A Quantum Model of Concepts

Sean Tull*, Razin A. Shaikh*†, Sara Sabrina Zemljič* and Stephen Clark*
*Quantinuum
17 Beaumont Street, Oxford, UK
†Department of Computer Science, University of Oxford
{sean.tull,razin.shaikh,sara.zemljic,steve.clark}@quantinuum.com

Abstract

In this paper we present a new modelling framework for concepts based on quantum theory, and demonstrate how the conceptual representations can be learned from data. Our approach builds upon Gärdenfors' classical framework of *conceptual spaces*, in which cognition is modelled geometrically through the use of convex spaces, which in turn factorise in terms of simpler spaces called *domains*. We show how concepts from the domains of SHAPE, COLOUR, SIZE and POSITION can be learned from images of simple shapes, where individual images are represented as quantum states and concepts as quantum effects. Concepts are learned by a hybrid classical-quantum network trained to perform concept classification. We also use discarding to produce mixed effects, which can then be used to learn concepts which only apply to a subset of the domains, and show how entanglement (together with discarding) can be used to capture interesting correlations across domains.

Keywords: conceptual spaces; quantum cognition; quantum machine learning

Introduction

The application of mathematical tools from quantum theory to the modelling of cognitive phenomena has led to an emerging area called quantum cognition (Pothos & Busemeyer, 2013). The idea is that some of the features of quantum theory, such as entanglement, can be used to account for psychological data which can be hard to model classically. Examples include ordering effects in how subjects answer questions (Trueblood & Busemeyer, 2011) and concept combination (Aerts & Gabora, 2005).

Another recent development in concept modelling has been the application of machine learning to the problem of how artificial agents can learn concepts from raw perceptual data (Higgins et al., 2017, 2018; Shaikh et al., 2022). The motivation for endowing an agent with conceptual representations is that this will enable it to reason and act more effectively, similar to how humans use concepts (Lake et al., 2017). One hope is that the explicit use of concepts will ameliorate some of the negative consequences of the "black-box" nature of neural architectures currently being used in AI.

In this paper we present a new modelling framework for concepts based on the mathematical formalism used in quantum theory (Coecke & Kissinger, 2017; Nielsen & Chuang, 2000), and demonstrate how the conceptual representations can be learned from data. We build upon the framework of *conceptual spaces* (Gärdenfors, 2004, 2014), in which cognition is modelled geometrically through the use of convex spaces, which in turn factorise in terms of simpler spaces called *domains*. We show how concepts from the domains of SHAPE, COLOUR, SIZE and POSITION can be learned from images of simple shapes, where individual images are represented as quantum states and concepts as quantum effects. The factoring of the conceptual space is represented naturally in our models through the use of the tensor product as the monoidal product. We also show how discarding—which produces mixed effects—can be used when the concept to be learned only applies to a subset of the domains, and how entanglement (together with discarding) can be used to capture interesting correlations across domains.

We choose to implement our framework using a hybrid network trained to perform concept classification, where the image processing is carried out by a convolutional neural network and the quantum representations are produced by a parameterised quantum circuit. Even though the framework can be described at an abstract level independent of any implementation, the use-case we have in mind is one in which the models are (eventually) run on a quantum computer. Here the implementation is a classical simulation of a quantum computation.[1]

What are the reasons for applying the formalism of quantum theory to the modelling of concepts? First, it provides an alternative, and interesting, mathematical structure to the convex structure of conceptual spaces. Second, this structure comes with features which are well-suited to modelling concepts, such as entanglement for capturing correlations, and partial orders for conceptual hierarchies. Third, the use of the tensor product for combining domains leads to machine learning models with different characteristics to the neural networks typically employed in concept learning (Havlicek et al., 2019; Schuld & Killoran, 2019), which may lead to advantages in the future, especially with the development of larger, fault-tolerant quantum computers.

Quantum Models

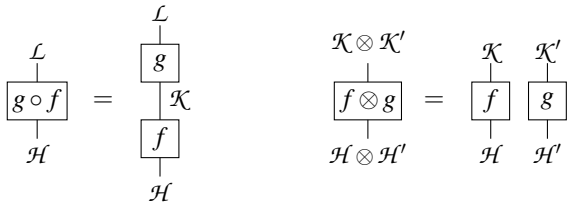
We provide a formalisation of our quantum model of concepts, giving definitions in terms of *category theory* (Fong

[1]Note that we are not making any claims in this paper of "quantum advantage" (Preskill, 2012). However, we do anticipate quantum models of concepts which require quantum hardware for their efficient use, especially as we scale to larger quantum circuits and datasets.

In M. Goldwater, F. K. Anggoro, B. K. Hayes, & D. C. Ong (Eds.), *Proceedings of the 45th Annual Conference of the Cognitive Science Society.*

& Spivak, 2019) and in particular *string diagrams* which describe morphisms in a symmetric monoidal category (Coecke & Kissinger, 2017; Selinger, 2010).[2]

We work in the category **Quant** of finite dimensional Hilbert spaces $\mathcal{H}, \mathcal{K} \ldots$ and completely positive maps. A finite dimensional Hilbert space $\mathcal{H}$ is depicted as a wire labelled by $\mathcal{H}$. A quantum process $\mathcal{H} \to \mathcal{K}$ is given by a completely positive map $f\colon L(\mathcal{H}) \to L(\mathcal{K})$ between the spaces of operators $L(\mathcal{H})$ and $L(\mathcal{K})$, respectively. We depict such a process f with a box read from bottom to top. The sequential composition $g \circ f$ of processes and tensor product $f \otimes g$ are depicted as below:

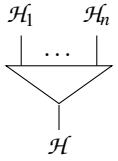


Important special cases are processes with no input (or formally with input $I = \mathbb{C}$), called *states*, which correspond to unnormalised density matrices ρ of $\mathcal{H}$. Similarly morphisms with no output are called *effects*, and correspond to positive operators $e \in L(\mathcal{H})$.

Composing a state with an effect yields a scalar $e \circ \rho \in \mathbb{R}+$. In particular the *discarding* effect $\clubsuit$ corresponds to the identity operator on $\mathcal{H}$. A *channel* is a process f which preserves discarding, so that $\clubsuit \circ f = \clubsuit$, or equivalently is trace-preserving as a CP map. In particular a state ρ is *normalised* when $\clubsuit \circ \rho = 1$, corresponding to a (normalised) density matrix (with trace 1).

Making use of our categorical formulation of conceptual spaces now gives us our notion of a quantum model.

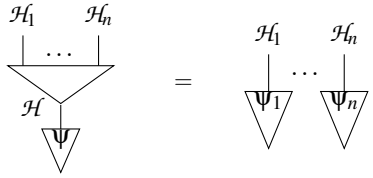
Definition 1. A *quantum conceptual model* is given by a Hilbert space $\mathcal{H}$ along with a list of Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$, called the *factors* of the space, and an embedding $\mathcal{H}$ as a subspace of $\mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_n$. We depict the embedding as follows.



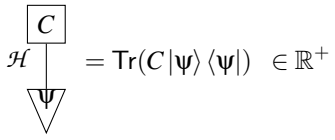
A *concept* of the model is an effect C on $\mathcal{H}$:

Thus concepts correspond to positive operators $C \in L(\mathcal{H})$. Concepts come with a natural partial order, which is given by $C \leq D \iff D - C$ is positive. This allows us to model conceptual hierarchies, for example.

We define a *crisp concept* to be a concept corresponding to a projection operator P onto a subspace $\mathcal{H}' \subseteq \mathcal{H}$. In particular any pure quantum effect $|\psi\rangle$ gives a crisp concept, projecting on to the ray spanned by ψ. Finally, an *instance* (following the terminology in Clark et al. (2021)) is given by a pure state of the form $\psi = \psi_i \otimes \cdots \otimes \psi_n$ for normalised pure states $\psi_i \in \mathcal{H}_i$, $i = 1, \ldots, n$.



Composing a concept C with an instance yields a scalar which determines how well the instance fits the concept:

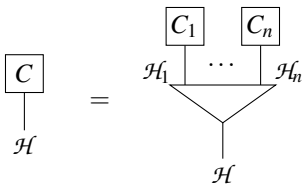


Our framework is inspired by that of conceptual spaces (Gärdenfors, 2004), in which cognitive spaces are described by convex spaces which further decompose into factors called *domains* and *dimensions*. Here the factors $\mathcal{H}_i$ play the same role, with the product of convex spaces replaced by the tensor product of Hilbert spaces. Fuzzy concepts on a conceptual space can also be modelled as effects in an appropriate category (Tull, 2021). Nonetheless both classes of models are distinct.

For example, on a qubit $\mathcal{H} = \mathbb{C}^2$, instances correspond to the Bloch sphere, which one could view as a convex set and hence a conceptual space. However, the concepts in our quantum model differ from those on a conceptual space; in particular there are no quantum effects which measure an arbitrary convex subset of the Bloch sphere, and conversely general quantum effects do not satisfy the critertion of *quasi-concavity* satisfied by fuzzy concepts on a conceptual space (Tull, 2021).

Despite these differences, we claim that quantum models share the benefits of conceptual spaces, with convex subsets replaced by linear subspaces, including the factorisation in terms of domains, and the hierarchy (partial order) on concepts. Additionally, they come with certain benefits including the presence of *entanglement*, which allows the representation of rich concepts which relate domains efficiently in a quantum model.

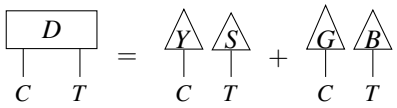
Definition 2. A *product concept* is one of the form



[2] A more detailed theoretical treatment can be found in a longer companion report: https://arxiv.org/abs/2302.14822.

for effects $C_1, \ldots, C_n$ on the factors $\mathcal{H}_1, \ldots, \mathcal{H}_n$. An *entangled concept* is a concept C whose operator cannot be written as a (weighted) sum of product concepts.

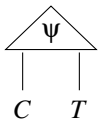
Entanglement allows us to describe concepts which capture correlations between domains beyond those which can be described classically. For example, consider a model with two factors COLOUR and TASTE, denoted C, T, and consider the learning of the concept *banana* from two instances, one which is yellow and sweet $|Y\rangle|S\rangle$ and one which is green and bitter $|G\rangle|B\rangle$, and assume $|Y\rangle$ and $|G\rangle$ are orthogonal. We can combine the instances classically by forming a sum of the two instances at the level of operators.



Alternatively we can relate the instances via entanglement by forming a new pure state via superposition

$$\psi = |Y\rangle|S\rangle + |G\rangle|B\rangle$$

and considering the pure concept



with corresponding operator $|\psi\rangle\langle\psi|$. Both concepts send the exemplars $|Y\rangle|S\rangle$, $|G\rangle|B\rangle$ to 1. However, the former (classical) combination D simply compares any given instance to the two given exemplars, while the latter (entangled) concept can be seen to encode a structural relationship between the domains COLOUR and TASTE.

In particular if we consider an instance ϕ given by a unit vector $\phi = (\alpha|Y\rangle + \beta|G\rangle) \otimes (\alpha|S\rangle + \beta|B\rangle)$, with $\alpha^2 + \beta^2 = 1$, which lies 'in-between' the exemplars, the entangled concept $\psi^\dagger$ gives value 1 while the classically correlated concept D gives $\alpha^4 + \beta^4 < 1$ in general. In this way entangled concepts can encode relationships between domains, rather than simply (weighted) collections of exemplars.

Data, Networks and Circuits

In this section we describe how the quantum concept models described above can be implemented. The key idea is to use a probabilistic classifier to implement a concept as an effect, where the (binary) classifier learns to distinguish between positive and negative examples of the relevant concept. For the generation of training and test data, we use the Spriteworld software (Watters et al., 2019) to generate simple images. These consist of coloured shapes of particular sizes in particular positions in a 2D box, against a black background. There are three shapes: {*square, triangle, circle*}; three colours: {*red, green, blue*}; three sizes: {*small, medium, large*}; and three (vertical) positions: {*bottom, centre, top*} (see Fig. 1). We ran the sampler to generate a training set of 3,000 instances, and development and test sets with 300 instances each.

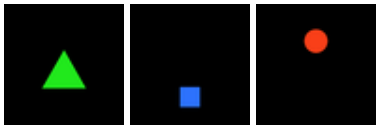


Figure 1: Example shapes: (*green, large, triangle, centre*); (*blue, small, square, bottom*); (*red, medium, circle, top*).

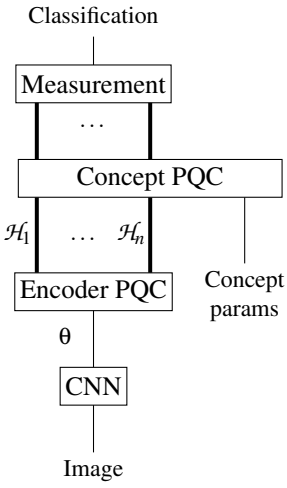
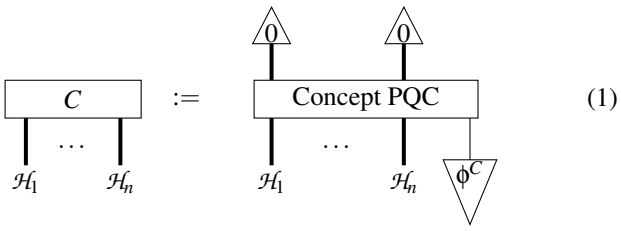


Figure 2: The hybrid classical-quantum network.

An input image is first processed by a convolutional neural network (CNN) (Goodfellow et al., 2016, Ch.9) which outputs classical parameters which are fed into a parameterised quantum circuit (PQC) (Benedetti et al., 2019). This PQC we call the *encoder PQC*; it implements a quantum state z which is the representation of the image in our model. Given a concept C, a separate *concept PQC* implements a quantum effect corresponding to C which can be applied to the instance z, as described above. We assume that the domains/factors $\mathcal{H}_1, \ldots, \mathcal{H}_n$ are known by the model; in our experiments these will be the four domains SHAPE, COLOUR, SIZE, POSITION.[3] The overall setup is shown in Figure 2, with thin wires denoting classical data and each thick wire denoting a Hilbert space given by some number of qubits.

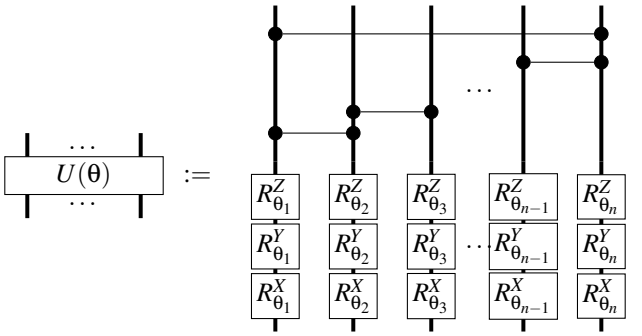
Each instance is a pure quantum state of $\mathcal{H}_1, \ldots, \mathcal{H}_n$ given by passing a particular image into the CNN and passing the output as parameters into the encoder PQC. Each specific concept C is given by running the concept PQC on some learned parameters ϕ^C and then performing a binary Pauli-Z measurement on each qubit. The value of the concept is given by the probability that all measurements return 0:

[3]The question of whether, and how, the domains could be learned automatically in the classical setting is an ongoing debate (Higgins et al., 2017; Locatello et al., 2019).



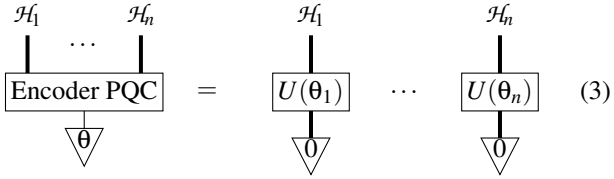
(1)

The CNN consists of 4 convolutional layers followed by a fully-connected layer, with the ReLU activation function used throughout. The PQCs make use of the circuit ansatz $U(\theta)$ in (2), containing entangling controlled Z gates, and where R^X_θ, R^Y_θ, R^Z_θ denote (potentially different) X,Y,Z rotations. We define another ansatz $V(\theta)$ in the same way but with rotations in the reverse order Z,Y,X. Sufficient layers of either ansatz will implement any unitary, and hence any instance or concept.



(2)

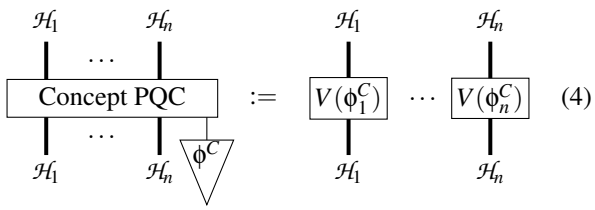
Now let us describe the encoder and concept PQCs in more detail. Both consist of some number of qubits per domain $\mathcal{H}_i$. The form of the encoder PQC is as follows:



(3)

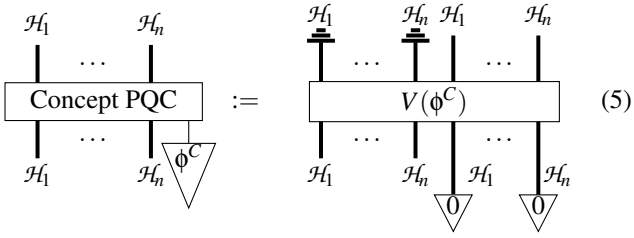
More generally we can compose multiple layers of such U circuits on each domain. Here the $|0\rangle$ states denote product states $|0\ldots0\rangle$ on each $\mathcal{H}_i$. Thus by construction the encoder never involves entanglement across factors, and can be viewed as a single encoder per factor.

In the initial experiments, we only have one qubit per domain $\mathcal{H}_i$, and use one layer in the encoder. In this case the encoder simply carries an X,Y and Z rotation per qubit, involving no entanglement. In this basic setup, the concept PQC also involves no entanglement, taking the following form.



(4)

In the extended experiments below, we use a richer form for the concept PQC, allowing us to capture entangled and mixed concepts. Here the full ansatz $V(\theta)$ is used over all domains, with an ancillary copy of each domain $\mathcal{H}_1,\ldots\mathcal{H}_n$, prepared in initial state $|0\rangle$, and discarding used to introduce mixing:



(5)

More generally one can include multiple layers of the form $V(\theta)$ prior to discarding.

Experiments

We train the quantum model to perform binary classification, with the loss function being the standard binary cross entropy (BCE) loss. The full set of parameters to be learned is $\psi \cup \phi$, where ψ is the set of parameters in the classical encoder CNN and ϕ is the set of PQC parameters associated with the set of 12 basic concepts. The training data contains 3,000 positive examples (as described earlier) and 3,000 negative examples. Each negative example is created from a positive one by randomly sampling an incorrect concept for each domain.

We trained a model using the circuit shown in (4) above, and tested it on the 300 examples in the development set. At test time we choose the concept for each domain which has the highest probability of applying to the input image. The implementation was in Tensorflow Quantum (Abadi et al., 2015), and the whole hybrid network—both the quantum and the classical parts—were trained end-to-end in simulation on a GPU. The training was run for 100 epochs (unless stated otherwise), with a batch size of 64, and the Adam optimizer was used. The classification model performed with almost perfect accuracy, obtaining 100% on the COLOUR and SHAPE domains, and 99% and 97% on the POSITION and SIZE domains, respectively. This accuracy carried over to the 300 examples in the test set, obtaining 100% on the COLOUR and SHAPE domains, and 96% and 97% on the POSITION and SIZE domains, respectively.

Fig. 3 visualises the pure effects for each of the 3 concepts on the 4 domains, by plotting the corresponding pure states on a Bloch sphere. The clusters of dots around each concept are the corresponding instances (pure states) in the training data. Note how the 3 concepts on each domain are being pushed apart (strikingly so in the case of the POSITION domain) and how the concepts sit neatly in the centre of each cluster of instances.

Adding a Decoder Loss One notable feature of the visualisations in Fig. 3 is how "tight" the instance clusters are.

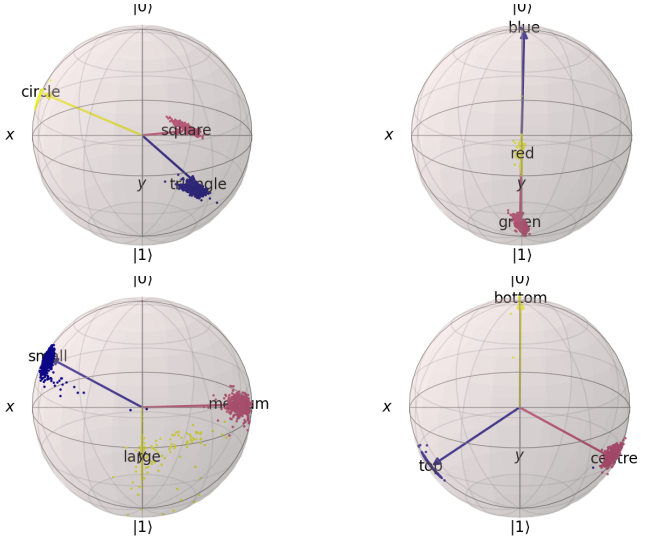


Figure 3: Visualisation of the pure concept effects and instance states on the Bloch sphere, for SHAPE, COLOUR, POSITION and SIZE (clockwise from top-left).

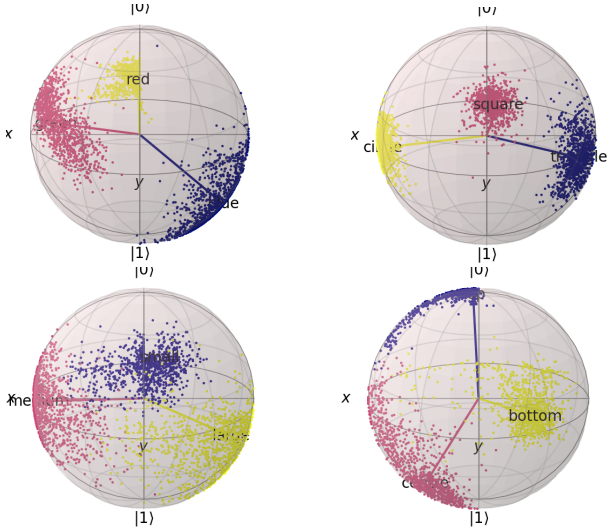


Figure 4: Visualisation of the concept effects and instance states with an additional decoder loss.

There may be use-cases where we would like the representation of instances to better reflect the variation in the underlying images, for example in order to better capture correlations across domains. In order to provide more of a "spread" of the instances, we experimented with an additional decoder loss in the loss function:

$$\text{Loss}(D,\psi,\phi,\chi) = \text{BCE}(D,\psi,\phi) + \frac{\lambda}{N}\sum_i \text{SE}(\text{DeCNN}(\chi,\text{CNN}(\psi,X_i)),X_i) \quad (6)$$

The decoder is a deconvolutional neural network (DeCNN), with parameters χ, which essentially is the CNN "in reverse": it takes as input the angles output by the CNN, given an image X_i, and outputs RGB values for each pixel in the image. SE is the sum of squared errors across all RGB values in the image, and λ is a weighting term in the overall loss. The intuition is that, in order to obtain a low SE loss, the encoder CNN has to output angles which are sufficiently informative for the DeCNN to accurately reconstruct the original image.

Figure 4 shows how the instances can be distributed more broadly around the Bloch sphere, using the additional decoder loss (with $\lambda = 0.1$). This model still performs well as a classification model on the development data, achieving over 98% accuracy on all domains.

Capturing Correlations Here we show how entanglement can be used to capture correlations across domains. We define a new concept called *twike*, which is defined as *(red and circle) or (blue and square)* (i.e. it applies to images containing red circles or blue squares). Figure 5 shows some examples of twikes and non-twikes.

The basic concept PQCs in (4) are unable to learn *twike*,

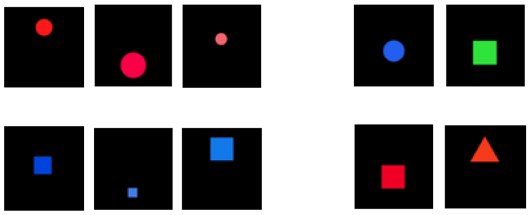


Figure 5: Example twikes (left) and non-twikes (right).

since the domains are treated independently. In order to create connections between the domains, we can apply our full ansatz V in (5). First we assume knowledge of the fact that, for *twike*, the correlations are across SHAPE and COLOUR, with entangling gates only between the qubits for these domains. We also assume that the remaining domains are not relevant and so are not measured. The resulting form of *twike* is shown in Fig. 6.

The training of this model only updates the parameters of the concept PQC; the parameters of the encoder (i.e. the CNN) are kept fixed from the training of the basic model. The loss function is binary cross entropy, with the 3,000 examples from before used as training data. Roughly 20% of these instances are positive examples, and the remaining negative examples. We trained this model for 50 epochs, using 2 layers of the rotation and entangling V ansatz for the concept PQC, and obtained 100% accuracy on the unseen test examples. It was only through the introduction of the entangling gates that we were able to learn the *twike* concept at all.

Learning General Mixed Concepts One assumption made in the *twike* experiments was that the relevant domains are known in advance. One question is whether the concept PQC could learn which domains are relevant, as well as which of

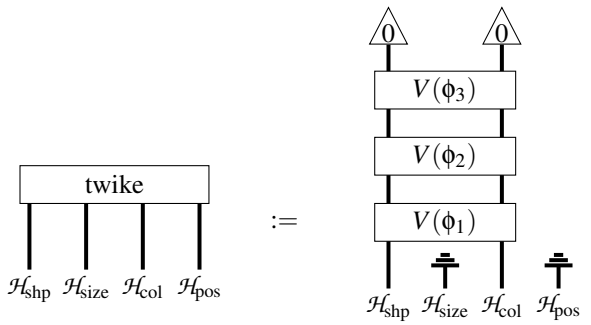


Figure 6: Encoder PQC for learning *twike*, here shown with 3 layers of the rotation and entangling V ansatz.

those domains should be correlated, if provided with all 4 wires as input. The concept PQC needs to allow for entanglement between any of its domains, and, to ignore a particular domain, the concept should effectively discard it, producing a mixed quantum effect. Both of these features can be included by using our most general form of concept PQC (5).

We set up a similar experiment to *twike*, but with just *red* as the concept to be learned. Of course the encoder had already learned *red* when trained to perform classification as part of the basic setup, but now we remove the knowledge of which wire the COLOUR domain lives on, and see whether a new concept PQC can learn *red*, given red and non-red instances as input. Again the training of this model only updates the parameters of the concept PQC; the parameters of the CNN are kept fixed. The loss function is again binary cross entropy, with the 3,000 examples used as training data. Roughly 33% of these instances are positive examples of *red*, with the remaining being negative examples. We trained this model for 50 epochs, using 2 layers of rotation and entangling gates for the concept PQC, and obtained 100% accuracy on the unseen test examples. It was only through the introduction of the discarding (plus entangling gates) that we were able to obtain these high accuracies.

Concepts containing Logical Operators Finally, we investigated whether the entangling and discarding PQC in (5) could learn concepts built from logical operators. The first concept we consider is *red and circle*, firstly with the knowledge of which domains are relevant. The encoder PQC is the simple one in (4), but with only the COLOUR and SHAPE wires. We used the same 3,000 training examples as previously, of which roughly 17% are positive and 83% negative examples. In this case the learning is particularly easy, and the model obtains 100% accuracy with only a single layer of rotations, without any entangling gates or discarding. The reason is that the factorisation of the domains through the tensor product has effectively provided all the structure required to use conjunction. When the knowledge of which domains are relevant is removed, and the more general encoder PQC in (5) is used, learning becomes harder but an encoder PQC with 4 layers of rotation and entangling gates is able to learn the concept with 100% accuracy.

Next we consider disjunction, but within rather than across domains, with the concept to be learned being *red or blue*. Of the 3,000 training examples, 61% are positive examples and 39% negative. Again, when knowledge of which domains are relevant is provided to the concept PQC, the learning is easy, with 100% accuracy obtained with a single layer of rotations.[4] When knowledge of which domains are relevant is not provided to the PQC, *red or blue* can also be successfully learned with the more general PQC in (5) with 3 layers of rotation and entangling gates, including discarding.

Future Work

One avenue for future work is to apply the quantum concepts model to data from a conceptual hierarchy—e.g. having shades of colour such as dark-red—making use of the natural ordering on effects. In addition, concepts in **Quant** come with a *negation* operation $C^{\perp} := \blacktriangle - C$, which has been studied in natural language (e.g. Rodatz et al. (2021)). In contrast, negation is harder to define for concepts in conceptual spaces; for example the complement of a convex region is generally non-convex.

Finally, even though all the practical work here has been carried out in simulation on a classical computer, the number of qubits is relatively small, and the circuits relatively shallow, and so the running of these models on real quantum hardware is a distinct possibility. Also left for future work is the search for tasks which could demonstrate advantages for our quantum representations, for example establishing whether non-separable effects in the theory do provide an advantage over classical correlation in modelling conceptual structure.

Acknowledgements

Thanks to Lia Yeh, Robin Lorenz and Douglas Brown for helpful comments, and also to the rest of the Oxford Quantum Compositional Intelligence team, and thanks to the reviewers for their thoughtful reviews and helpful comments.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems.* Retrieved from `https://www.tensorflow.org/` (Software available from tensorflow.org)

Aerts, D., & Gabora, L. (2005). A state-context-property model of concepts and their combinations I: the structure of the sets of contexts and properties. *Kybernetes*, *34*, 151-175.

[4] If each point on the Bloch sphere were to correspond to an instance of the COLOUR domain, then the PQC learning such a pure effect for *red or blue* will in fact be simply learning a single colour, intuitively somewhere "in between" *red* and *blue*. Mixed effects can learn a more general concept.

Benedetti, M., Lloyd, E., Sack, S., & Fiorentini, M. (2019). Parameterized quantum circuits as machine learning models. *Quantum Sci. Technol.*, *4*(043001).

Clark, S., Lerchner, A., von Glehn, T., Tieleman, O., Tanburn, R., Dashevskiy, M., & Bosnjak, M. (2021). *Formalising concepts as grounded abstractions* (Tech. Rep.). London, UK: DeepMind. (https://arxiv.org/abs/2101.05125)

Coecke, B., & Kissinger, A. (2017). *Picturing quantum processes: A first course in quantum theory and diagrammatic reasoning*. Cambridge University Press. doi: 10.1017/9781316219317

Fong, B., & Spivak, D. I. (2019). *An invitation to applied category theory: seven sketches in compositionality*. Cambridge University Press.

Gärdenfors, P. (2004). *Conceptual spaces: The geometry of thought*. MIT press.

Gärdenfors, P. (2014). *The geometry of meaning: Semantics based on conceptual spaces*. MIT press.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. The MIT Press.

Havlicek, V., Corcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M., & Gambetta, J. M. (2019). Supervised learning with quantum-enhanced feature spaces. *Nature*, *567*, 209-212.

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M., ... Lerchner, A. (2017). β-VAE: Learning basic visual concepts with a constrained variational framework. In *Proceedings of ICLR 2017*.

Higgins, I., Sonnerat, N., Matthey, L., Pal, A., Burgess, C. P., Bošnjak, M., ... Lerchner, A. (2018). SCAN: Learning hierarchical compositional visual concepts. In *Proceedings of ICLR 2018*.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, *40*.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., & Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th international conference on machine learning*. Long Beach, California.

Nielsen, M. A., & Chuang, I. L. (2000). *Quantum computation and quantum information*. Cambridge University Press.

Pothos, E. M., & Busemeyer, J. R. (2013). Can quantum probability provide a new direction for cognitive modeling? *Behavioral and Brain Sciences*, *36*(3).

Preskill, J. (2012). *Quantum computing and the entanglement frontier*. arXiv:1203.5813. (Rapporteur talk at the 25th Solvay Conference on Physics - The Theory of the Quantum World)

Rodatz, B., Shaikh, R. A., & Yeh, L. (2021). Conversational negation using worldly context in compositional distributional semantics. *arXiv preprint arXiv:2105.05748*.

Schuld, M., & Killoran, N. (2019, Feb). Quantum machine learning in feature hilbert spaces. *Phys. Rev. Lett.*, *122*, 040504. doi: 10.1103/PhysRevLett.122.040504

Selinger, P. (2010). A survey of graphical languages for monoidal categories. In *New structures for physics* (pp. 289–355). Springer.

Shaikh, R. A., Zemljič, S. S., Tull, S., & Clark, S. (2022). *The conceptual VAE* (Tech. Rep.). Oxford, UK: Cambridge Quantum / Quantinuum. (https://arxiv.org/abs/2203.11216)

Trueblood, J. S., & Busemeyer, J. R. (2011). A quantum probability account of order effects in inference. *Cognitive Science*, *35*, 1518–1552.

Tull, S. (2021). A categorical semantics of fuzzy concepts in conceptual spaces. *Proceedings of Applied Category Theory 2021*.

Watters, N., Matthey, L., Borgeaud, S., Kabra, R., & Lerchner, A. (2019). *Spriteworld: A flexible, configurable reinforcement learning environment*. https://github.com/deepmind/spriteworld/. Retrieved from `https://github.com/deepmind/spriteworld/`