizes LM via the formulations of Quantum theory. However, it has the following limitations. First, in QLM, the representation for each term is a one-hot vector, which only encodes the local occurrence, yet without taking into account the global semantic information. Second, QLM represents a sequence of terms (e.g., a query or a document) by a density matrix, which is estimated via an iterative process, rather than an analytical procedure. Thus it is difficult to integrate such a matrix in an end-to-end design. Third, QLM deals with the representation, estimation and ranking processes sequentially and separately. As a consequence, these three steps cannot be jointly optimized, thus limiting QLM's applicability and impact in the related research areas.

In this paper, we aim to broaden the theoretical and practical basis of QLM, by addressing the above problems. Specifically, we adopt the word embeddings to represent each word since such a distributed representation can encode more semantic information than one-hot vectors. By treating each embedding vector as an observed state for each word, a sentence (e.g., a question or an answer) can correspond to a mixed state represented by a density matrix. Then, we can derive such a density matrix without an iterative estimation step. This makes the density matrix representation feasible to be integrated into a neural network architecture and automatically updated by a back propagation algorithm. After getting the word level and sentence level representations, we can have a joint representation for two sentences (e.g., question and answer sentences in the answer selection task).

Based on the above ideas, we propose an end-to-end model, namely Neural Network based Quantum-like Language Model (NNQLM). Two different architectures are designed. The first is just adding a single softmax layer to the diagonal values and its trace value of the joint representation. The second is built upon a Convolutional Neural Network (CNN), which can automatically extract more useful patterns from the joint representation of density matrices. We clarify that our motivation of using quantum theory is to inspire new perspectives and formulations for the natural language applications, instead of developing quantum computation algorithms. Indeed, one can build the analogy between quantum theory (e.g., quantum probability) and some macro-world problems (Bruza, Wang, and Busemeyer 2015; **?**). For sake of applicability, our model does not fully comply with the theory of quantum probability, so that we will use "quantum-like" instead of "quantum" when referring to the language model we propose in this paper. To the best of our knowledge, this is the first attempt to integrate the quantum-like probability theory with the neural network architecture in Natural Language Processing (NLP) tasks.

We apply the proposed end-to-end quantum-like language models to a typical QA task, namely Answer Selection, which aims to find accurate answers from pre-selected set of candidates (Yang, Yih, and Meek 2015). For each single sentence (question or answer), the density matrix will represent the *mixture of semantic subspaces* spanned by embedding vectors. For each question-answer pair, the joint density matrix representation encodes the *inter-sentence similarity* information between the question and the answer. The neural network architecture (e.g. 2-dimensional CNN) is adopted to learn *useful similarity pattens* for matching and ranking the answers against the given question. A series of systematic experiments on TREC_QA and WikiQA have shown that the proposed NNQLM significantly improves the performance over QLM on both datasets, and also outperforms a state of the art end-to-end answer selection approach on TREC_QA.

Quantum Preliminaries

The mathematical formalism of quantum theory is based on linear algebra. Now, we briefly introduce some basic concepts and the original quantum language model.

Basic Concepts

In quantum probability (Von Neumann 1955), the probabilistic space is naturally represented in a Hilbert space, denoted as $\mathbb{H}^n$. For practical reasons, the previous quantum-inspired models limited the problem in the real space, denoted as $\mathbb{R}^n$ (Sordoni, Nie, and Bengio 2013). The Dirac's notation is often used, which denotes a unit vector $\vec{u} \in \mathbb{R}^n$ as a *ket* $|u\rangle$ and its transpose $\vec{u}^T$ as a *bra* $\langle u|$. The *inner product* between two *state vectors* is denoted as $\langle u|v\rangle$. The projector onto the direction $|u\rangle$ is $|u\rangle\langle u|$, which is an *outer product* (also called *dyad*) of $|u\rangle$ itself. Each rank-one projector $|u\rangle\langle u|$ can represent a *quantum elementary event*.

Density matrices are a generalization of the conventional finite probability distributions. A density matrix ρ can be defined as a mixture of dyads $|\psi_i\rangle\langle\psi_i|$:

$$\boldsymbol{\rho} = \sum_i p_i \, |\psi_i\rangle\langle\psi_i| \tag{1}$$

where $|\psi_i\rangle$ is a pure state vector with probability p_i. ρ is symmetric, positive semidefinite, and of trace 1 ($\mathrm{tr}(\boldsymbol{\rho}) = 1$). According to the Gleason's Theorem (Gleason 1957), there is a bijective correspondence between the quantum probability measure μ and the density matrix $\boldsymbol{\rho}$ (i.e., $\mu_{\boldsymbol{\rho}}(|u\rangle\langle u|) = \mathrm{tr}(\boldsymbol{\rho}\,|u\rangle\langle u|)$).

Quantum Language Model

A quantum language model represents a word or a compound dependency between words by a quantum elementary event. For each single word w_i, the corresponding projector $\mathbf{\Pi}_i = |e_i\rangle\langle e_i|$, where $|e_i\rangle$, the standard basis vector associated to a word, is an one-hot vector. Sordoni, Nie, and Bengio (2013) utilized the Maximum Likelihood Estimation (MLE) to estimate the density matrices $\boldsymbol{\rho}_q$ and $\boldsymbol{\rho}_d$, which represent the query q and document d, respectively. The original MLE estimation uses a so-called $R\rho R$ algorithm, which is an Expectation-Maximization (EM) iterative algorithm and does not have an analytical solution. After the estimation of density matrices, the ranking is based on the negative Von-Neumann (VN) Divergence between $\boldsymbol{\rho}_q$ and $\boldsymbol{\rho}_d$ (i.e., $-\triangle_{VN}(\boldsymbol{\rho}_q||\boldsymbol{\rho}_d) = \mathrm{tr}(\boldsymbol{\rho}_q \log \boldsymbol{\rho}_d)$). It turns out that the original QLM cannot be directly integrated into an end-to-end mechanism. This motivates us to consider a Neural Network architecture.

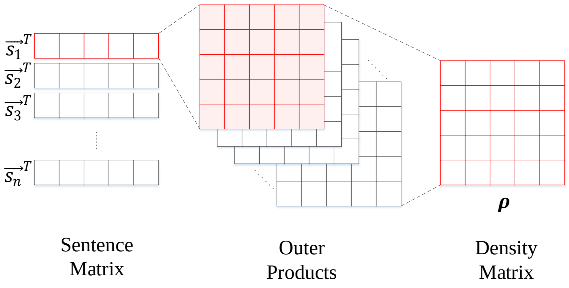

Figure 1: Single Sentence Representation

Neural Network based Quantum-like Language Model

In this section, we will describe our model in the context of a typical Question Answering task, namely answer selection, which aims to find accurate answers from a set of pre-selected candidate answers based on a question-answer similarity matching process. Note that the proposed model is general and can also be applied to other NLP and IR tasks that involve similarity matching and ranking.

Specifically, we introduce our model in three steps. Firstly, we design an embedding based density matrix representation for single sentences to model the intra-sentence semantic information carried by a question/answer. Then, we introduce a joint representation to model the inter-sentence similarities between a question and an answer. Finally, the question and answer are matched according to similarity features/patterns obtained from the joint representation. All these parts are integrated into a neural network structure.

Embedding based Density Matrix for Single Sentence Representation

Formally, word embeddings are encoded in an embedding matrix $\boldsymbol{E} \in \mathbb{R}^{|V| \times d}$, where $|V|$ is the length of the vocabulary and d is the dimension of the word embeddings. Different from the one-hot representation, word embeddings are obtained from the whole corpus or certain external large corpora, and thus contain global semantic information. As shown in Figure 1, the i^{th} word in a sentence is represented by a vector $\vec{s_i} \in \boldsymbol{E}$. Such a distributed representation for each word can naturally serve as an observed state for a sentence. To obtain a unit state vector, we normalize each word embedding vector ($\overrightarrow{s_i} \in \boldsymbol{E}$) as follows:

$$|s_i\rangle = \frac{\overrightarrow{s_i}}{||\overrightarrow{s_i}||_2} \tag{2}$$

Then, a sentence (e.g., question or answer) can correspond to a mixed state represented by a density matrix. According to the definition of density matrix, we can derive it as

$$\boldsymbol{\rho} = \sum_i p_i \mathbf{S}_i = \sum_i p_i \, |s_i\rangle\langle s_i| \tag{3}$$

where $\sum_i p_i = 1$, and $\boldsymbol{\rho}$ is symmetric, positive semidefinite and of trace 1. $\mathbf{S}_i$ is a semantic subspace spanned by the embedding-based state vector $|s_i\rangle$. Each outer product $|s_i\rangle\langle s_i|$ can be regarded as a partial Positive Operator-Valued Measurement (partial POVM) (Blacoe 2014), which is more general than the Projector-based Measurement in the original QLM. In addition, compared with the projectors $\boldsymbol{\Pi}_i$ in QLM, $\mathbf{S}_i$ carries more semantic information due to the embedding based state vectors $|s_i\rangle$ as against the one-hot vectors $|e_i\rangle$ that forms $\boldsymbol{\Pi}_i$.

In Eq. 3, p_i ($\sum_i p_i = 1$) is the corresponding probability of the state $|s_i\rangle$ with respect to the i^{th} word s_i in a given sentence. In practice, the values of p_i reflect the weights of the words in different positions of the sentence, and can be considered as a parameter automatically adjusted in the training process of the network.

To our best knowledge, current QA systems often directly align the embedding vector for each word, but without considering the mixture of the semantic subspaces spanned by the embedding vectors. With such a mixture space, we will show that some useful similarity features/patterns will be derived in our neural network based architecture. We can also interpret the density matrix in Eq. 3 from the perspective of covariance matrix. The density matrix to some extent reflects the covariance of different embedding dimensions for a sentence. In other words, it represents how scattered the words (in the sentence) will be in the embedded space.

Joint Representation for Question Answer Pair

Instead of separately modeling/projecting a text into one dimensional vector and then computing a distance-based score between a pair of text fragments, two dimensional matching model which uses a joint representation (a matrix or multi-dimension tensor) have proven more effective (Hu et al. 2014; Wan et al. 2016). Based on the density matrices for a question and an answer, we can build a joint representation for modeling the interaction between two density matrices by their multiplication:

$$\boldsymbol{M}_{qa} = \boldsymbol{\rho}_q \boldsymbol{\rho}_a \tag{4}$$

where $\boldsymbol{\rho}_q$ and $\boldsymbol{\rho}_a$ are the density matrix representation for the question q and the answer a, respectively.

In order to analyze the property of this joint representation, we can first decompose the density matrix of the query through spectral decomposition:

$$\boldsymbol{\rho}_q = \sum_i \lambda_i \, |r_i\rangle\langle r_i| \tag{5}$$

where λ_i is an eigenvalue and $|r_i\rangle$ is the corresponding eigenvector. The eigenvector can be interpreted as a latent semantic basis, and the eigenvalue can reflect how scattered the words are in the corresponding basis. Similarly, the answer density matrix ρ_a can be decomposed into $\boldsymbol{\rho}_a = \sum_j \lambda_j \, |r_j\rangle\langle r_j|$. Then the joint representation between $\boldsymbol{\rho}_q$ and $\boldsymbol{\rho}_a$ can be written as:

$$\begin{aligned}\boldsymbol{\rho}_q \boldsymbol{\rho}_a &= \sum_{i,j} \lambda_i \lambda_j \, |r_i\rangle\langle r_i|r_j\rangle\langle r_j| \\ &= \sum_{i,j} \lambda_i \lambda_j \, \langle r_i|r_j\rangle |r_i\rangle\langle r_j|\end{aligned} \tag{6}$$

In Eq. 6, the more similar two bases are, the bigger the $\langle r_i|r_j\rangle$ (representing the inner product and the Cosine similarity between $|r_i\rangle$ and $\langle r_j|$) is. Since $\langle r_i|r_j\rangle = \mathrm{tr}(|r_i\rangle\langle r_j|)$, we have

$$\mathrm{tr}(\boldsymbol{\rho}_q \boldsymbol{\rho}_a) = \sum_{i,j} \lambda_i \lambda_j \, \langle r_i|r_j\rangle^2 \tag{7}$$

which is the sum of Cosine similarities of between the latent semantic dimensions. In this way, the joint representation can retain the distribution of similar bases and ignore the dissimilar ones. More generally, $\mathrm{tr}(\boldsymbol{\rho}_q \boldsymbol{\rho}_a)$ is a generalization of inner product from vectors to matrices, which is called trace inner product (Balkır 2014). Thus, the joint representation matrix $\boldsymbol{M}_{qa}$ encodes the similarity information across the question and the answer.

Learning to Match Density Matrices

Based on the above ideas, we propose two Neural Network based Quantum-like Language Models (NNQLM) to match the question-answer pairs.

NNQLM-I As shown in Figure 2, the first architecture is designed to allow a direct and intuitive comparison with the original QLM. In QLM, the similarity between a question and an answer is obtained by the negative VN divergence. However, because of the log operation for the matrix, the negative VN divergence is hard to be integrated into an end-to-end approach. In this paper, we adopt the trace inner product. Formally, the trace inner product between two density matrices $\boldsymbol{\rho}_q$ and $\boldsymbol{\rho}_a$ (for a question q and an answer a, respectively) can be formulated as:

$$S(\boldsymbol{\rho}_q, \boldsymbol{\rho}_a) = \mathrm{tr}(\boldsymbol{\rho}_q \boldsymbol{\rho}_a) \tag{8}$$

Trace inner product has been used to calculate the similarity between words or sentences (Blacoe, Kashefi, and Lapata 2013; Blacoe 2014) and has been proven to be an approximation of the negative VN divergence (Sordoni, Bengio, and Nie 2014). As aforementioned, it can be rewritten as $x_{trace} = \mathrm{tr}(\boldsymbol{\rho}_q \boldsymbol{\rho}_a) = \sum_{i,j} \lambda_i \lambda_j \, \langle r_i|r_j\rangle^2$, which can be understood as the semantic overlaps used to compute the similarity between the density matrices of the question and the answer. In addition, the diagonal elements (forming $\vec{x}_{diag}$) of $\boldsymbol{M}_{qa}$ are also adopted to enrich the feature representation, since different diagonal elements may have different degrees of importance for similarity measurement. Then, the feature representation can be denoted as:

$$\vec{x}_{feat} = [x_{trace}; \vec{x}_{diag}] \tag{9}$$

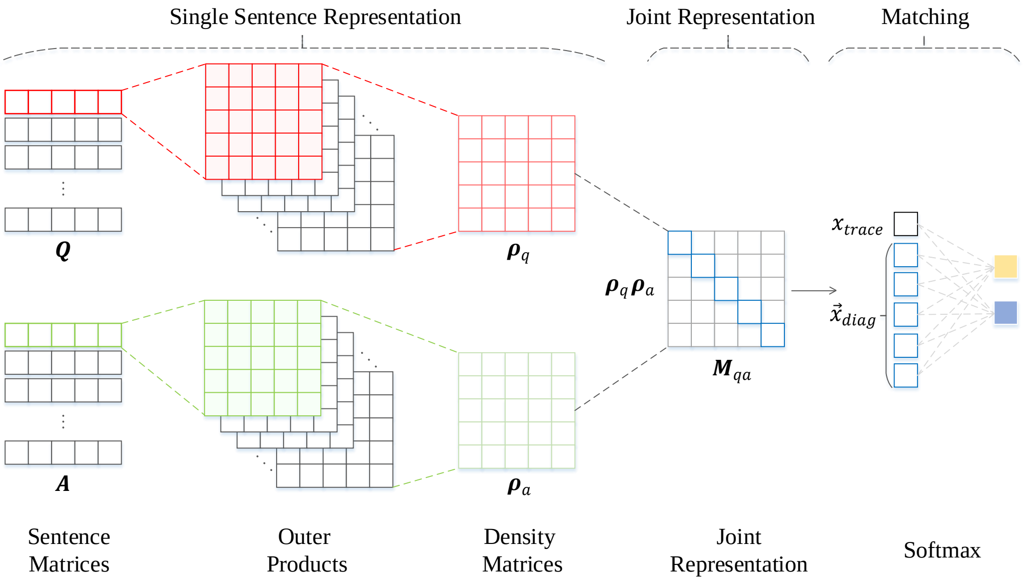

Figure 2: NNQLM-I. The first three layers are to obtain the single sentence representation, the fourth layer is to obtain the joint representation of a QA pair, and the softmax layer is to match the QA pair.

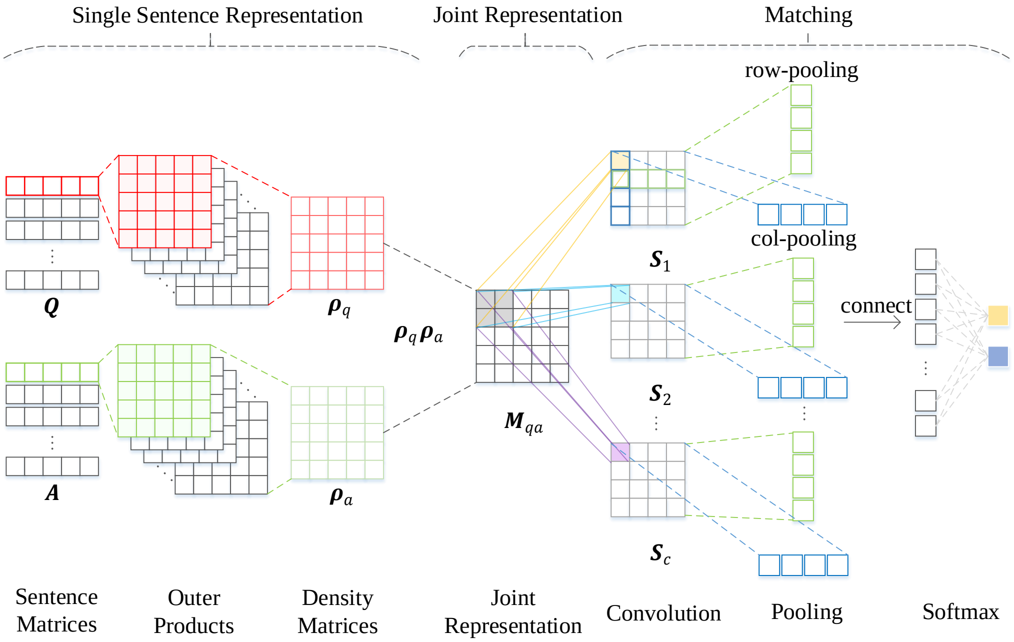

Figure 3: NNQLM-II. The single sentence representation and the joint representation are the same as those in NNQLM-I, and the rest layers are to match the QA pair by the similarity patterns learned by 2D-CNN.

A fully-connected layer and a softmax activation are adopted. The outputs of the softmax layer are the probabilities of positive and negative labels of an answer. The probability of the positive label is used as the similarity score for ranking. The back propagation is trained with the negative cross entropy loss:

$$\boldsymbol{L} = -\sum_{i}^{N}[y_i \log h(\vec{x}_{feat}) + (1-y_i)\log(1-h(\vec{x}_{feat}))] \tag{10}$$

where $h(\vec{x}_{feat})$ is the output from softmax. In this way, we extend the original QLM to an end-to-end method.

NNQLM-II In this architecture, we will adopt a "two-dimensional" (2D) convolution (Hu et al. 2014) to learn relatively more abstract representations, which are different from the intuitive features (e.g., trace inner product x_{trace} which is a similarity measure) in NNQLM-I. We think the 2D convolution is more suitable for the joint representation $\boldsymbol{M}_{qa}$, as $\boldsymbol{M}_{qa}$ is *not* a simple concatenation of word embedding vectors. For the simple concatenation representation, the convolution kernels slide only along each single dimension, so that the corresponding convolution can be considered as "one-dimensional" (1D) convolution (Hu et al. 2014), which is actually used in many CNN-based QA models (Severyn and Moschitti 2015; Yu et al. 2014; Kim 2014).

The second architecture, namely NNQLM-II, is shown in Figure 3. Recall that in the first architecture, only the diagonal values and the trace value of the joint representation $\boldsymbol{M}_{qa}$ are involved in the training process (see Figure 2). In NNQLM-II, we use 2D convolution kernels to scan all the local parts (including the diagonal elements in the first architecture) of the joint representation and extract/filter as many similarity patterns as possible in $\boldsymbol{M}_{qa}$.

Suppose the number of filters is c. The i^{th} convolution operation is formulated as:

$$\boldsymbol{C}_i = \delta(\boldsymbol{M}_{qa} * \boldsymbol{W}_i + b_i) \tag{11}$$

where $1 \leqslant i \leqslant c$, δ is the non-linear activation function, $*$ denotes the 2D convolution, $\boldsymbol{W}_i$ and b_i are the weight and the bias respectively for the i^{th} convolution kernel, and $\boldsymbol{C}_i$ is the feature map. After the convolution layer obtains the feature maps, we then use row-wise and column-wise max-pooling to generate vectors $\vec{r}^{q}_{i} \in \mathbb{R}^{d-k+1}$ and $\vec{r}^{a}_{i} \in \mathbb{R}^{d-k+1}$, respectively, with the formulations as follows:

$$\begin{aligned} \vec{r}^{q}_{i} &= (q_j : j = 1, 2, \ldots, d-k+1) \\ q_j &= \max_{1 \leq m \leq d-k+1}(\boldsymbol{C}_{i(j,m)}) \end{aligned} \tag{12}$$

$$\begin{aligned} \vec{r}^{a}_{i} &= (a_j : j = 1, 2, \ldots, d-k+1) \\ a_j &= \max_{1 \leq m \leq d-k+1}(\boldsymbol{C}_{i(m,j)}) \end{aligned} \tag{13}$$

We concatenate these vectors as follow:

$$\vec{x}_{feat} = [\vec{r}^{q}_{1}; \vec{r}^{a}_{1}; \ldots; \vec{r}^{q}_{i}; \vec{r}^{a}_{i}; \ldots; \vec{r}^{q}_{c}; \vec{r}^{a}_{c}] \tag{14}$$

where $1 \leqslant i \leqslant c$. The above convolution operation aims to extract useful similarity patterns and each convolution kernel corresponds to a feature. We can adjust the parameters of the kernels when the model is being trained.

Related Work

Now, we present a brief review of the related work, including the recent quantum-inspired work in Information Retrieval (IR) and Natural Language Processing (NLP), and some representative work in Question Answering.

Quantum-inspired Models for IR and NLP

van Rijsbergen (2004) argued that quantum theory can unify the logical, geometric, and probabilistic IR models by its mathematical formalism. After this pioneering work, a range of quantum-inspired methods have been developed (Zuccon and Azzopardi 2010; Piwowarski et al. 2010; **?**), based on the analogy between IR ranking and quantum phenomena (e.g., double-slit experiment). Quantum theory has been also used for semantic representation in combination with Dependency Parsing Tree (Blacoe, Kashefi, and Lapata 2013), where state vectors represent word meanings and density matrices represent the uncertainty of word meanings.

Sordoni, Nie, and Bengio (2013) successfully applied quantum probability in Language Modeling (LM) and proposed a Quantum Language Model (QLM), for which the estimation of the density matrix is crucial. It is proven that the density matrix is a more general representation for texts, by looking at vector space model and language model in the quantum formalism (Sordoni and Nie 2013). Using the idea of quantum entropy minimization in QLM, Sordoni, Bengio, and Nie (2014) proposed to learn latent concept embeddings for query expansion. By devising a similarity function in the latent concept space, the query representation will get closer to the relevant document terms, thus benefiting the likelihood of selecting good expansion terms. Indeed, this work inspires us to develop QLM towards a supervised approach and estimating density matrix analytically. Compared with this work, our proposed model targets different application task, and uses different approaches to density matrix estimation and learning architectures.

More recently, a session-based adaptive QLM was proposed to model the evolution of density matrix and capture the dynamic information need in search sessions (Li et al. 2015; 2016). The concept of quantum entanglement has also been integrated into the quantum language model (Xie et al. 2015). Practically, the so-called pure high-order term association patterns (as a reflection of entanglement) are selected as the compound terms for the input of density matrix estimation. The above variants of QLM keeps the main architecture of QLM. In other words, they still carry out the representation, estimation and ranking processes sequentially, without a joint optimization strategy. Thus their potential to improve the retrieval effectiveness is limited.

Answer Selection

In this paper, we apply the proposed end-to-end QLM in the answer selection task. The aim of answer selection is to find correct answer sentences from pre-selected answer candidates given a question. In the answer selection task, end-to-end methods represent the current state of the art.

Yu et al. (2014) used CNN to capture bigram information in the question/answer. Severyn and Moschitti (2015)

Table 1: Statistics of TREC-QA and WikiQA

	TREC-QA			WIKIQA		
	TRAIN	DEV	TEST	TRAIN	DEV	TEST
#Question	1229	82	100	873	126	243
#Pairs	53417	1148	1517	8672	1130	2351
%Correct	12.0	19.3	18.7	12.0	12.4	12.5

further developed this idea and capture n-gram of higher order dependency. Qiu and Huang (2015) also modeled n-gram information in a single sentence and model the interactions between sentences with a tensor layer. Later, more effective components are added to the C-NN model, such as attention mechanism (Yin et al. 2015; dos Santos et al. 2016) and Noise-Contrastive Estimation (NCE) (Rao, He, and Lin 2016). Long-Short Term Memory (LSTM) has also been utilized (Wang and Nyberg 2015; Tay et al. 2017).

Our work is the first attempt to introduce Quantum Language Model (QLM) in the answer selection task. We will show that the density matrix representation has a great potential to effectively encode the mixture of semantic subspaces and reflect how scattered the words of a sentence are in the embedded space. To our best knowledge, such representation for sentences and a further joint representation for two sentences, are different from those representations used in the aforementioned end-to-end QA approaches. In addition, different from existing QLM based models, we design a new method to obtain density matrices and propose an end-to-end model to integrate the density matrix representation and the similarity matching into neural network structures.

Experiment

Datasets and Evaluation Metrics

Extensive experiments are conducted on the TREC-QA and WikiQA datasets. TREC-QA is a standard benchmarking dataset used in the Text REtrieval Conference (TREC)'s QA track (8-13) (Wang, Smith, and Mitamura 2007). WikiQA (Yang, Yih, and Meek 2015) is an open domain question-answering dataset released by Microsoft Research, and we use it for the subtask assuming there is at least one correct answer for each question. The basic statistics of the datasets are presented in Table 1. The evaluation metrics we use are mean average precision (MAP) and mean reciprocal rank (MRR), which are commonly used in previous works for the same task with the same datasets.

Methods for Comparison and Parameter Settings

The methods for comparison are as follows. QLM is the original quantum language model while QLM$_T$ replaces non-negative VN divergence with trace inner product as the ranking function. NNQLM-I and NNQLM-II are our proposed end-to-end QLMs.

For QLM, we initialize $\boldsymbol{\rho}_0$ by a diagonal matrix, in which the diagonal elements are Term Frequency (TF) values of the corresponding words. The initial matrices are normalized with a unit trace. The size of sliding window is 5.

For NNQLMs, the hyper parameters are listed in Table 2. The parameters that need to be trained are the position weights p_i in density matrices (see Eq. 3), the weights of the softmax layer, and the parameters for the convolution layer. The word embeddings are trained by word2vec (Mikolov et al. 2013) on English Wikimedia[1]. The dimensionality is 50, and the Out-of-Vocabulary words are randomly initialized by a uniform distribution in the range of (-0.25, 0.25). In NNQLM-I, the embeddings are updated during training. For NNQLM-II, we keep the embeddings static, which is found in our pilot experiments to outperform dynamically updating the embeddings.

Results

A series of experiments are carried out for a systematic evaluation. Table 3 summarizes the experiment results.

In the first group, we compare QLM with QLM$_T$. On both TREC-QA and WIKIQA, QLM achieves similar results to QLM$_T$. This is consistent with our previous explanation that the trace inner product is an approximation of negative VN divergence. Thus, it is reasonable to use it as a similarity measure in our first architecture (NNQLM-I) based on the joint representation.

In the second group, we compare two NNQLMs with QLM. NNQLM-I can outperform QLM, which shows the effectiveness of our new definition of the density matrix together with the simple training algorithm. Recall that in NNQLM-I, only the diagonal and trace values of the joint representation are involved in the training process.

NNQLM-II, which uses 2D convolution neural network (2D-CNN for short), largely outperforms NNQLM-I. It verifies our earlier analysis that the 2D-CNN is able to learn richer similarity features than those learned from the first architecture. It also implies that the joint representation of density matrices has a great potential as a kind of representation for learning effective inter-sentence similarity information.

As we can see from Table 3, NNQLM-II can significantly improve the original QLM on both datasets (by 11.87% MAP and 13.61% MRR on TREC-QA, and by 27.15% MAP and 28.09% on WIKIQA). The significant test is performed using the Wilcoxon signed-rank test with $p<0.05$.

In the third group, we compare NNQLM-II with the existing neural network based approaches (Yu et al. 2014; Severyn and Moschitti 2015; Yin et al. 2015; Wang and Nyberg 2015; Yang et al. 2016)[2]. On TREC-QA, NNQLM-

[1] https://dumps.wikimedia.org/

[2] The WIKIQA results of the methods proposed in (Yu et al. 2014) and (Yin et al. 2015) are extracted from (Yang, Yih, and Meek 2015) and (dos Santos et al. 2016), respectively, excluding handcrafted features.

Table 2: The Hyper Parameters of NNQLM

Method	TREC-QA		WIKIQA	
	NNQLM-I	NNQLM-II	NNQLM-I	NNQLM-II
learning rate	0.01	0.01	0.08	0.02
batchsize	100	100	100	140
filter number	/	65	/	150
filter size	/	40	/	40

Table 3: Results on TREC-QA and WIKIQA

Method	TREC-QA		WIKIQA	
	MAP	MRR	MAP	MRR
QLM	0.6784	0.7265	0.5109	0.5148
QLM$_T$	0.6683	0.7280	0.5108	0.5145
NNQLM-I	0.6791	0.7529	0.5462	0.5574
NNQLM-II	**0.7589**	**0.8254**	0.6496	0.6594
(Yu et al. 2014)	0.5693	0.6613	0.6190	0.6281
(Severyn and Moschitti, 2015, 2016)	0.6709	0.7280	**0.6661**	**0.6851**
(Yin et al. 2015)	/	/	0.6600	0.6770
(Wang and Nyberg 2015)	0.5928	0.5928	/	/
(Yang et al. 2016)	0.7407	0.7995	/	/

II achieves the *best* performance over the comparative approaches. Note that NNQLM-II outperforms a strong baseline (Yang et al. 2016) by 2.46% MAP and 3.24% MRR. On WikiQA, NNQLM-II has outperformed a baseline method proposed (Yu et al. 2014), for which its WikiQA results are reported in (Yang, Yih, and Meek 2015).

Discussions

NNQLM-II has not yet outperformed the other two baselines on WikiQA. The possible reason is that although 2D-CNN can learn useful *similarity* patterns for the QA task, there can be other useful features (e.g, the structure in a sentence) that could influence the performance. In fact, We can add some other features and get improvements on WikiQA. In our future work, we will improve the network structure of NNQLM-II for a better model.

In addition, we would like to further explain the mechanism of our model in comparison with the existing QA models. The difference of our proposed model stems from the density matrix representation. Such a matrix can represent the mixture of the semantic subspaces, and the joint representation of question and answer matrices can encode similarity patterns. By using the 2D-CNN, we can extract useful similarity patterns and obtain a good performance on the answer selection task. On the other hand, most existing neural network based QA models concatenates word embedding vectors. Based on such concatenation, the 1D convolution neural networks (1D-CNN for short) can be directly performed. We have carried out the above experiments for a comparison. In the future, we will systematically analyze and evaluate the above two different mechanisms in-depth.

Conclusions and Future Work

In this paper, we have proposed an Neural-Network based Quantum-like Language Models (namely NNQLM), which substantially extend the original Quantum Language Model (QLM) to an end-to-end mechanism, with application to the Question Answering (QA) task. To the best of our knowledge, this is the first time for the QLM to be extended with neural network architectures and for the quantum or quantum-like models to applied to QA. We have designed a new density matrix based on word embeddings, and such a density matrix for a single sentence, together with the joint representation for sentence pairs, can be integrated into the neural network architectures for an effective joint training.

Systematic experiments on TREC_QA and WikiQA have demonstrated the applicability and effectiveness of QLM and NNQLMs. Our proposed NNQLM-II achieves a significant improvement over QLM on both datasets, and outperforms a strong baseline (Yang et al. 2016) by 2.46% (MAP) and 3.24% (MRR) on TREC-QA.

A straightforward future research direction is to explore other neural networks for NNQLM. Our models can also be applied to other tasks, especially for short text pair matching tasks. In addition, for those QA pairs that are not selected by the similarity matching, e.g., in a casual inference case, we can encode an additional component (e.g., an inference function) in the quantum-like models. It is also interesting to explore the capability of NNQLMs using density matrices for the representation learning task.

Acknowledgements

This work is supported in part by the Chinese National Program on Key Basic Research Project (973 Program, grant No. 2014CB744604), Natural Science Foundation of China (grant No. U1636203, 61772363, 61402324), Tianjin Natural Science Foundation (grant No. 15JCQNJC41700), and European Union' Horizon 2020 research and innovation programe under the Marie Skłodowska-Curie grant agreement No. 721321.

References

Balkır, E. 2014. *Using density matrices in a compositional distributional model of meaning*. Ph.D. Dissertation, Masters thesis, University of Oxford.

Bengio, Y.; Ducharme, R.; Vincent, P.; and Janvin, C. 2003. A neural probabilistic language model. *Journal of Machine Learning Research* 3:1137–1155.

Blacoe, W.; Kashefi, E.; and Lapata, M. 2013. A quantum-theoretic approach to distributional semantics. In *HLT-NAACL*, 847–857.

Blacoe, W. 2014. Semantic composition inspired by quantum measurement. In *Proc. of QI*, 41–53. Springer.

dos Santos, C. N.; Tan, M.; Xiang, B.; and Zhou, B. 2016. Attentive pooling networks. *CoRR, abs/1602.03609*.

Gleason, A. M. 1957. Measures on the closed subspaces of a hilbert space. *Journal of mathematics and mechanics* 6(6):885–893.

Hu, B.; Lu, Z.; Li, H.; and Chen, Q. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, 2042–2050.

Li, Q.; Li, J.; Zhang, P.; and Song, D. 2015. Modeling multi-query retrieval tasks using density matrix transformation. In *Proc. of SIGIR*, 871–874. ACM.

Liu, X.; Bouchoucha, A.; Sordoni, A.; and Nie, J. 2014. Compact aspect embedding for diversified query expansions. In *AAAI Press*, 115–121.

Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111–3119.

Piwowarski, B.; Frommholz, I.; Lalmas, M.; and van Rijsbergen, K. 2010. What can quantum theory bring to information retrieval. In *Proc. of CIKM*, 59–68.

Rao, J.; He, H.; and Lin, J. 2016. Noise-contrastive estimation for answer selection with deep neural networks. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 1913–1916. ACM.

Severyn, A., and Moschitti, A. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proc. of SIGIR*, 373–382. ACM.

Severyn, A., and Moschitti, A. 2016. Modeling relational information in question-answer pairs with convolutional neural networks. *arXiv preprint arXiv:1604.01178*.

Sordoni, A., and Nie, J.-Y. 2013. Looking at vector space and language models for ir using density matrices. In *International Symposium on Quantum Interaction*, 147–159. Springer.

Sordoni, A.; Bengio, Y.; and Nie, J.-Y. 2014. Learning concept embeddings for query expansion by quantum entropy minimization. In *AAAI*, volume 14, 1586–1592.

Sordoni, A.; Nie, J.-Y.; and Bengio, Y. 2013. Modeling term dependencies with quantum language models for ir. In *Proc. of SIGIR*, 653–662. ACM.

Van Rijsbergen, C. J. 2004. *The geometry of information retrieval*. Cambridge University Press.

Von Neumann, J. 1955. *Mathematical foundations of quantum mechanics*. Number 2. Princeton university press.

Wang, D., and Nyberg, E. 2015. A long short-term memory model for answer sentence selection in question answering. In *ACL (2)*, 707–712.

Wang, M.; Smith, N. A.; and Mitamura, T. 2007. What is the jeopardy model? a quasi-synchronous grammar for qa. In *EMNLP-CoNLL*, volume 7, 22–32.

Xie, M.; Hou, Y.; Zhang, P.; Li, J.; Li, W.; and Song, D. 2015. Modeling quantum entanglements in quantum language models.

Yang, L.; Ai, Q.; Guo, J.; and Croft, W. B. 2016. anmm: Ranking short answer texts with attention-based neural matching model. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 287–296. ACM.

Yang, Y.; Yih, W.-t.; and Meek, C. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, 2013–2018. Citeseer.

Yin, W.; Schütze, H.; Xiang, B.; and Zhou, B. 2015. Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *arXiv preprint arXiv:1512.05193*.

Yu, L.; Hermann, K. M.; Blunsom, P.; and Pulman, S. 2014. Deep learning for answer sentence selection. *arXiv preprint arXiv:1412.1632*.

Zhai, C. 2008. *Statistical Language Models for Information Retrieval*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Zuccon, G., and Azzopardi, L. 2010. Using the quantum probability ranking principle to rank interdependent documents. In *ECIR*, 357–369.

Zuccon, G. 2013. Document ranking with quantum probabilities. *SIGIR Forum* 47(1):69–70.