DICE ROLL ACTION GENERATIVE OPERATIONAL NETWORK (DRAGON)

Shawn Bray[1], Reece Robertson[1], Josh li[1],
University of Maryland, Baltimore County[1]

DRAGON generates action effect descriptions in response to player input. Model output is evaluated using a quantum cost function.

INTRODUCTION

DRAGON Architecture System

- Traditional approach: bag-of-words models, and grammar rules, pattern matching.
- Our approach: leverage quantum computing to handle large scale datasets. Optimize massive parameter spaces, using exponential parallelism.

This Work

- Aims to target the computational efficiency of LLM IF inference on GPU's.
- An improved exponentially faster factorization and decomposition.
- An enhanced semantic representation.

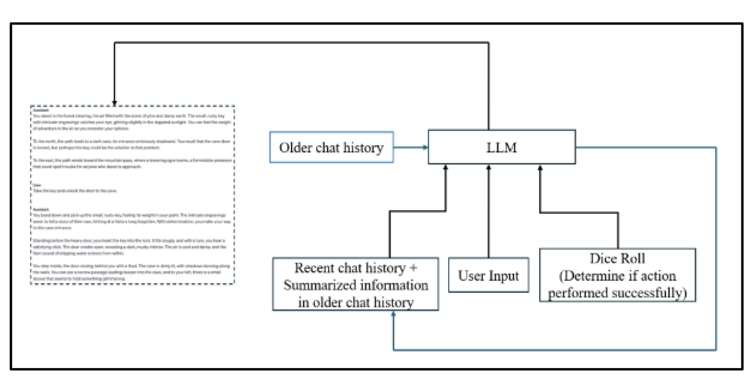


Figure 1: Dragon Architecture D

METHODOLOGY

The design adopts the LLM model.
1. Selected a LLAMA-2 model fine-tuned on D&D text data
2. Prompted the model with an action description and a die result
3. Evaluated the model output using the following metrics:
    1. BLEU score
    2. ROGUE score
    3. λambeq score*

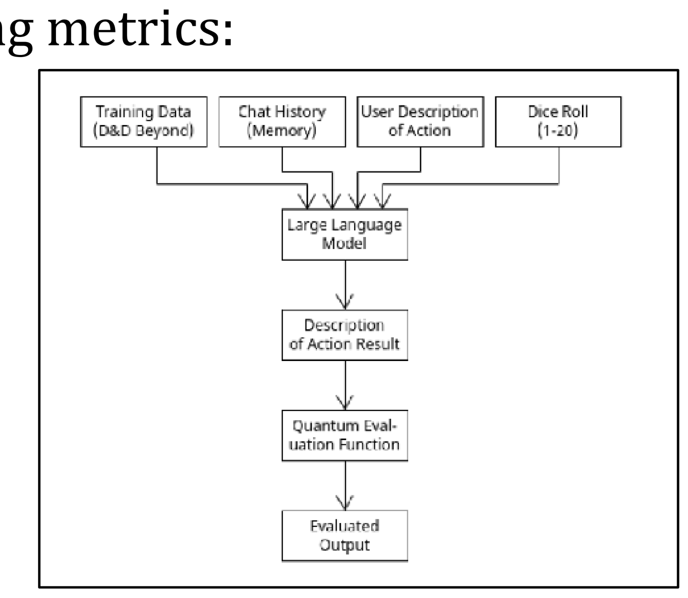
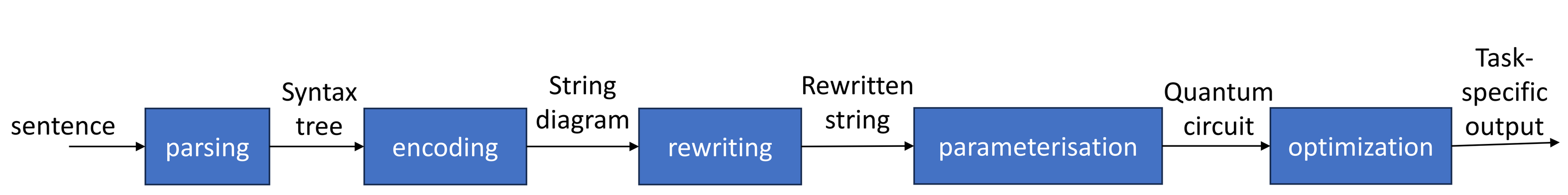


Figure 2: Flow Diagram

ALGORITHM

The λambeq
1. Encoded NLP circuit into DisCoCat.



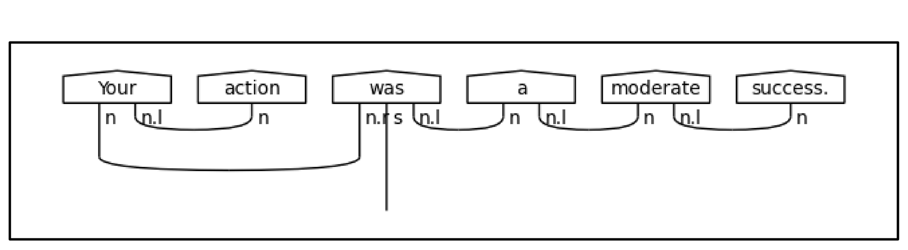


Figure 3: Target

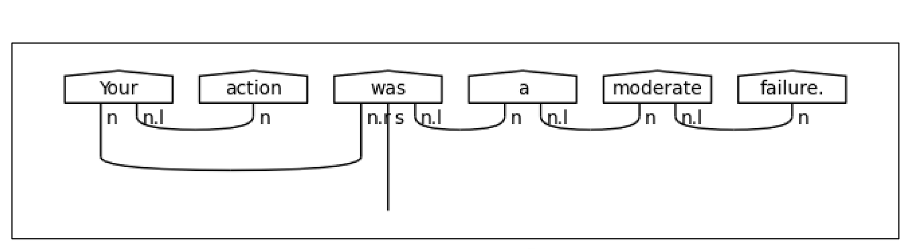


Figure 4: Hypothesis

Accuracy Analysis

- Circuit representation quantum neural network with parameterized gates.
- Qubit Hadamard gates, rotation gates and controlled opertation highly entangled.

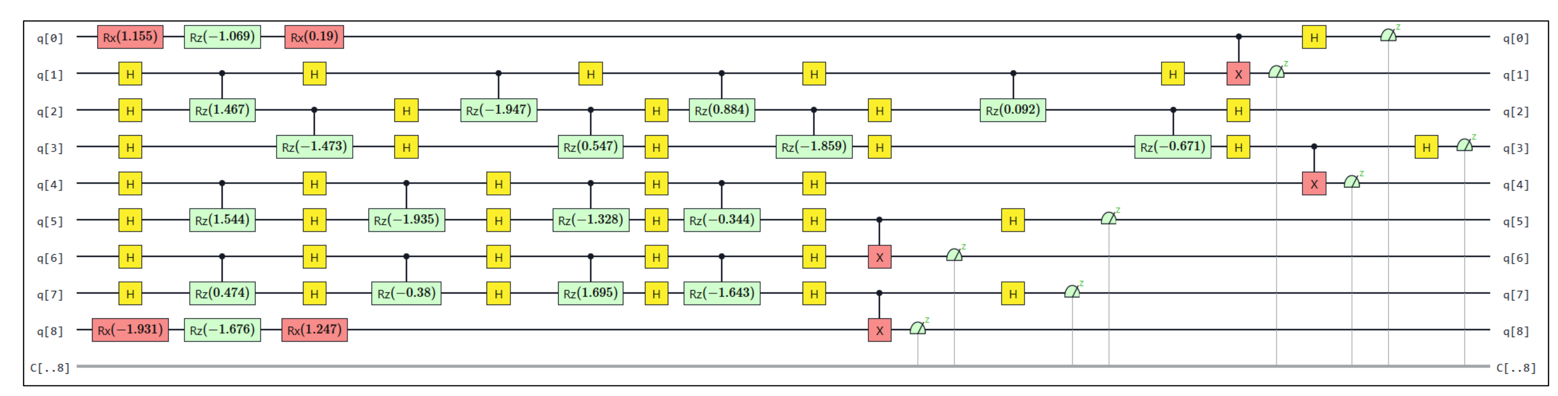


Figure 5: Circuit representation DisCoCat:

EVALUATION/RESULTS

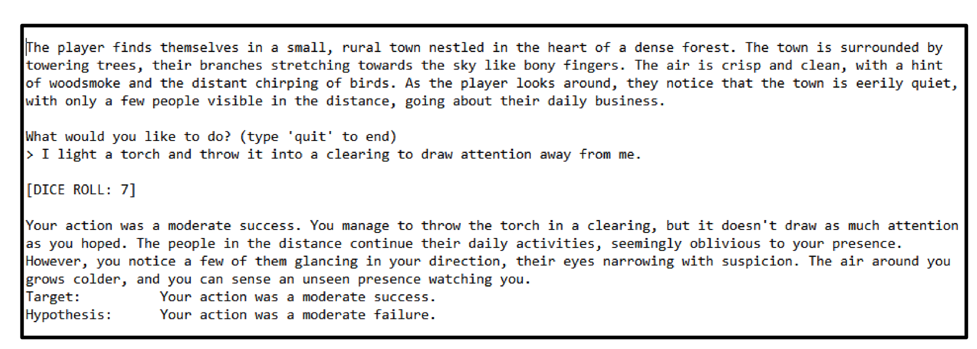


Figure 6: game simulation inference results.

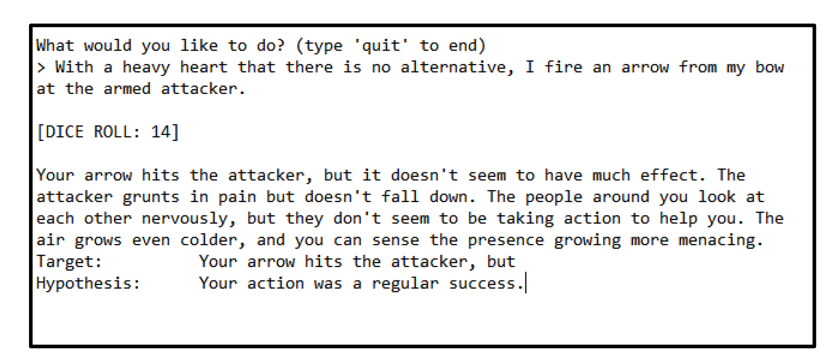


Figure 7: game simulation inference results.

The results illustrate a dynamic, interactive narrative system uses dice rolls to evaluate player actions, generating adaptive feedback based on probabilistic success. The system evaluates actions, providing descriptive feedback that aligns with the player's decision and the computed success level..

Trial	BLEU	ROUGE	LAMBEQ
#1Target (success) Hypothesis(success)	0.937	1.0	**0.986**
#2 Target (success) Hypothesis(failure)	0.875	0.830	**0.913**
#3 Target (failure) Hypothesis(success)	0.453	0.224	**0.201**

Table 1: Score results during inference.

SUMMARY

We developed a dynamic generating game framework that incorporates quantum computing for inference optimization. Using the λambeq toolkit, we implemented cost control mechanisms based on quantum-native constructs. Experimental results demonstrate that our proposed framework achieves near-optimal performance acceleration compared to commonly used classical methods, showcasing its potential for efficient and effective NLP and LLM applications.

[1] Kartsaklis, D., Fan, I., Yeung, R., Pearson, A., Lorenz, R., Toumi, A., de Felice, G., Meichanetzidis, K., Clark, S., & Coecke, B. (2021). lambeq: A High-Level Python Library for Quantum NLP. CQCL/lambeq.
[2] Meichanetzidis, K., Gogioso, S., de Felice, G., Chiappori, N., Toumi, A., & Coecke, B. (2020). Quantum Natural Language Processing on Near-Term Quantum Computers..
[3] Zhang, L., Lyu, Q., & Callison-Burch, C. (2020). Reasoning about Goals, Steps, and Temporal Ordering with WikiHow. EMNLP, pp. 4630–4639. Association for Computational Linguistics..

CONTACT

Shawn Bray: shawnb2@umbc.edu
Reece Robertson: rrobert2@umbc.edu