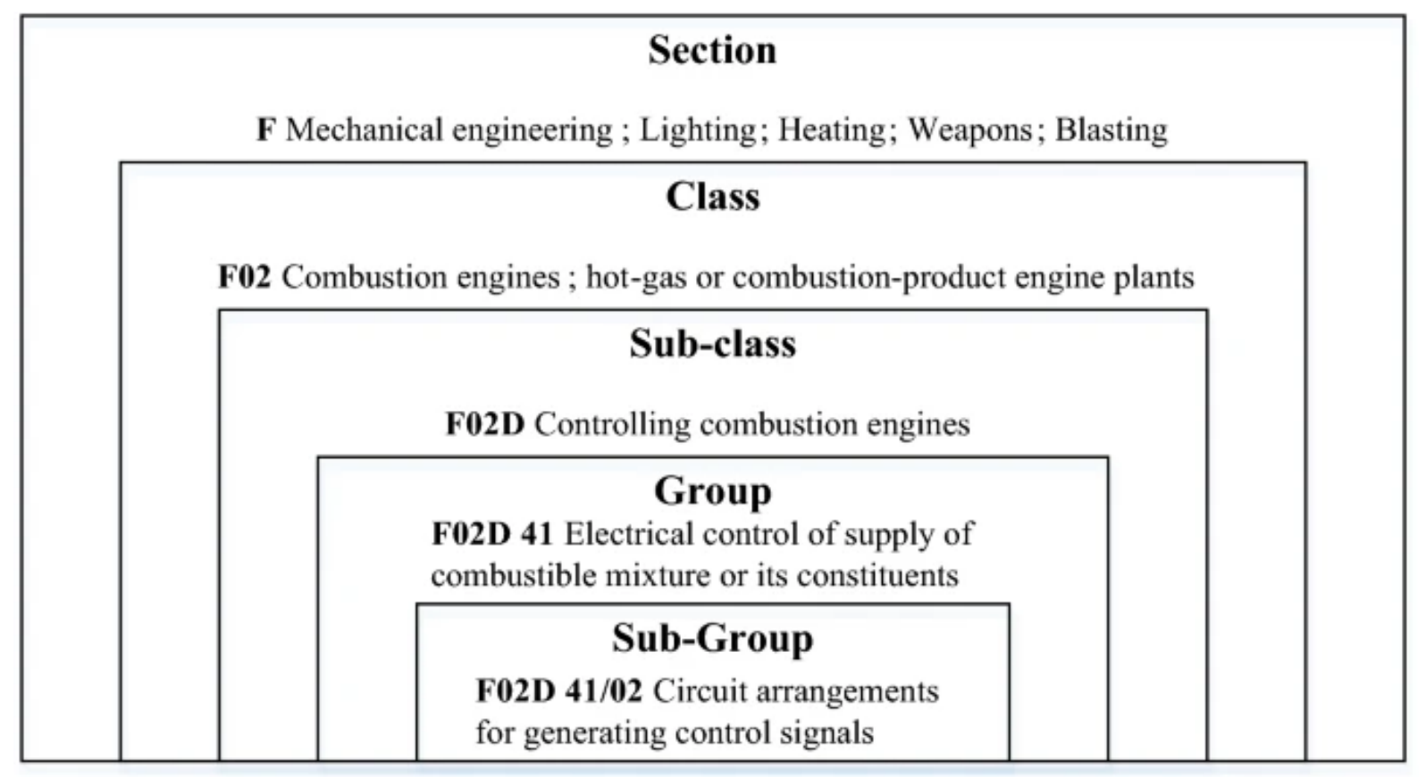
Multi-label Classification of Patent Text for Efficient Information Retrieval

Ashley Tarrant
tarrant.11@osu.edu
Department of Physics, The Ohio State University

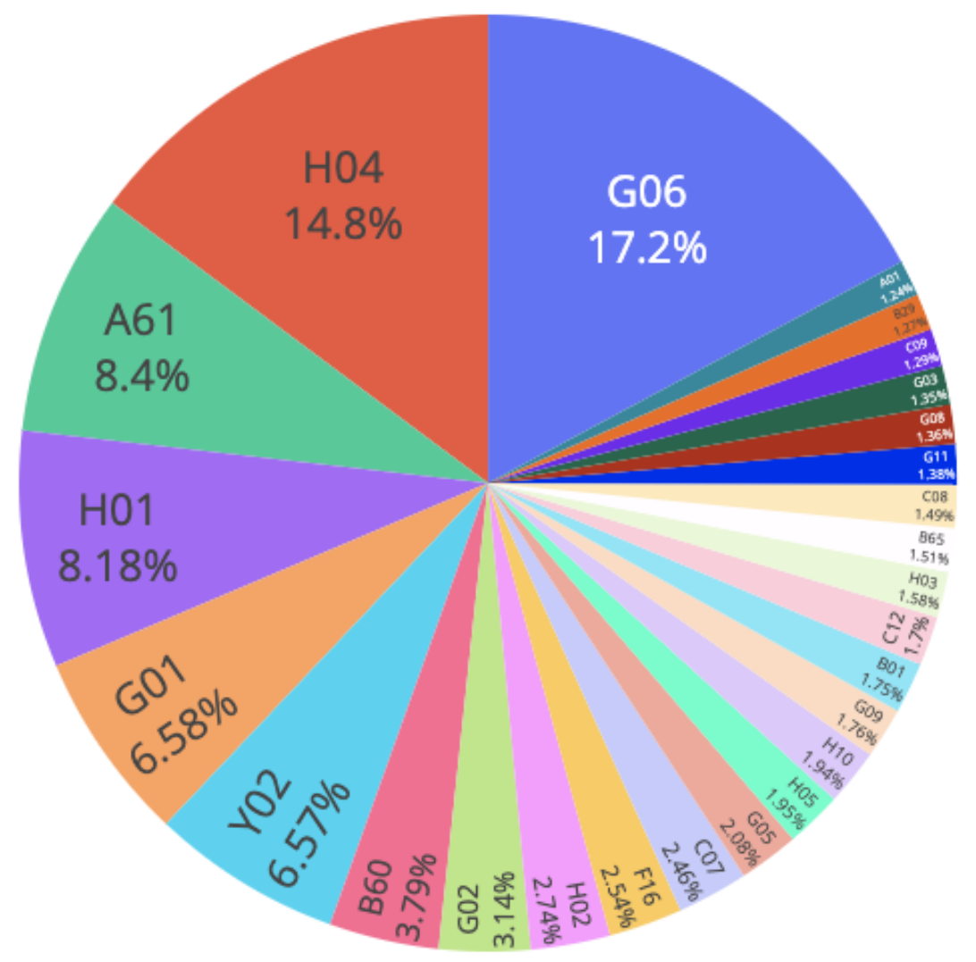
Motivation

Patents play a critical role in economic development by protecting inventor rights to encourage innovative products and services. Each year, the quantity and complexity of patent applications increases, which current operating systems are unequipped to manage. One key step in the patent granting process is its classification in a hierarchical scheme which categorizes the technical information in the patent for information retrieval and infringement prevention purposes. This task currently relies on domain-experts to read and interpret the new technology, but automation of this task would increase efficiency and accuracy of patent classification. Here, we use machine learning algorithms to predict the CPC class(es) a patent corresponds to given its title and abstract.

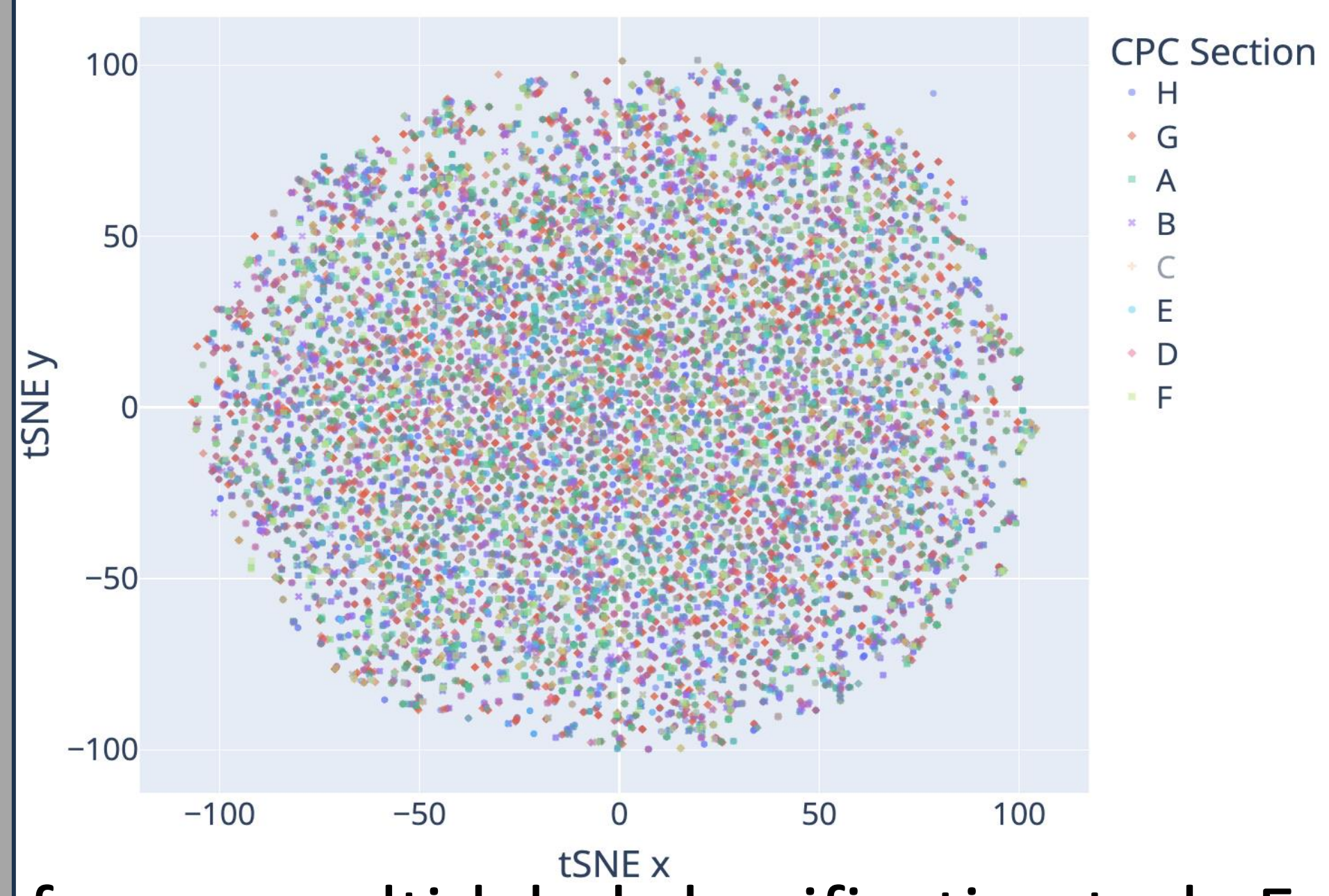

Section
F Mechanical engineering ; Lighting; Heating; Weapons; Blasting
Class
F02 Combustion engines ; hot-gas or combustion-product engine plants
Sub-class
F02D Controlling combustion engines
Group
F02D 41 Electrical control of supply of combustible mixture or its constituents
Sub-Group
F02D 41/02 Circuit arrangements for generating control signals

Dataset

We use publicly available data from the United States Patent and Trademark Office's PatentsView dataset. We randomly selected patents granted within the last five years with 8 or less corresponding CPC classes. Our dataset contains 130 CPC class labels, but recent technologies like optical computing devices and electrical communication systems dominate the sample as seen in the figure to the right.



We use the pretrained embedding model PatenstSBERTa to generate embeddings of the combined title and abstract of our patent data. PatenstSBERTa is trained on a corpus of millions of US patents providing ideal embeddings for our multi-label classification task. Each embedding is a 1x768 dimensional vector representing the semantics of the patent text. In figure above, we show a visual representation of 20% of the embeddings through t-Distributed Stochastic Neighbor Embedding (t-SNE) dimensionality-reduction. The classes are colored by CPC Section instead of class to increase readability. We one-hot-encode the 130 CPC classes for the labels in the dataset.



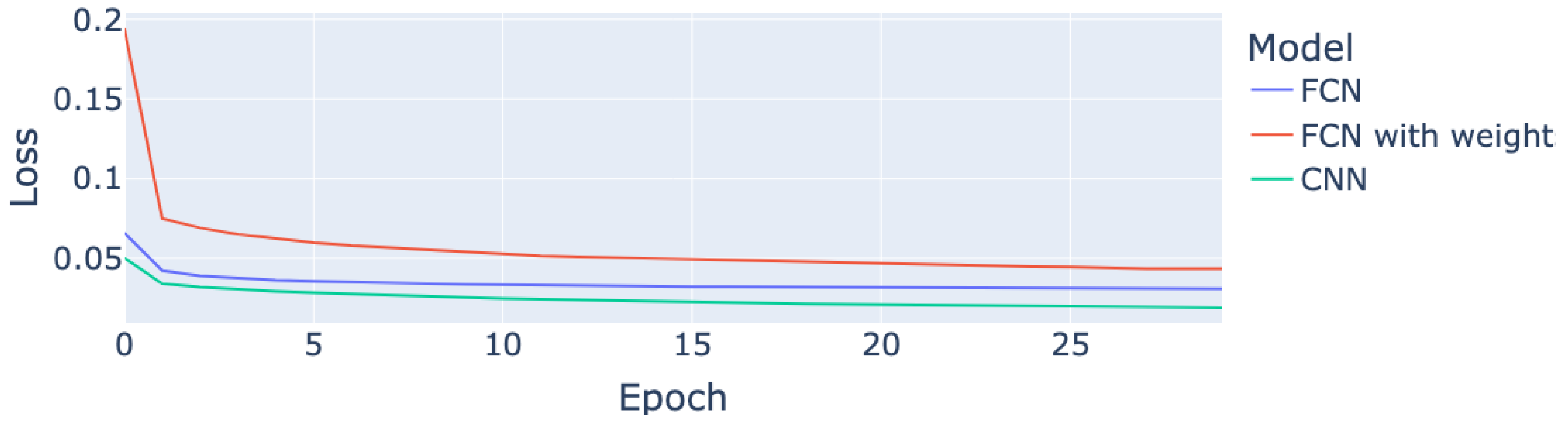
Number of Patent Samples	100,000
Number of Patent Classes	130
Average number of classes per patent	1.94

Methods

We evaluate the performance of three models, a fully connected network (FCN), an FCN with class weights, and a convolutional neural network (CNN). The three models are summarized below:

	FCN	FCN with weights	CNN
Total trainable parameters	340,730	340,730	2,470,638
Number of layers	12	12	7
Training time (for 30 epochs)	3 min	3 min	33min

Below we show the training curves for our models, which display consistent improvement throughout the training process. We use the binary-cross entropy function during training to minimize the loss.

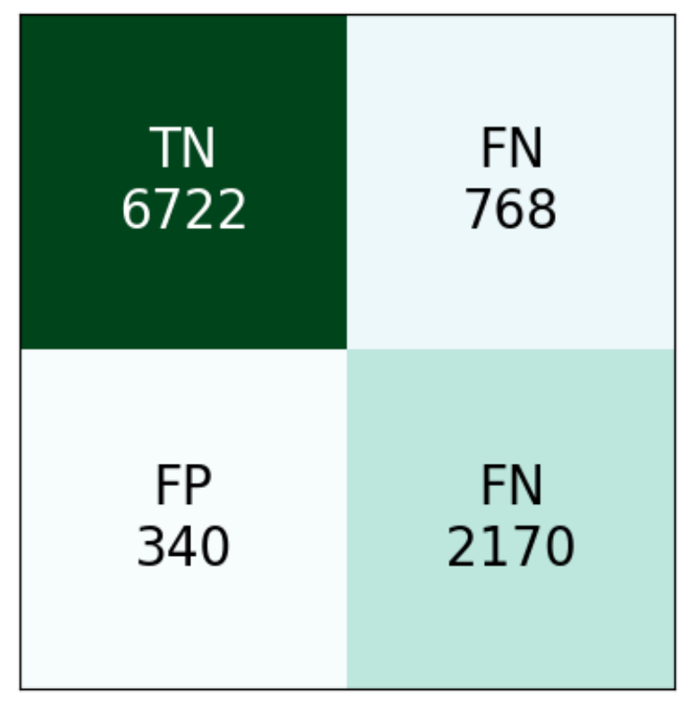


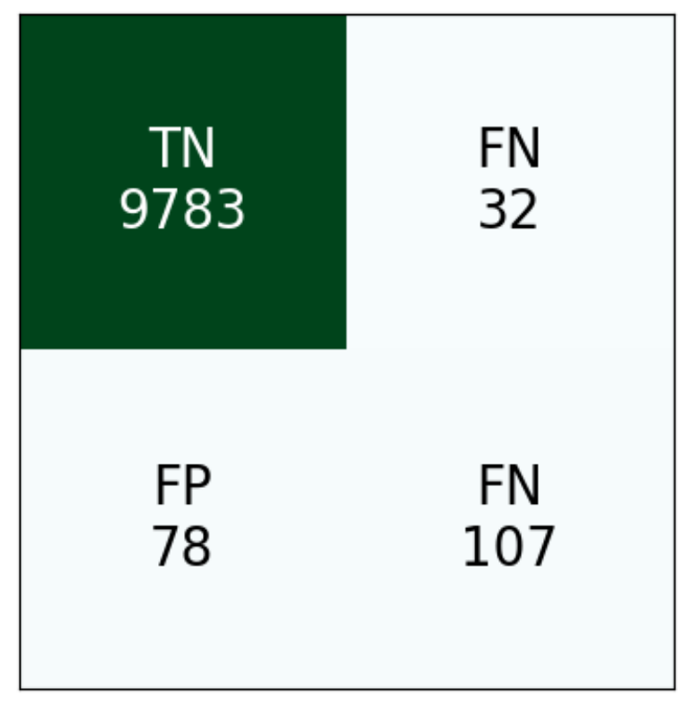
Data usage: Train (70%), Validation (20%), Test (10%)

Results

The performance of our models varies for each class, but generally the models accurately identify when a patent does not belong to a CPC class but struggle to predict the true positive class. In the table above, we show the macro-average performance metrics of the three models across all 130 classes. In all three metrics, the CNN outperforms the other two models.

	FCN	Weighted FCN	CNN
Avg. Precision	0.48	0.46	0.52
Avg. Recall	0.22	0.31	0.28
Avg. F1	0.28	0.33	0.34



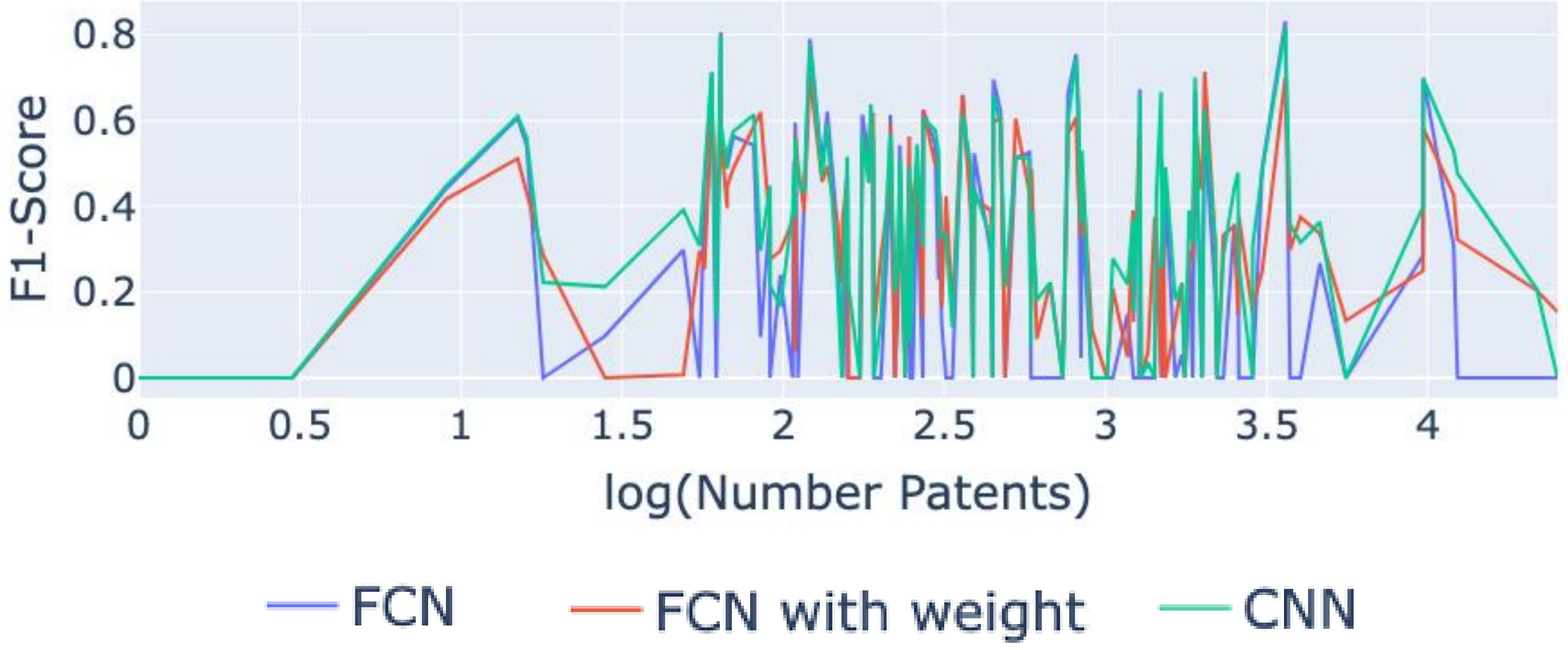


To the left we show two confusion matrices for the class with the best overall performance (left) and an average performing class for the CNN (right). The true negatives are accurately identified as shown by the dark green color, but the model misidentifies half as many patents as it predicts the correct class for in this case. The matrix on the right shows the typical model performance over our 130 classes, where the model labels as many false-positives and false-negatives as it finds true positives.

Discussion

The CNN outperforms both FCNs in all three metrics, likely because the convolutional and max pooling layers help focus more on key features in the embedding data representing technical vocabulary. Max pooling will also take the larger of two values, helping generate less frequent negative predictions.

Because our models do not perform as well as expected, we analyze the effect of the number of available patents per class on the overall model performance. In the figure below, we show the F1 score from the unseen test data as a function of the number of patents available for training and validation. There is no apparent relationship between the two, suggesting that our model does not struggle because of too little data, but rather due to the nature of our labeling scheme.



Conclusions and Future Work

We trained and evaluated three models to predict the CPC classification of patent text from text embeddings. We find that a 1D CNN outperforms the FCNs, but the model still struggles to generate precise class predictions. We suggest the following improvements:

- **Generate label embeddings as well as text embeddings**
  - We do not retain information about the hierarchical structure of patent classification; encoding labels would relate information to other classes, helping with under-sampled classes and more robust class separation
  - Replace sparse matrices resulting in negative predictions
- **Increase model complexity**
  - Vary the number of nodes, the number of nodes per layer
  - Experiment with dimensionality of convolution, as increasing the dimension could relate earlier text like the title to later text containing other technical vocabulary
- **Grid search for hyperparameters**
  - Vary learning rate and batch size to maximize performance
- **Incorporate more data from less frequent CPC classes**
  - Help to learn data from imbalanced classes instead of all negative predictions; provide stronger distinction between classes

References: Risch, J. and Krestel, R. (2019), "Domain-specific word embeddings for patent classification", Data Technologies and Applications, Vol. 53 No. 1, pp. 108-122. https://doi.org/10.1108/DTA-01-2019-0002; Chollet, Fran¸cois and others, Keras, 2015, https://keras.io; Devlin J., Chang M., Lee K., and Toutanova K., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, North American Chapter of the Association for Computational Linguistics, 2019, https://api.semanticscholar.org/