Quantum Computing Qubit Patent Classification with Long Short-Term Memory Recurrent Neural Networks

Tom Plunkett

Disclaimers

- Everything in this project is my personal opinion and does not represent the opinion of my employer.
- This presentation does not constitute legal advice.

Agenda

- Project Overview and Goal
- Dataset Selection
- Project Plan
- Exploratory Data Analysis
- Results
- Classification Accuracy Impact from Patent Family Members
- Conclusion

Overview

- Patent offices around the world issue millions of patents every year. It is difficult for companies with large patent portfolios to understand and manage their patent assets for a specific technology area.
- Machine learning can help automate tasks that have previously been entirely handled by human beings. This project is an attempt to demonstrate the ability of machine learning to classify patents as belonging in a particular category of asset, in this case whether a patent is related to Quantum Computing Qubit technology or not.
- I rely on patent metadata information (title, publication date, serial number, family id). While more accurate results would be achievable using the full text and drawings of a full patent document, that would require significantly more processing power.

Machine Learning and Deep Learning

- Machine Learning, which uses statistical models and algorithms to make predictions, includes Supervised Learning, Unsupervised Learning, Reinforcement Learning, Deep Learning, and other techniques
- Deep Learning, which uses Artificial Neural Networks to make predictions, includes Feed Forward Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks, Generative Adversarial Networks, Multilayer Perceptrons, and other techniques
- I selected Recurrent Neural Networks (RNN) because of their applicability to text analysis for my project with patent information. Long Short Term Memory (LSTM), Bidirectional LSTM, and Gated Recurrent Units are specific types of RNNs I chose to focus on.

Long Term Short Memory is a solution to the Vanishing Gradient problem in RNNs

- Recurrent Neural Networks suffer from short term memory. If a sequence is long enough, the RNN has a hard time carrying information from earlier time steps to later time steps.
- During back propagation, RNNs suffer from the vanishing gradient problem. Gradients are used to update a neural network weights during back propagation. The vanishing gradient problem is when the gradient shrinks as it back propagates through time. If the gradient becomes too small, the gradient won’t contribute to learning. The earliest layers are typically most affected. This leads to the short term memory issue.
- LSTMs, Bidirectional LSTMs, and GRUs are solutions to this short-term problem using Gates to regulate the flow of information. This way you can keep RNN information for the long term to make predictions.

Goal: Analyze Patents with Deep Learning

- I wanted to demonstrate the use of Long Short-Term Memory (LSTM) for analyzing sequential text information related to patents
- Patent Documents include the body of the patent document (specification text, claims text, and patent drawings pdfs) and patent metadata (text and numeric data)
- Analyzing the body of patent documents would require significant processing capabilities that I didn't want to utilize on this project
- I decided to focus on patent metadata for a binary classification problem using LSTM, bidirectional LSTM, and Gated Recurrent Units (GRU)

Dataset: Quantum Computing Gold Standard

- The next step was to find a dataset with labeled patent metadata that I could use for training data for my deep learning project.
- I checked Kaggle and didn't find anything that looked appealing. I then checked several patent journals and still didn't find what I was looking for.
- Finally, a search engine result led me to a site that then led me to a github site created by Steve Harris
- https://github.com/swh/classification-gold-standard
- This classification gold standard data set looked perfect for my deep learning project using LSTM, Bidirectional LSTM, and GRU
- The data set was created by Tony Trippe. Steve Harris, Tony Trippe, David Challis, and Nigel Swycher published an article about the dataset
- https://www.sciencedirect.com/science/article/pii/S0172219019300791

Binary Classification Problem Description

- Qubit Generation for Quantum Computing refers to patents that discuss the various means of generating qubits for use in a quantum mechanics based computing system.
- Positive Patent Documents: Types of qubits included superconducting loops, topological, quantum dot based and ion-trap methods as well as others.
- Negative Patent Documents: The excluded technologies are applications, algorithms and other auxiliary aspects of quantum computing that do not mention a hardware component, and hardware for other quantum phenomena outside of qubit generation.

Quantum Computing Gold Standard

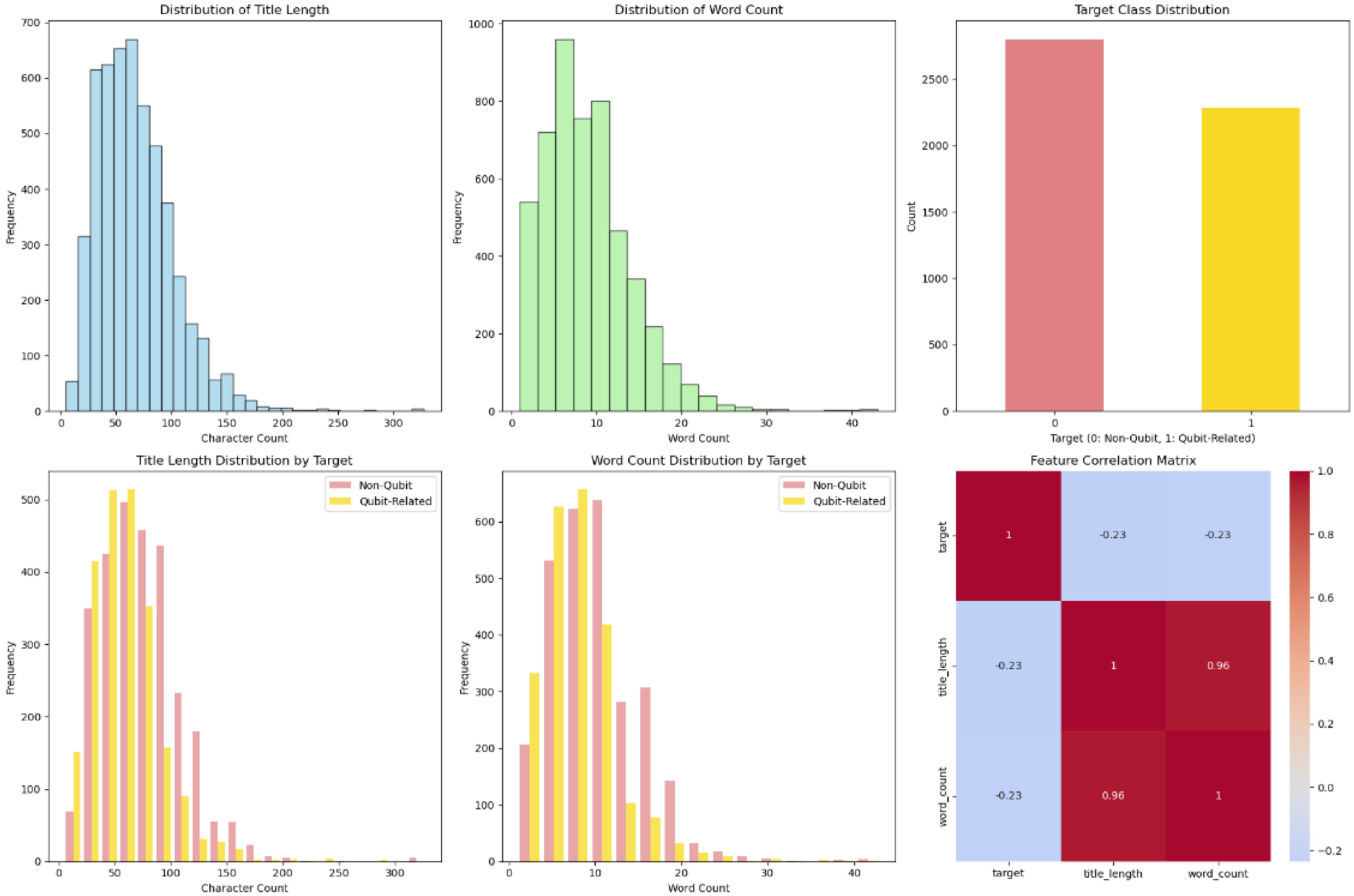
Class	DocDB Family ID	Serial no.	Title	Publication date
negative	10750653	AT196013T	NACHWEISUNG VON ELEKTROMAGNETISCHEN FELDERN	9/15/2000
negative	10750653	AU1703095A	Detection of electromagnetic fields	9/11/1995
negative	10750653	CA2183742A1	DETECTION OF ELECTROMAGNETIC FIELDS	8/31/1995
negative	10750653	CA2183742C	DETECTION OF ELECTROMAGNETIC FIELDS	8/5/2008
negative	10750653	DE69518629D1	According to signals of electromagnetic fields	10/5/2000
negative	10750653	DE69518629T2	According to signals of electromagnetic fields	5/3/2001
negative	10750653	DK0746773T3	Detection of electromagnetic fields	1/2/2001
negative	10750653	EP0746773A1	DETECTION OF ELECTROMAGNETIC FIELDS	12/11/1996
negative	10750653	EP0746773B1	DETECTION OF ELECTROMAGNETIC FIELDS	8/30/2000
negative	10750653	ES2152388T3	Detection of electromagnetic fields.	2/1/2001

3054	positive	32073118	US20040071019/	Method of forming quantum-mechanical memory and computati	4/15/2004
3055	positive	32073118	US20060264069/	Method of forming quantum-mechanical memory and computati	11/23/2006
3056	positive	32073118	US7042004B2	quantum system comprising computational elements, consisting	5/9/2006
3057	positive	32073118	US7541198B2	Method of forming quantum-mechanical memory and computati	6/2/2009
3058	positive	32109740	US6728281B1	Quantum-dot photon turnstile device	4/27/2004
3059	positive	32302354	US20040098443/	Sub-flux quantum generator	5/20/2004
3060	positive	32302354	US20050162302/	Sub-flux quantum generator	7/28/2005
3061	positive	32302354	US6885325B2	Sub-flux quantum generator	4/26/2005

Deep Learning Project Plan

- Load the Data
- Exploratory Data Analysis
- Data Preprocessing and Cleansing
- Keras Word Embedding Layer for Patent Title Vocabulary Tokens
- Create LSTM, Bidirectional LSTM, and GRU models
- Train Models and Determine Validation Accuracies for Models
- Evaluate Best Performing Model Against Test Data Set
- Analysis and Conclusion

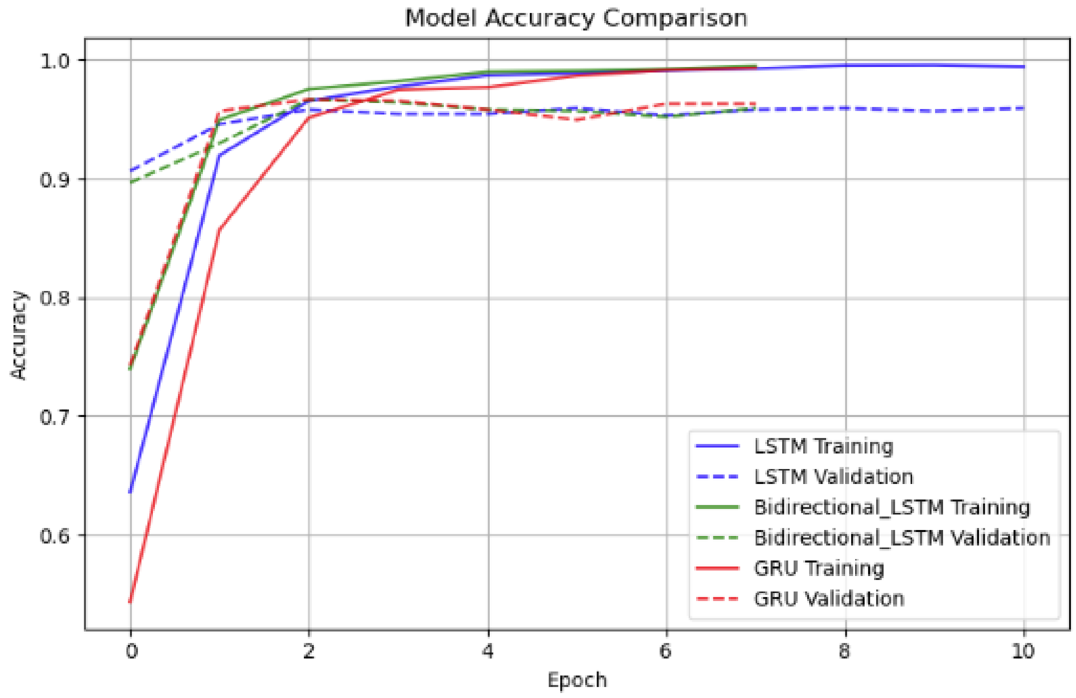
EDA

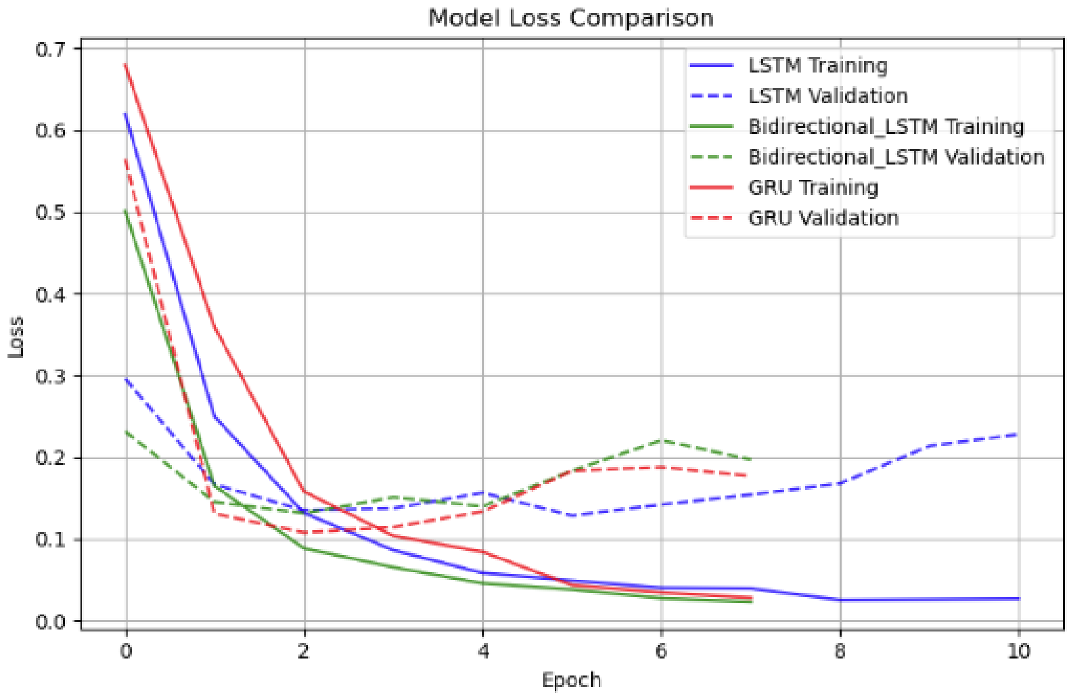


Distribution of Title Length
Frequency
Character Count
Distribution of Word Count
Frequency
Word Count
Target Class Distribution
Count
Target (0: Non-Qubit, 1: Qubit-Related)
Title Length Distribution by Target
Non-Qubit
Qubit-Related
Frequency
Character Count
Word Count Distribution by Target
Non-Qubit
Qubit-Related
Frequency
Word Count
Feature Correlation Matrix
target
title_length
word_count
1
-0.23
-0.23
-0.23
1
0.96
-0.23
0.96
1

Results with Full Data Set
Bidirectional LSTM was best, with other models very close to its results

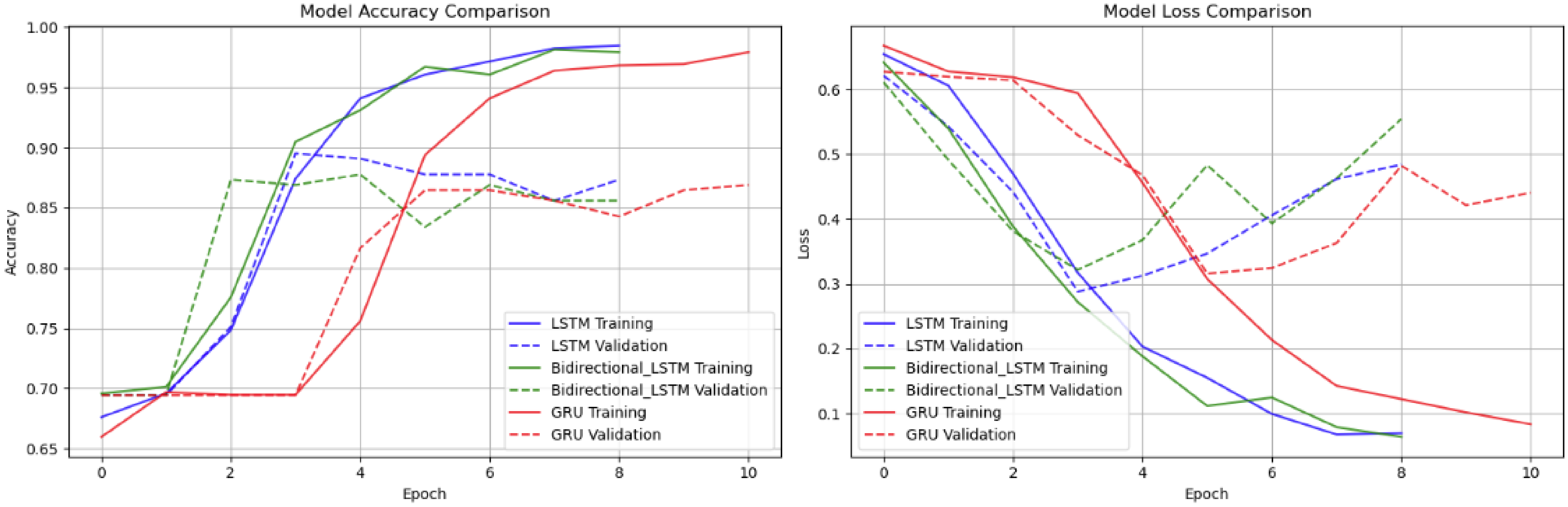
```
Final Test Accuracy: 0.9675
```





LSTM has best results with 1429 unique patent family documents (3654 additional patent family documents removed)

```
Final Test Accuracy: 0.8287
```



Patent Families Impact Classification Accuracy

- Why does the presence of additional patent family documents have such a big impact on patent accuracy, going from .8287 on the test set with a single entry from each patent family to an accuracy of .9675 on the test set with the full patent family?
- At first glance, I reasoned that the algorithm rapidly learned that if one patent family member is positive or negatively classified, then all remaining patent family members must have the same classification (i.e. 1.0 accuracy for 2^{nd} and subsequent family members)

Is Additional Patent Family Member Accuracy = 1.0 driving the .9675 accuracy rate?

- But this potentially does not explain all of the improvement as the math below shows:
- P (full_data)=.9675 (5083) = 4917.8
- P (1 patent_document per family)=.8287 (1429) + 1.0 (3654)=4838.2
- It is possible that the difference is within the margin of error, but an alternative explanation is that the algorithm is also learning which vocabulary words are most important by analyzing different wording differences between patent family members (i.e. more data leads to better results)

Conclusions

- My thanks to Tony Trippe, Steve Harris, and the team that created the Classification Gold Standard for Quantum Computing. I hope more groups release labeled classification data sets of patent information.
- I learned quite a bit about how LSTM, bidirectional LSTM, and GRU process text information with regards to Patent Titles
- I was impressed with the accuracy rates for classifying patents based on title information. The accuracy rates were higher than I expected.
- This project emphasized the value of fully understanding your data set and the relationships within your data set.