Quantum Natural Language Processing: A New and Promising Way to Solve NLP Problems

Yousra Bouakba(✉) and Hacene Belhadef

LISIA Laboratory, University of Abdelhamid MEHRI, Constantine, Algeria
{yousra.bouakba,hacene.belhadef}@univ-constantine2.dz

Abstract. Natural Language Processing (NLP) has known important interest and growth in recent years as may witness the increasing number of publications in different NLP tasks over the world. The primary focus of recent research has been to develop algorithms that process natural language in quantum computers, hence the emergence of new subdomain called QNLP were proposed. Hence the objective of this paper is to provide to NLP researchers a new vision and way to deal with the NLP problems basing on Quantum computing techniques. The present paper aims to provide a list of existing alternatives and classify them by; implementation on classical or quantum hardware, theoretical or experimental work and representation type of sentence. Our study focuses on the Distributional Compositional Categorical model (DisCoCat), from its mathematical and theoretical demonstration into its experimental results.

Keywords: Quantum computing · Natural language processing · Quantum machine learning · Quantum natural language processing

1 Introduction

Natural Language Processing is at the heart of most information processing tasks. It has a long history in computer science and has always been a central discipline in artificial intelligence. With the availability of data and the advancement of processing technology (such as GPUs, TPUs,...), we have recently witnessed a rising success of machine algorithms and deep learning, that marked the renewal of AI and the beginning of its new age. But a new vision to solve NLP problems is to use quantum theory in order to accelerate the process of building an AI model. As a result, quantum machine learning algorithms has involved the natural language processing (NLP) field. This has created the so-called Quantum Natural Language Processing (QNLP). Following that, many works are proposed which implement QNLP on either classical or quantum computers.

The purpose of this paper would provide a general overview of quantum approaches to solve natural language processing tasks with a strong attention

M. Salem et al. (Eds.): ICAITA 2022, CCIS 1769, pp. 215–227, 2023.
https://doi.org/10.1007/978-3-031-28540-0_17

on DisCoCat Model. Firstly, we presenting the most major limitations of current NLP models !. After a summary introduction of quantum computing fundamentals, the connection between quantum and NLP is described. Subsequently, quantum algorithms are listed by their type: implemented in quantum or classical computers and distinguishing between their sentence representation. Focusing on DisCoCat model, we present detailed experimental results of three QNLP application: question-answering, text classification and machine translation. Finally, important libraries for QNLP implementation are discussed.

This paper is structured as follows: Sect. 2, present limits of current NLP models. Then in Sect. 3, introduce the required foundation for understanding the intersection between NLP and quantum computing. First in Sect. 3.1, quantum computing fundamentals are described. And Sect. 3.2, Similarity points between NLP and Quantum theory are specified. Moreover, in Sect. 4 quantum algorithms per type are listed, divided into quantum-inspired algorithms (Sect. 4.1): full theoretical or algorithms with classical implementation, and quantum algorithms (Sect. 4.2): "bag-of-word" and DisCoCat models. Section 5, provides the two main QNLP toolkit: Lambeq and TensorFlow quantum.

2 Limits of Current NLP Models

Transfer learning and the application of Pre-Trained Language Models to varied NLP tasks have identified as the primary directions in current NLP research works. It is well known that more dataset and more parameters are essential for training deep transformers from scratch such as, GPT-3 which has used 175 billion parameters, 96 attention layers, and a Common Crawl data-set that is 45 TB in size [7]. However, it is costly in terms of time, resources, and processing power.

While some IA specialists, such as Anna Rogers, believe that gaining achievements by just exploiting additional data and computer power is not new research, it is a SOTA (State-Of-The-Art): it refers to the best models that can be used for achieving the results in an AI-specific task only [17].

Furthermore, because of their impractical resource requirements, these models are difficult to adapt to real-world business challenges. Several studies, on the other hand, have been carried out to understand if neural language models effectively encode linguistic information! or just replicate patterns observed in written texts! [9] [3]. With both of these reservations about current models, the lookout for new methodologies has taken priority in the field.

3 NLP and Quantum Computing

A number of recent academic contributions investigate the notion of using quantum computing advantages to improve machine learning methods. It has resulted an increasing number of strong applications in fields including NLP, cryptography [22].etc. QNLP is a branch of quantum computing that Implementing quantum algorithms for NLP problems.

3.1 Quantum Computing Fundamentals

This section introduces basic quantum computing fundamentals for a better understanding of QNLP.

Quantum Computer and Qubits. In quantum computing, qubits are the basic unit of information. A qubit can be 0, 1 or a superposition of both and represented using the Dirac notation: $|0>$ and $|1>$.

Superposition. A qubit can be represented by the linear combination of states:

$$|\psi >= a|0> +b|1> \tag{1}$$

where a and b are complex numbers and

$$|a|^2 + |b|^2 = 1 \tag{2}$$

When we measure a qubit we obtain either 0, with probability $|a|^2$, or 1, with probability $|b|^2$.

Entanglement. Bell states are specific quantum states of two qubits representing the simplest and maximal examples of quantum entanglement.

$$\frac{|00> +|11>}{\sqrt{2}} \tag{3}$$

Due to the entanglement property, the second qubit must now obtain exactly the same measurement as the first qubit because the measurement results of these two entangled qubits are correlated.

Measurement. When a qubit is in a superposition state, once we measure it collapse the superposition state and takes either 1 or 0.

QRAM: Quantum Random Access Memory. Is the quantum equivalent of classical random access memory (RAM). QRAM architecture has been proposed by [8] using n qubits to address any quantum superposition of N memory cells where a classical RAM uses n bits to randomly address $N = 2^n$ distinct memory cells. However, it is still unachievable at the implementation level.

NISQ Device: Noise Intermediate-Scale Quantum. Is a quantum hardware to run quantum algorithms with a maximum memory size of 100 qubits. It's "Noisy" because there aren't enough qubits for error correction. And "Intermediate-Scale" because the quantity of quantum bits is insufficient to calculate advanced quantum algorithms but sufficient to demonstrate quantum advantage [16]

3.2 Why NLP Is Quantum-Native?: Similarity Points

The term quantum-native appears often in several literature, showing the intuitive relationship between quantum computing and NLP; we outline this connection below:

1. A word multiple meaning is a superposition state; where each meaning represent a quantum state. for example the word "apple" can have different meanings, it can indicate Fruit or Enterprise depending the context. Using the Dirac notation it can be represented as a superposition state:

$$|apple >= a|fruit > +b|enterprise > \tag{4}$$

   the representation above means, the probability of being a fruit is $|a|^2$ and the probability of being an enterprise is $|b|^2$.
2. Context of sentence act like measurement; once a word is observed in context, one of the available meanings is typically selected. for example, "I received a phone from Apple". When it appears in the context of purchasing, it collapses to a fixed meaning. where Apple refers to Enterprise.
3. The grammatical structure entangles words: grammar is what connects the meanings of words and words are encoded as quantum state, then the grammatical structure is to entangle these states. [4]
4. Vector spaces and linear algebra are very used in NLP and quantum mechanics.

4 Quantum Algorithms Types

There are two types of quantum algorithms: those that can be implemented with classical computers and those that can only be executed with quantum computers.

4.1 Quantum-inspired/Quantum-like Algorithms

Several types of work are concerted quantum-inspired, such as full theoretical work that has never had a real implementation and work based on quantum physics but executing on classical hardware.

Full Theoretical Quantum Approaches: In terms of theory, [23] provides a method for implementing distributional compositional models, such as the Distributional Compositional Categorical model (DisCoCat), on quantum computers that is based on the usage of unavailable QRAM. This study demonstrates theoretically improved performance in sentence similarity. which will be applicable once QRAM is released.

To overcome for this unavailable QRAM limitation, [5] [13] papers proposes a theoretical full-stack NLP pipeline using a NISQ device that makes use of the classical ansätz parameters without the requirement of QRAM. Because they are followed by experimental proofs (Sect. 4.2), these theoretical approaches are considered as being the most fundamental works in the QNLP field.

Quantum-Inspired Algorithms: Quantum-inspired or quantum-like algorithms adopt mathematical foundations from quantum theory, although they are built to run on classical computers rather than quantum computers. A general method of information retrieval is proposed by [18] which models term dependencies using density matrix for more general representation of text. Paper [11] is another quantum learning model to information retrieval that has been presented in includes a query expansion framework to overcome limits due to limited vocabulary. based on [18], reference [2] implement the same general method for a speech recognition, in addition to using the evolution of the state.

Question Answering (QA) is another task where the use of quantum-inspired approaches have addressed the task in very different ways. In [24], A Neural Network-based Quantum-like Language Model (NNQLM) has been developed and utilizes word embedding vectors as the state vectors, from which a density matrix is extracted and integrated into an end-to-end Neural Network (NN) structure. An connection between the quantum many-body system and language modeling proposed by [25]. Text classification, and sentiment analysis [27] [26], is another NLP applications that has benefited from various quantum-based approaches.

4.2 Quantum Algorithms

The categorization of algorithms is not limited to their ability to be implemented on a quantum computer but also in terms of how they represent a sentence!There are various approaches to achieving this in literature including, sentence is a "bag-of-word", map words to vector [11] [4] [20] and map words to matrix [18]. This section concentrates on two different methods: "bag-of-word model" and "DisCoCat model".

Bag-of-word Model. The term'bag-of-words' refers to the use of "just" individual words as classifier features: The position of words, or which combination of words exist, is ignored. Reference [15] proposed a"bag-of-word" classifier with accuracy 100. The following is a full explanation of the bag-of-word classifier: training and classification phases.

Training phase:

1. Each (word, topic) pair is assigned to a specific qubit
2. Every time a given word occurs with a given topic, the (word, topic) weights increment the rotation for the corresponding qubit.
3. An additional qubit is declared for each topic in order to keep the summation for each topic.

Classification phase:

1. Pre-processing step, identifies the most common words for all topics.
2. Each detected word in the sentence has its topic weights associated with the sum qubit for that topic.
3. Each of the topic qubits is measured, and the winner is the topic that measures the most $|1>$ states over a number of shots.

A circuit implementing this process is shown in Fig. 1.

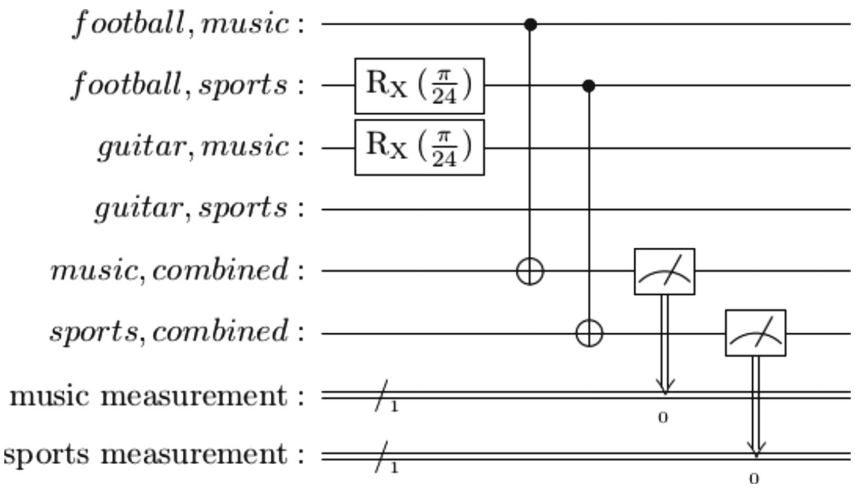

Fig. 1. Classification circuit of two words and two topics [21]

As a result, this method is only used to demonstrate extremely small vocabularies because the number of qubits required is (number words + 1) * number topics [21]. This model treating sentence as a structure-less "bag" containing the meanings of individual words.

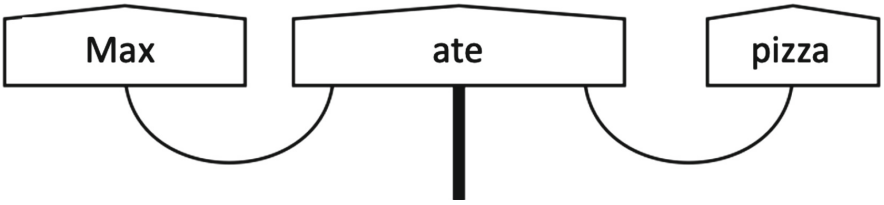

Fig. 2. A sample sentence using string diagram [6]

DisCoCat Model. A sentence in the DisCoCat model is not just a "bag-of-words" but also taking into account types and grammatical structure. In this model, using a string diagram to represent the meaning of words by tensors whose order is determined by the types of words, expressed in a pregroup grammar [6]. This graphical framework use boxes to represents the meaning of words that are transmitted via wires. Figure 2 shows that, the subject noun "Max" and the object noun "pizza" are both connected to the verb "ate" and the combination of these words contributes to the overall meaning of the sentence. In quantum terms, [5] provide a diagrammatic notation in which sentence meaning is independent to grammatical structure (see Fig. 3).

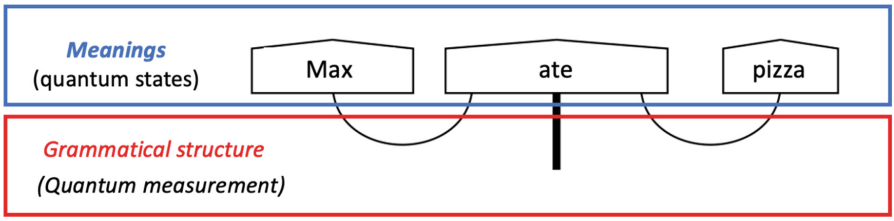

Fig. 3. Diagrammatic notation: word meaning is quantum states and grammatical structure is quantum measurements [5].

The DisCoCat model real origin is the categorical quantum mechanics (CQM) formalism [1]. As a result, it is natural to assume that it is suitable for quantum hardware.

Following to [13], The first NLP Questions-Answering task implementation on NISQ hardware has been proposed by [14] using a labeled dataset of 16 randomly generated phrases with 6 words vocabulary then the circuit runs on two different IBM quantum computers names respectively, *ibmq_ montreal* and *ibmq_ toronto*. The experimental results are, a train error of 12.5% and a test error of 37.5% on *ibmq_ toronto* and, a train error of 0% while the test error is 37.5% on *ibmq_ montreal*.

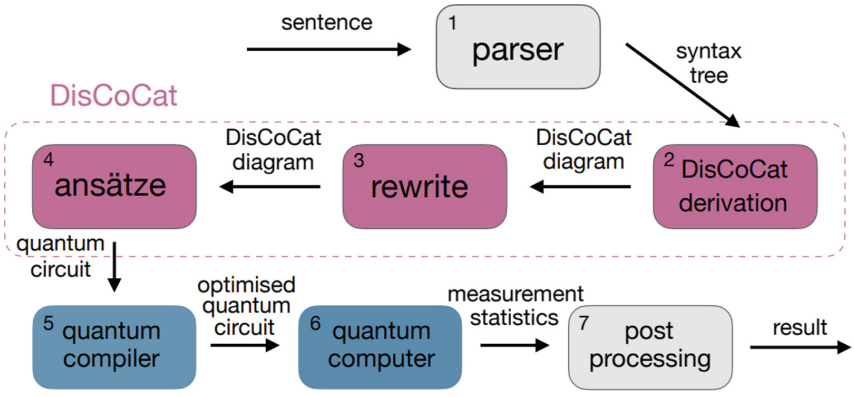

Fig. 4. The general quantum pipeline [12]

Therefore, to explore the challenges and limitations of training and running an NLP model on a NISQ device at a medium-scale, [12] propose the first quantum pipeline equivalent to the classical one (see Fig. 4).

This pipeline has been tested on two different tasks, meaning classification task and prediction task. The first task, Meaning classification(MC) uses a dataset of 130 sentences plain-syntax generated from a 17 words vocabulary using a simple CFG that can refer to one of two possible topics, food or IT. MC running on *ibmq_ bogota* has achieving a train error of 0% and a test error of 16.78%.

The second task, is to predict whether a noun phrase contains a subject-based or an object-based relative clause. This task rise the challenge by using 105 noun phrases are extracted from the RelPron dataset with 115 words vocabulary. The experimental results are, a train error of 0% and a test error of 32.3 % on *ibmq_ bogota*.

Machine translation is an another task that has benefited from the DisCoCat model. [19] work trying to use DisCoCat for Spanish-English translation. The work achieved a high percentage (95 %) of sentence similarity between the two languages and good result even when the complexity and length of the sentence increase.

5 QNLP Implementation Tools

The objective of quantum language processing toolkit and libraries is to enable real-world quantum natural language processing applications. Currently there are two ways to implement QNLP tasks: TensorFlow quantum or Lambeq libraries.

5.1 TensorFlow Quantum

Is a framework that uses the NISQ Processors for development of hybrid quantum-classical ML models. This Quantum Machine Learning library provides updated algorithms like QNN (Quantum Neural Networks) & PQM(Parameterized Quantum Circuits). TensorFlow Quantum (TFQ) can be implemented to design QNLP specific quantum circuits.

5.2 Lambeq

It is the first and the only available Quantum natural language processing library developed by Cambridge Quantum [10]. The Lambeq QNLP library is perfectly compatible with Cambridge Quantum's TKET, which is also open-source and frequently used as a software development platform.

At a high level, the library allows the conversion of any sentence to a quantum circuit, based on a given compositional model and certain parameterisation and choices of ansätze, and facilitates training for both quantum and classical NLP experiments. The process of using Lambeq is divided into four steps [10]:

Step 1: Sentence Input is to convert a given sentence into a string diagram using a given compositional models. In fact, any compositional schema that presents sentences as string diagrams or as tensor networks can be added to the toolbox. Currently, the toolkit includes a variety of compositional models that use different levels of syntactic information:

1. Bag-of-words models, do not use any syntactic information instead it renders simple monoidal structures consisting of boxes and wires. For example Spiders reader which composing the words using a spider (see Fig. 5).

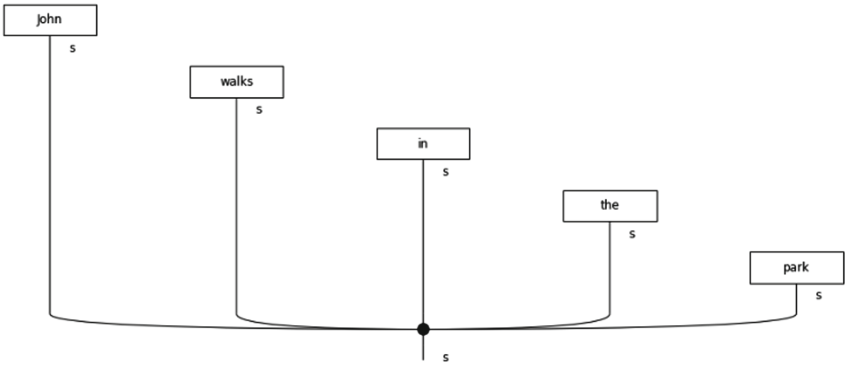

Fig. 5. Spiders reader diagram [10]

2. Word-sequence models, respect word order. There are two models, Cups reader generates a tensor train (see Fig. 6) and Stairs reader combines consecutive words using a box ("cell") in a recurrent fashion, similarly to a recurrent neural network.

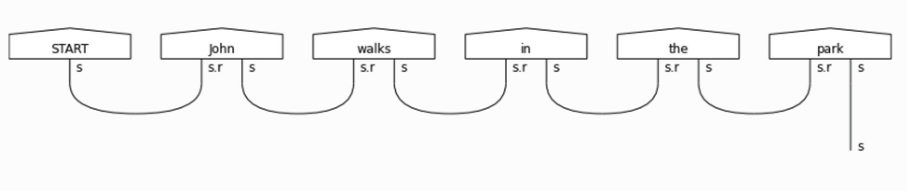

Fig. 6. Cups reader diagram [10]

3. fully syntax-based models, based on grammatical derivations given by a parser. Two cases of syntax-based models in lambeq are: BobCatParser in order to obtain a DisCoCat-like output (see Fig. 7), and tree readers which can be directly interpreted as a series of compositions without any explicit conversion into a pregroup form.

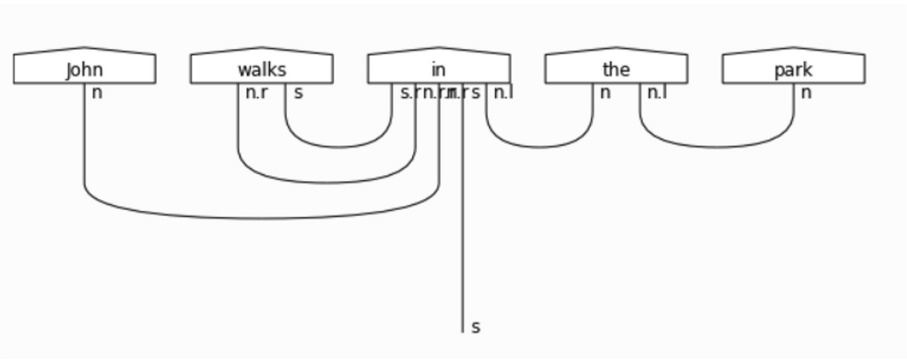

Fig. 7. BobCatParser diagram [10]

Step 2: Diagram Rewriting Syntactic derivations in pregroup form can become extremely complicated, resulting in unnecessary hardware resource usage and unacceptably long training times. Lambeq provide a predefined rewriter rules that simplify string diagrams. Example in (see Fig. 8) where eliminating completely the determiner "the".

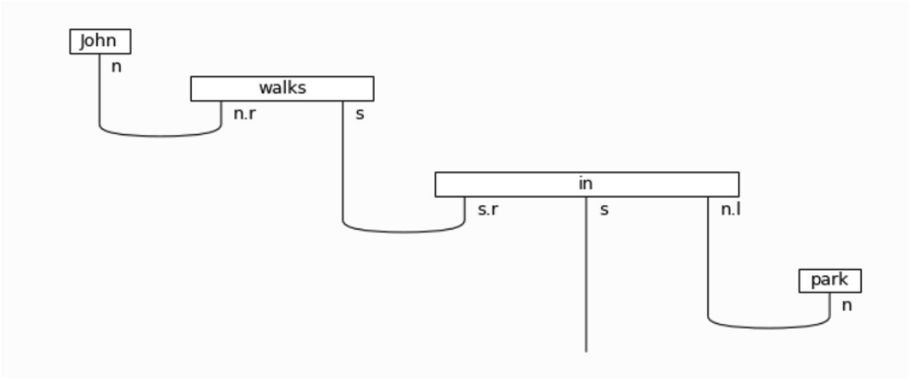

Fig. 8. BobCatParser simplified diagram [10]

This is clear that string diagrams are more flexible than tensor networks. The Toolkit supports auxiliary verbs, connectors, coordinators, adverbs, determiners, relative pronouns, and prepositional phrases as basic rewriting rules.

Step 3: Parameterisation A string diagram can be turned into a concrete quantum circuit (quantum case) or tensor network (classical case) by applying ansätze. An ansatz specifies options like the number of qubits associated with each string diagram wire and the concrete parameterised quantum states that correspond to each word (see Fig. 9).

5.3 Step 4: Training

Using the Lambeq toolkit, provides easy high-level abstractions for all essential supervised learning situations, both classical and quantum. The most significant aspect of Lambeq is that it is a high level tool i.e. we don't need to focus on low level architectural details and can therefore focus on our application.

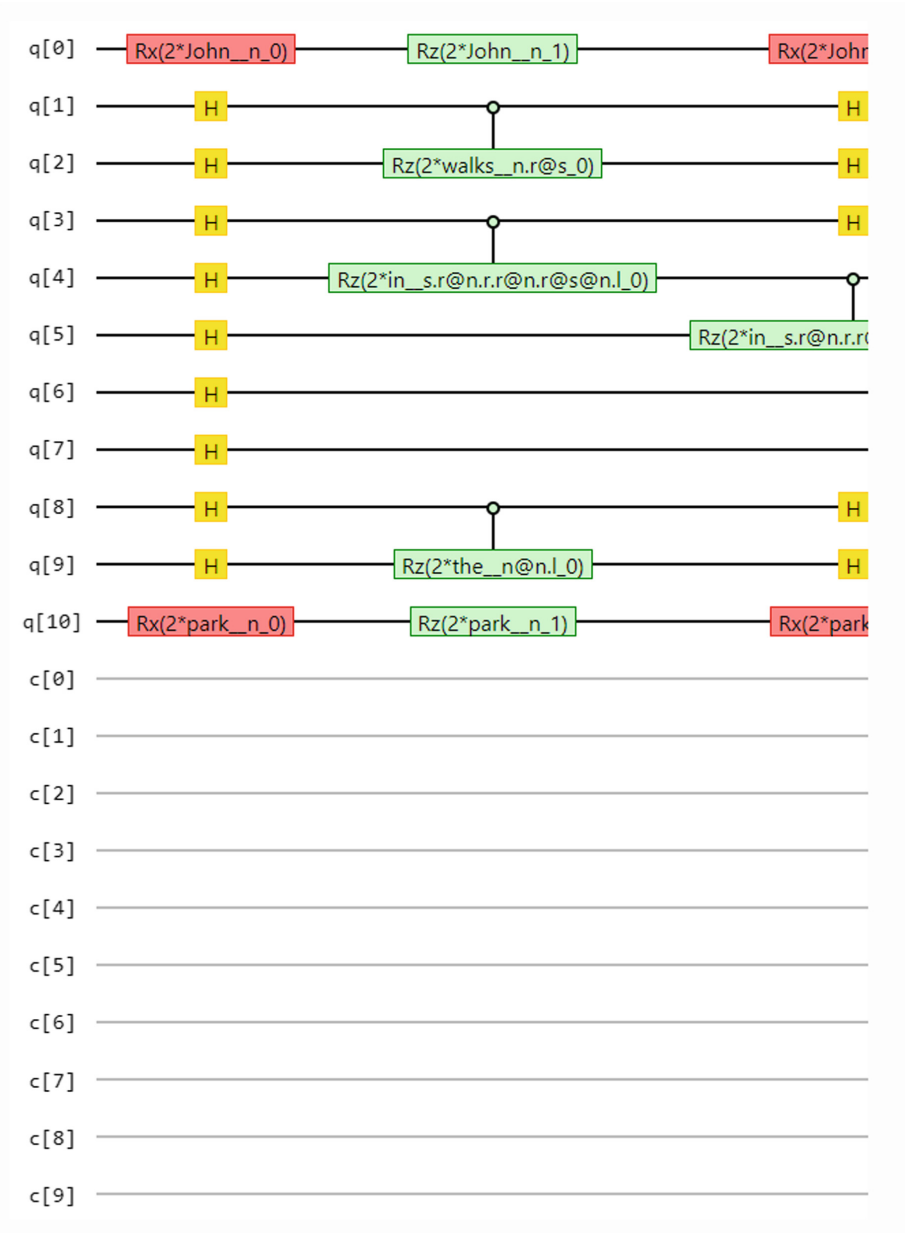

Fig. 9. IQP circuit in pytket form [10]

6 Conclusion

Quantum natural language processing is in the initial stage of research. Many theoretical and mathematical foundations were initiated in the early research studies in literature, the most important was the DisCoCat model. Some approaches have used DisCoCat structure while not utilizing the entire model. Other research, include DisCoCat model into their methodology, while others present alternate models. The experimental stage of QNLP is now realistic, with the use of NISQ devices.

Quantum algorithms [12,14,19] demonstrating promising results even experimenting on small datasets and simple situations. Currently NISQ device have 50–100 qubits therefore using a more sentences or more words vocabulary is impossible!. In terms of future development, the QNLP will be able to use large datasets and complex models with a high number of parameters. Another future goal is to explore new and more complex tasks such as Text generation, Text translation and text summarizing.

Acknowledgements. I'm extremely grateful to my supervisor Pr.Belhadef for his continuous encouragement to just do my best in this promising domain.

References

1. Abramsky, S., Coecke, B.: Categorical quantum mechanics. Handb. Quantum Logic Quantum Struct. **2**, 261–325 (2009)
2. Basile, I., Tamburini, F.: Towards quantum language models. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1840–1849 (2017)
3. Casals, A.: Medical robotics at UPC. Microprocess. Microsyst. **23**(2), 69–74 (1999)
4. Coecke, B., de Felice, G., Meichanetzidis, K., Toumi, A.: Foundations for near-term quantum natural language processing. arXiv preprint arXiv:2012.03755 (2020)
5. Coecke, B., de Felice, G., Meichanetzidis, K., Toumi, A., Gogioso, S., Chiappori, N.: Quantum natural language processing (2020)
6. Coecke, B., Sadrzadeh, M., Clark, S.: Mathematical foundations for a compositional distributional model of meaning. arXiv preprint arXiv:1003.4394 (2010)
7. FUJII, A.: Reach and limits of the supermassive model GPT-3 (2022). https://medium.com/analytics-vidhya/reach-and-limits-of-the-supermassive-model-gpt-3
8. Giovannetti, V., Lloyd, S., Maccone, L.: Quantum random access memory. Phys. Rev. Lett. **100**(16), 160501 (2008)
9. Jiang, Z., Xu, F.F., Araki, J., Neubig, G.: How can we know what language models know? Trans. Assoc. Comput. Linguist. **8**, 423–438 (2020)
10. Kartsaklis, D., et al.: LAMBEQ: an efficient high-level python library for quantum NLP. arXiv preprint arXiv:2110.04236 (2021)
11. Li, Q., Melucci, M., Tiwari, P.: Quantum language model-based query expansion. In: Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval, pp. 183–186 (2018)
12. Lorenz, R., Pearson, A., Meichanetzidis, K., Kartsaklis, D., Coecke, B.: QNLP in practice: running compositional models of meaning on a quantum computer. arXiv preprint arXiv:2102.12846 (2021)

13. Meichanetzidis, K., Gogioso, S., De Felice, G., Chiappori, N., Toumi, A., Coecke, B.: Quantum natural language processing on near-term quantum computers. arXiv preprint arXiv:2005.04147 (2020)
14. Meichanetzidis, K., Toumi, A., de Felice, G., Coecke, B.: Grammar-aware question-answering on quantum computers. arXiv preprint arXiv:2012.03756 (2020)
15. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
16. Preskill, J.: Quantum computing in the NISQ era and beyond. Quantum **2**, 79 (2018)
17. Rogers, A.: How the transformers broke NLP leaderboards (2022). https://hackingsemantics.xyz/2019/leaderboards/
18. Sordoni, A., Nie, J.Y., Bengio, Y.: Modeling term dependencies with quantum language models for IR. In: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 653–662 (2013)
19. Vicente Nieto, I.: Towards machine translation with quantum computers (2021)
20. Wang, B., Zhao, D., Lioma, C., Li, Q., Zhang, P., Simonsen, J.G.: Encoding word order in complex embeddings. arXiv preprint arXiv:1912.12333 (2019)
21. Widdows, D., Zhu, D., Zimmerman, C.: Near-term advances in quantum natural language processing. arXiv preprint arXiv:2206.02171 (2022)
22. Xu, F., Ma, X., Zhang, Q., Lo, H.K., Pan, J.W.: Secure quantum key distribution with realistic devices. Rev. Mod. Phys. **92**(2), 025002 (2020)
23. Zeng, W.; Coecke, B.: Quantum algorithms for compositional natural language processing (2016)
24. Zhang, P., Niu, J., Su, Z., Wang, B., Ma, L., Song, D.: End-to-end quantum-like language models with application to question answering. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
25. Zhang, P., Su, Z., Zhang, L., Wang, B., Song, D.: A quantum many-body wave function inspired language modeling approach. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 1303–1312 (2018)
26. Zhang, P., Zhang, J., Ma, X., Rao, S., Tian, G., Wang, J.: TextTN: probabilistic encoding of language on tensor network (2020)
27. Zhang, Y., Li, Q., Song, D., Zhang, P., Wang, P.: Quantum-inspired interactive networks for conversational sentiment analysis (2019)