Machine Learning Enhanced Quantum Dynamics for Excited State Dynamics

Adam O'Hare (210106369) • School of Natural and Environmental Sciences • Newcastle University
A.O'Hare2@ncl.ac.uk

1 Background

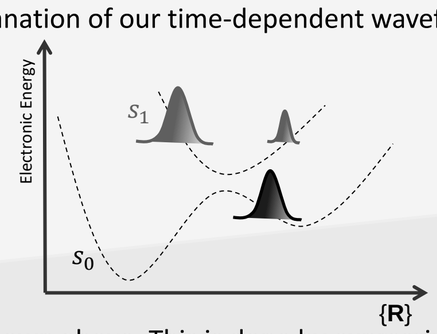
- Quantum Dynamics (QD) refers to the motions of sub-atomic matter, essentially it is the quantum version of Classical (or Newtonian) mechanics.
- QD comes from the foundations of quantum theory, which was build upon the ideas of early 20th century Nobel laureates Albert Einstein, Max Planck and Niels Bohr. Which was then developed into Quantum Mechanics, accredited to Werner Heisenberg.
- Currently, the best way to describe the QD is through solving the time-dependent Schrodinger equation (TDSE):

$$i\hbar\frac{d}{dt}\psi(t) = H(t)\psi(t)$$

- However, this is mathematically complex to solve and computationally is very time-consuming. So, is there a better way to do this?
- Machine learning (ML) has been applied to study time-evolving processes, such as stock prices and the weather, but can it be applied to quantum dynamics?
- Promising results have come from implementing artificial neural networks (ANN), a form of machine learning, into the TDSE in 1 and 2 dimensions. [1]

2 Aims & Methods

- To Build a ML model with the ability to map the evolution of a non-stationary wavefunction over time, without having to solve the TDSE directly.
- A Recurrent Neural Network (RNN) based model will be used to recognize patterns in data and predict the next likely scenario.
- By using a Trajectory-guided method, allows us to give an accurate yet compact explanation of our time-dependent wavefunction.

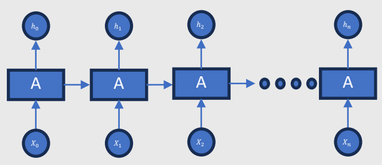


- As shown above, This is done by expressing the wavefunction as a combination of these “trajectories”, which means that we can express the overall wavefunction as a product of both position (x) and coefficients (c) against the number of trajectories. Just like the equation below:

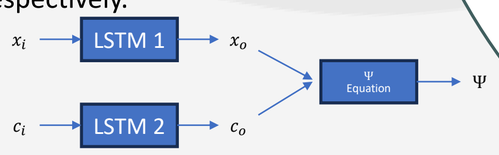
$$\Psi = \frac{(\sum x \cdot c)}{traj}$$

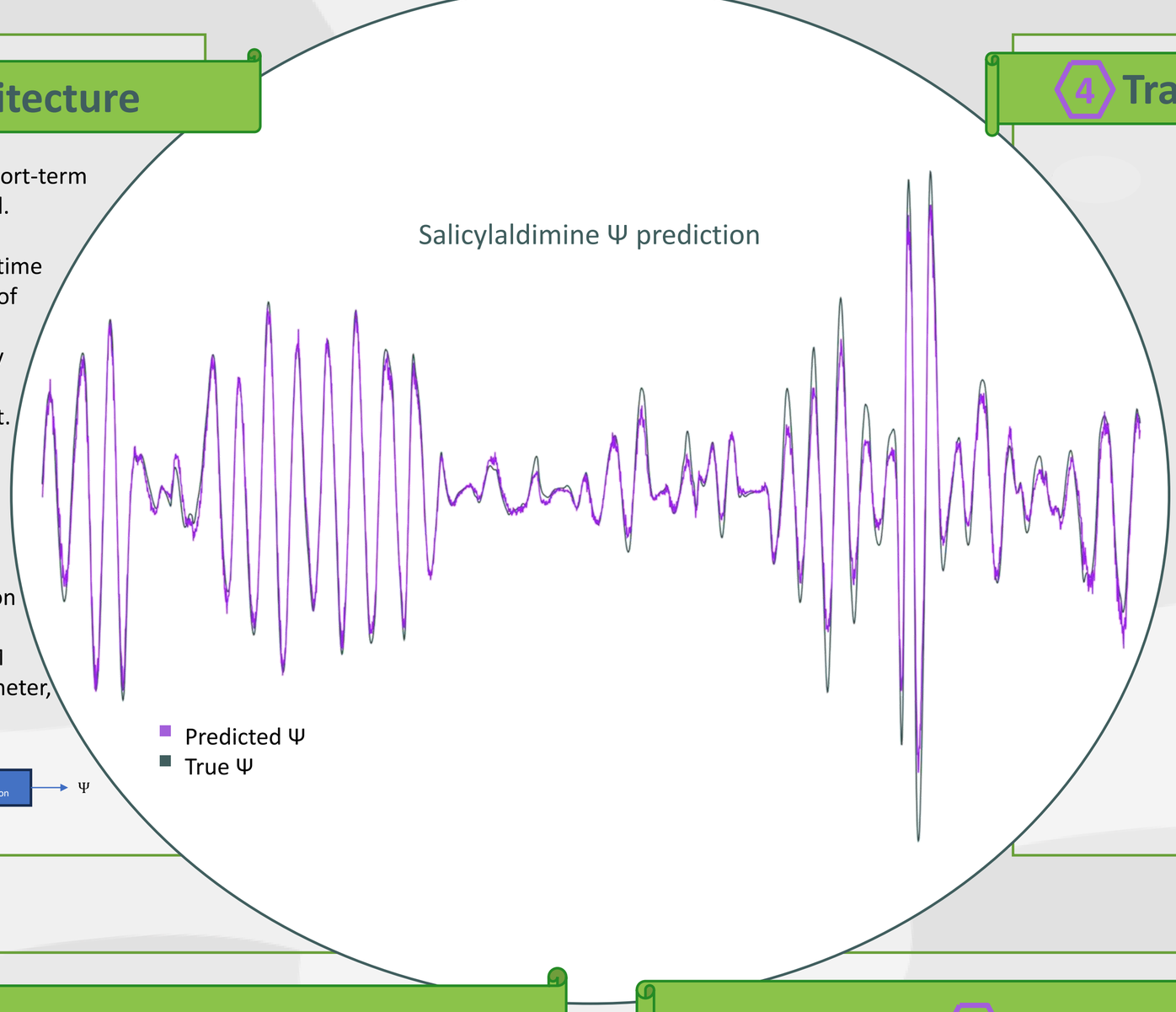
3 Model Architecture

- A type of RNN called a long short-term memory (LSTM) model is used.
- This model uses the previous time step output as the next input of the current step, allowing the LSTM cell’s long-term memory to be implemented so even more parameters can be learnt.



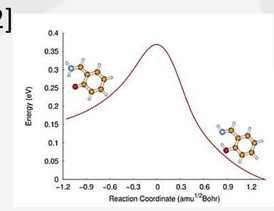
- By expressing the wavefunction as a product of position and coefficient, we have two LSTM models trained on each parameter, respectively.





4 Training the Model

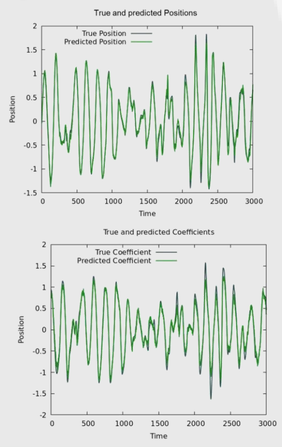
- Before testing, we trained the model via learning individual parameters (x and c separately) and then together, the test concluded that individual training produced the best results
- For testing, the model was trained on generated Hamiltonians based upon the transitional modes of salicylaldimine proton transfer[2]



- This data has 36 modes, which means that the model can be trained and then tested on different Hamiltonians

5 Results

- This image shows the predicted Vs True wavefunction of a generated Salicylaldimine proton transfer.
- This is a product of both the True Vs predicted positions and coefficients.
- The examples given are taken from the 7th Trajectory of Salicylaldimine, which is just an arbitrary value.
- These results are just some of the predictions from Simulations, but they’re accurate and exciting results.



6 Conclusion

- The project aimed to provide an alternative to solving the TDSE directly, using machine learning.
- The project is far from over, but for an 8-week Summer Project it has proved exciting results into the uses of machine learning in quantum dynamics. Particularly, its predictive effects of time-series data.

So, what’s next?

- Continue testing on more complex and a wider range of time-series data.
- Implementing an uncertainty metric, so the code can decide when it knows enough to take over.
- Currently, built as post-processing approach, therefore convert the code to be used during propagation of complex systems.

References

[1] J. Phys. Chem. Lett. 2021, 12, 10654−10662
[2] *J. Chem. Phys.* 143, 084121 (2015)

Acknowledgements

Supervisor: Prof. Tom Penfold
I would like to give a massive thank you to Tom Penfold for making this possible, and the rest of the Penfold group for their aid and contributions.
Also, thanks to Newcastle University for funding this research.