Qiskit Hackathon Europe Report: Quantum Natural Language Processing

Edwin Agnew, Pablo Díez-Valle, Paula García-Molina, María Hita-Pérez, and Carlos Vega

I. MOTIVATION

Natural Language Processing (NLP) is devoted to the design of machine learning architectures capable of solving problems involving human language. Although classical NLP models are able to achieve high accuracies in multiple tasks, they tend to encode the meaning of words only through the frequency of appearance in the training text corpus (*bag of words*), thus neglecting the role of grammar [1]. It is known that, in a quantum setting, grammar can be encoded through the entanglement degrees of freedom of a quantum circuit, an idea that is formalized in the categorical distributional semantics (DisCoCat) paradigm [2, 3]. Representing and processing words and grammar in terms of quantum circuits is known as Quantum Natural Language Processing (QNLP), and takes advantage of this quantum-native nature of language. Its core idea consists in optimizing quantum variational circuits representing sentences, according to some cost function specifically designed for the language processing task of interest. Such paradigm offers grammar-aware encodings of language that may outperform classical vector-space models whenever grammar plays a key role in the considered task. In addition, the exponential scaling of the state space dimension available in a quantum computer could alleviate the memory constrains that limit classical approaches. Previously, the main contributions to this field have consisted in the theoretical development of the circuital representation of sentences [4, 5] and its application to classification algorithms for language processing. It was not until recently [6, 7] that these proposals have been implemented in Noisy Intermediate-Scale Quantum (NISQ) devices, achieving high performance in noisy computations with relatively small datasets and showing the suitability of this approach for state-of-the-art quantum computers. In this work, we follow the ideas developed therein to extend these implementations to more complex scenarios: multiclass classification applied to sentiment analysis and semantic interpretation for translation.

II. IMPLEMENTATION

A. Quantum Natural Language Processing

QNLP is rooted on the DisCoCat model of semantics [2]. In this context, sentences are conceived as a product of different word types, which are built from a set of elemental or atomic types with certain reduction rules. Given a type t, there exist a right-adjoint t^r and a left-adjoint t^l such that $t \cdot t^r = 1$ and $t^l \cdot t = 1$. Models that have already been implemented work with two atomic types: noun (n) and sentences (s). For instance, a noun as *woman* or *car* has a word type n, a transitive verb as *steals* has word type $n^r \cdot s \cdot n^l$ (where the left-adjoint type acquaints for the subject and the right-adjoint type for the object) and an adjective such as *red* has a word type $n \cdot n^l$, connecting verbs and nouns. Hence, the sentence *Woman steals red car* has type s,

$$n \cdot (n^r \cdot s \cdot n^l) \cdot (n \cdot n^l) \cdot n = 1 \cdot s \cdot 1 \cdot 1 = s. \qquad (1)$$

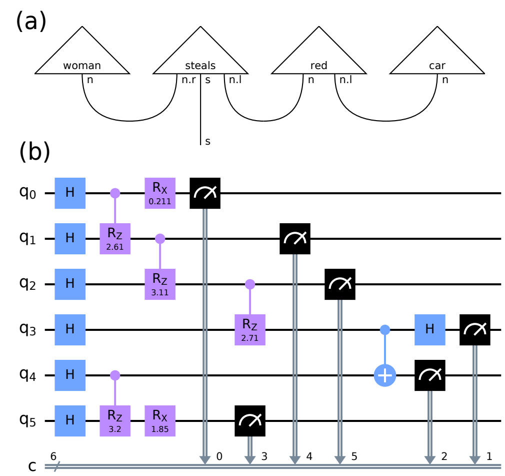

Figure 1. DisCoCat diagram (a) and Qiskit quantum variational circuit (b) for the sentence *Woman steals red car.*

The key feature that makes this construction quantum-native is the mapping from DisCoCat model to a tensor space representation (Fig. 1a). In this transformation, atomic types are mapped to vector spaces ($n \to N$, $s \to S$) and compositional word types are mapped to tensor spaces ($n^l \cdot s \cdot n^r \to N \otimes S \otimes N$), which implies that atomic types can be encoded in qubit-Hilbert spaces. Now, using $q_n (q_s)$ qubits to encode noun (sentence) types, a generic word class can be built as follows: a noun can be written as $\mathcal{O}^N(\vec{\alpha})|0^{\otimes q_n}\rangle$, a transitive verb as $\mathcal{O}^V(\vec{\beta})|0^{\otimes q_n}\rangle|0^{\otimes q_s}\rangle|0^{\otimes q_n}\rangle$ and an adjective as $\mathcal{O}^{\mathcal{A}}(\vec{\gamma})|0^{\otimes q_n}\rangle|0^{\otimes q_n}\rangle$, where $\vec{\alpha}$, $\vec{\beta}$ and $\vec{\gamma}$ are variational parameters of some ansatz operators $\mathcal{O}^N$, $\mathcal{O}^V$ and $\mathcal{O}^A$. Once the form of the ansatz is assumed, all words in the vocabulary can be written as the output of a variational operator over a set of qubits. In this project, we have worked with the IQP ansatz and single R_x and R_z gates, due to the good results already obtained in the literature [7]. Diagrams can then be written as variational quantum circuits (Fig. 1b), and optimized via a variational hybrid quantum-classical algorithm where classical optimizer (SPSA in our implementation) uses quantum measurements of the output state $|P(\Theta)\rangle$ to evaluate the cost function.

B. Our tasks

In this work, we propose two QNLP algorithms to solve two common tasks in NLP.

*a. **Sentiment analysis.*** Classification is one of the main problems studied in supervised machine learning. In Ref. [7] a QNLP binary classifier was introduced, achieving a successful implementation. Using this scheme as a baseline, we take a step forward and propose a QNLP algorithm for multi-class classification. This is of utmost importance, as most real problems are not binary. More concretely, we have focused on sentiment classification for the sentiments happy, sad, angry, and scared, using a 366-sentence dataset (e.g. *Morose woman cries.*).

First, inspired by classical multi-class classification and the binary classification in Ref. [7], we use the 1vs1 and 1vsAll approaches [8], with $q_n = q_s = 1$. Then, we propose a new multi-class classification circuit, extending the binary classification circuits to 4 labels by using 2 qubits to encode the nouns and sentences ($q_n = q_s = 2$), meaning we can distinguish between four different outcomes.The cost function evaluated for some variational parameters Θ can then be computed as the cross entropy

$$C(\Theta) = -\sum_i L_i \cdot \log\left(p_i(\Theta) - \varepsilon\right), \; i = 1, .., n_{\text{labels}} \,, \quad (2)$$

where L is the true label and p the predicted one, each component corresponding to the probability of measuring a certain quantum state for a variational circuit $W(\theta)$. We introduce a small $\varepsilon \sim 10^{-9}$ for numerical stability. We also try a train error formula as cost function, although it can lead to overfitting.

*b. **Multi-language Semantic interpretation.*** Addressing the problem of interpreting different human languages with quantum computers and infering cross-meanings between them comes with specific issues compared to a single-sentence classifier. Indeed, the learning algorithm in this context is required to capture correlations between words rather than the association of certain words with specific target classes. Moreover, since there are grammatical differences between languages, it seems natural that the DisCoCat implementation of word types should be formulated in a language-dependent manner. Following the discussion on this issue presented in Ref. [9, 10] from a category-theoretic perspective, we modified the functor carrying sentences to quantum circuits by changing the adjective word type from $n \cdot n^l$ (English) to $n^r \cdot n$ (Spanish). This construction is motivated by the fact that the noun-adjective ordering in English (e.g. as in *cute cat*) is reversed with respect to the one in Spanish (as in *gato bonito*). With this difference in mind, we build an optimization pipeline by training two independent quantum variational circuits Circ_E and Circ_S (one for each language) and building the cost function to optimize from the inner product of the two outputs as follows

$$C(\Theta_S, \Theta_E) = \sum_i \left| |\langle P^i(\Theta_S) | P^i(\Theta_E)\rangle|^2 - L_i - \varepsilon \right| \,, \quad (3)$$

where i labels each training example, $\Theta_{S,E}$ are the variational parameters of each circuits, $|P(\Theta_{S,E})\rangle$ are the output states of the circuits $\text{Circ}_{S,E}$ for an input pair of sentences with ground-truth label $L_i \in \{0, 1\}$.

C. Work distribution

In this section, we state the work distribution among the team-mates Edwin Agnew (EA), Pablo Díez-Valle (PD), María Hita-Pérez (MH), Paula García-Molina (PG), and Carlos Vega (CV) during the Hackathon.

- **First week**: Study DiscoPy documentation and its interface with Qiskit (EA, PG), learn the fundamentals of DisCoCat research on optimization approaches for our problem(PD, MH, CV).
- **Second week**: Scheme design for the code in sentiment analysis/semantic interpretation (EA, PG, PD), dataset preparation (MH, CV)
- **Third week**: Execution of optimization algorithms for sentiment analysis (EA, PG) and semantic interpretation (PD, MH, CV), global analysis and discussion of results (EA, PD, PG, MH, CV).
- **Fourth week**: Writing of the report and video script (EA, CV), video make-up (MH, PG), GitHub repository preparation (PD, PG).

III. RESULTS

A. Sentiment Analysis

The successful implementation of this task in the statevector simulator with cross entropy as cost function is a strong proof of concept of the power of this approach (Fig. 2). We have achieved a test accuracy of 81% for 2-qubit classification, 75% for 1vsAll (1vsAll does not consider the entire dataset), and 63% for 1vs1. These approaches were implemented for a different number of iterations due to time limitations and the nature of the proposals. From now on we will focus on the 1vs1 approach, as the smaller binary datasets (less quantum circuits), together with the smaller number of qubits, makes it more suitable for the implementation in a quantum computer. To study this algorithm we also use the train error as cost function, reaching an accuracy of 77% for the statevector simulator. Moreover, when we implement it using the qasm simulator, the performance improves- 79% cross entropy, 87% train error, and hence no overfitting-, due to the intrinsic randomness of the quantum computer and the SPSA optimizer. With view to the implementation on a real quantum computer, we choose the cross entropy as cost function for the ibm_16_melbourne simulator, as it converges faster. We achieve a 78% accuracy, showing the noise resilience of our approach (Fig. 2b).

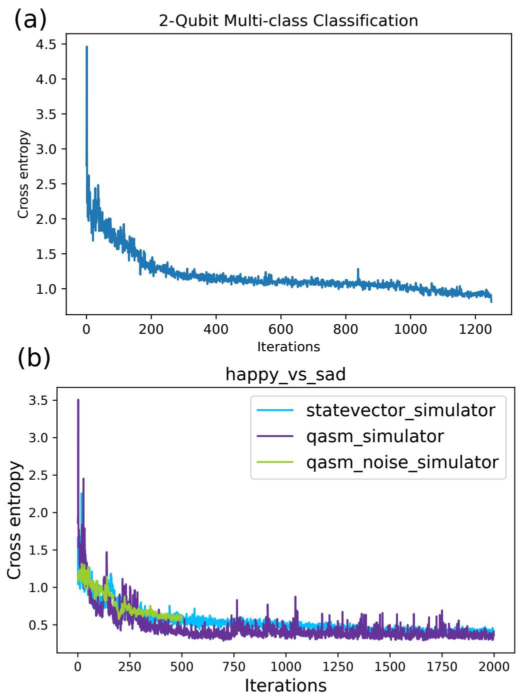

Figure 2. Train cost for sentiment analysis: (a) 2-qubit multi-classification (statevector), and (b) binary *happy vs sad* 1vs1 classifier (with 500 convergence is practically achieved).

B. Multi-language semantic interpretation

Using $q_s = 2$ and $q_n = 1$, we optimize the cost function in Eq. 3 in different contexts (Fig. 3). Firstly, in the ideal case, we used an exact calculation using the statevector simulator, finding that eventually the cost function is stuck in a plateau, overfitting in the small dataset case. Next, we used a more realistic approach using the QASM simulator subject to a finite number of shots to evaluate the cost function. Interestingly, we observe that in this approach the learning algorithm does not get stuck in a plateau, and indeed yields better results ($> 70\%$ accuracy) than the ideal case. Finally, we run the experiment emulating the ibmq_16_melbourne real quantum device, both its architecture and the noise model, seeking to include the impact of real hardware limitations. We can observe a similar level of performance as in the Statevector simulator, proving the algorithm's resilience ($\sim 75\%$ accuracy with medium dataset). In this case, the number of iterations is smaller for larger datasets due to time limitations.

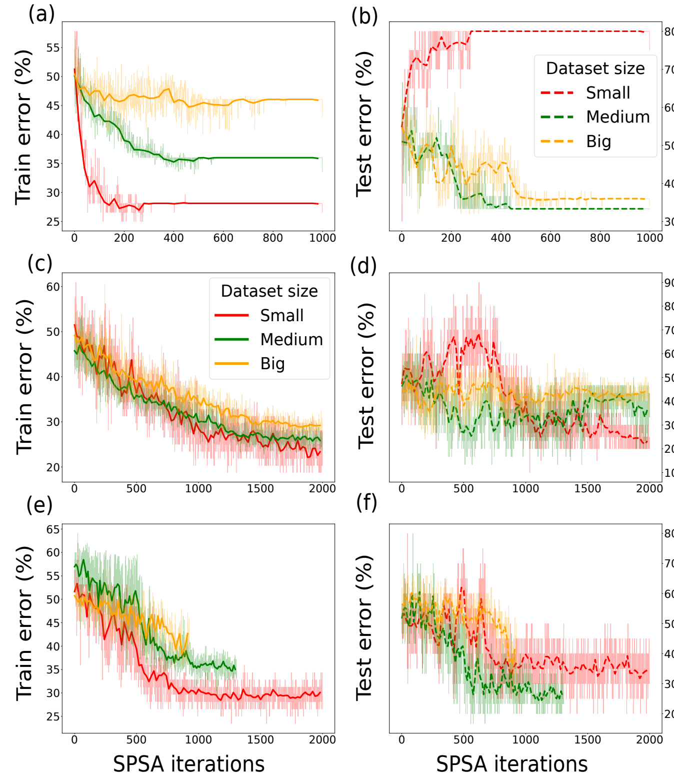

Figure 3. Train/test error in the semantic interpretation task, for (a-b) Statevector ideal simulation (c-d) QASM noiseless simulation and (e-f) real quantum device simulation (ibmq_16_melbourne). The datasets sizes are $N_E = N_S = 20$ (small), 65 (medium) and 90 (large) where N_E, N_S are the number of sentences in English/Spanish respectively.

IV. IMPACT AND OUTLOOK

We have created the first multi-class classification and multi-language semantic interpretation implementations for QNLP. Our work demonstrates the effectiveness of QNLP and provides further examples of quantum computers being able to solve useful problems. Classical NLP is known to be very computationally expensive, while our approach uses at most 10 qubits demonstrating its efficiency. Moreover, the good performance under the effect of noise makes it suitable for NISQ era devices, even for hundreds of circuits and parameters for the optimization (our approach avoids barren plateaus at an early stage of the optimization process). This together with the extension of multi-class classification to other areas offers a promising perspective for the future of QNLP and QML. Due to the devices' queues, the great number of circuit evaluations and time limitations, we could not run the codes in a real quantum computer. In the future, we will study its suitability for real quantum computers and optimize it using a variety of ansatz choices, cost functions and classical optimizers. Nevertheless, these results already constitute a step forward in the digital understanding of language.

[1] K. S. Jones, Current issues in computational linguistics: in honour of Don Walker , 3 (1994).

[2] B. Coecke, M. Sadrzadeh, and S. Clark, "Mathematical foundations for a compositional distributional model of meaning," (2010), arXiv:1003.4394 [cs.CL].

[3] B. Coecke, arXiv preprint arXiv:1904.03478 (2019).

[4] B. Coecke, G. de Felice, K. Meichanetzidis, and A. Toumi, "Foundations for near-term quantum natural language processing," (2020), arXiv:2012.03755 [quant-ph].

[5] W. Zeng and B. Coecke, arXiv preprint arXiv:1608.01406 (2016).

[6] K. Meichanetzidis, A. Toumi, G. de Felice, and B. Coecke, "Grammar-aware question-answering on quantum computers," (2020), arXiv:2012.03756 [quant-ph].

[7] R. Lorenz, A. Pearson, K. Meichanetzidis, D. Kartsaklis, and B. Coecke, "Qnlp in practice: Running compositional models of meaning on a quantum computer," (2021), arXiv:2102.12846 [cs.CL].

[8] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, Emerging artificial intelligence applications in computer engineering **160**, 3 (2007).

[9] M. Abbaszadeh, S. S. Mousavi, and V. Salari, "Parametrized quantum circuits of synonymous sentences in quantum natural language processing," (2021), arXiv:2102.02204 [quant-ph].

[10] T.-D. Bradley, M. Lewis, J. Master, and B. Theilman, Electronic Proceedings in Theoretical Computer Science **283**, 50 (2018).