Classifying Research Proposals for ALMA using Neural Networks

Arnav Boppudi*†, Ryan Lipps*†, Brendan Puglisi*†
*School of Data Science
University of Virginia, Charlottesville, Virginia
†These authors contributed equally to this work.

Abstract— **Every year, astronomers from around the world submit research proposals to the Atacama Large Millimeter Array (ALMA), the largest radio telescope array in the world. The aim of our work is to streamline the proposal process for astronomers submitting projects to ALMA by suggesting frequency ranges that may be relevant to their research based on the title and abstract of their proposal. In this paper we assess the performance of BERT and SciBERT compared to simple machine learning models in three tasks involved in the project proposal process: project proposal binary classification, a high-level frequency band prediction based on technical limitations of the array, and a low-level measurement prediction based on engineered clusters of measurements from Hierarchical Density Based Spatial Clustering of Applications with Noise (HDBSCAN). All models are trained on the combined proposal and abstract of accepted proposals with various qualitative and quantitative interpretations of the technical setups for the projects. Ultimately simpler models like logistic regression provided the best balance between ease of implementation and accuracy for hard classification problems. However when predicting probability distributions for both the bands and the HDBSCAN clusters, more complex transformer models were needed to effectively capture this target.**

I. Introduction

The ALMA Observatory is a radio telescope array located in the Atacama Desert in Chile. ALMA is the largest astronomical project in existence and is open to anyone in the world to use via a project proposal process, which requires a technical plan outlining the specific settings and measurements the research team wishes to make. This can be daunting since the design of the telescope array allows ALMA to detect electromagnetic radiation on a fairly continuous range of frequencies from 35 GHz to 950 GHz at two-decimal-point precision. Furthermore, ALMA offers two high-level setups researchers may choose from to outline their measurements. "Line" setups focus on detecting specific frequencies associated with spectral lines of molecules, revealing chemical compositions and physical conditions of celestial objects. "Continuum" setups measure broad-spectrum emissions to understand the energy output, temperature, and structure of astronomical bodies. In the case researchers choose a "line" setup, they must set individual measurements that span a range of approximately 4 GHz into one of ten discreet frequency bands. According to these technical requirements, researchers face three critical decisions when preparing a project proposal: the decision to run a "line" or "continuum" project, and in the case of choosing "line", the decision of which band(s) to measure in, and which specific measurements to set.

On top of the technical nature of project proposals, ALMA can only accept around 400 proposals to enact throughout the year, making it a valuable and competitive resource. Thus, it is paramount that researchers submit well designed and technically proficient proposals to increase their chances of acceptance.

The goal of our work is to streamline the ALMA proposal process by providing researchers insight and assistance on how to prepare the technical setup, using the combined title and abstract of their proposal. First we predict whether or not they will conduct a "line" or a "continuum" project. After this classification, we attempt to predict the broader frequency band a "line" project will measure in, followed by the specific measurements that may be useful for their research.

II. Methodology

Given the successful adaptation of BERT to various natural language processing tasks, we evaluate the performance of BERT and SciBERT on three tasks according to the technical specifications described above: one to classify a project as "line" or "continuum" (*line/continuum classifier*), the second to predict the frequency bands a project might place measurements in (*band prediction*), and a third to predict the specific measurements a project might make (*measurement prediction*). Even though tools such as Hugging Face are lowering the barrier to entry in deploying complex models such as BERT and its many variations, we compare BERT and SciBERT's performance to simple machine learning methods where relevant. This provides context to model accuracy with the intention of justifying the choice of model compared to the knowledge and computing resources required to deploy them.

For the *line/continuum classifier*, we compare the performance of BERT and SciBERT to both a simple Logistic Regression using TF-IDF vectorized text as a quick and popular machine learning approach, and a Multilayer Perceptron using TF-IDF vectorized text as a computationally lightweight neural network approach. These models are trained and tested on both "line" and "continuum" projects. Each of the models are evaluated on the standard binary classification metrics of overall accuracy and class precision, recall, and F1 scores.

For the *band prediction* model, we compare BERT and SciBERT to Multinomial Naive Bayes on two interpretations of prediction. For the first interpretation we treat the problem of predicting band as a multi-class classification problem where each model predicts the single most probable band a project should measure in. In this approach we compare models on classification accuracy. For the second interpretation we treat the problem as predicting the probability distribution of the bands a project should measure in. In this approach we compare model performance on the mean squared error between a ground-truth vector generated from the proportion of measurements a project has placed in each band to the corresponding predictions. It is important to note that these projects are only trained on "line" projects as band is not relevant to "continuum" projects.

For the *measurement prediction* model, we extend the probability distribution interpretation of the *band prediction* approach described above to a finer-grained task by predicting the distribution of a significantly larger number of precise areas a project may measure in. To do so, we use Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) to generate clusters of measurements made by all "line" projects. We then create ground-truth vectors from the proportion of measurements a project has in each of the HDBSCAN clusters and train BERT and SciBERT to predict these vectors. As such a task is generally too complex for traditional machine learning models, we have no baseline comparison to measure BERT and SciBERT against. Thus, we compare the two models to each other on mean squared error between the ground-truth and predicted vectors, while offering a simple comparison of accuracy to the similar interpretation from the *band prediction* models. Accordingly, this approach is also trained only on "line" projects.

III. Data Discussion

A. *Text Processing*

The full dataset consists of 4,528 accepted projects (3,628 line projects, 900 continuum projects), each with a title and abstract that we concatenated as input for our models. These combined texts vary from 16 to 560 characters in length, with four projects accounting for lengths greater than 507 characters. As BERT models can accept texts with a maximum of 512 characters, we dropped these four projects. The remaining 4,524 projects serve as the input for the *line/continuum classifier* models. We used the specific tokenizers for both BERT and SciBERT to process the text for the two transformer models. For the Logistic Regression, Multilayer Perceptron, and Multinomial Naive Bayes models, we stripped whitespace and removed punctuation, numeric characters, and stopwords from the nltk stopword list. Furthermore, we lemmatized the text using the nltk lemmatizer to reduce words to their inflected forms with the purpose of reducing the input feature space for these models. We then converted the text to a vectorized form using Term Frequency-Inverse Document Frequency (TF-IDF) using scikit-learn's TF-IDF vectorizer. The equation this vectorizer uses is shown in Equation 1 below.

$$\text{TF-IDF}(q_i, p_a) = \frac{q_{ia}}{w_a} * [log\left(\frac{N+1}{u_i+1}\right) + 1], \qquad (1)$$

where q_i is the given word for $i = 1, 2, \dots,$ the total number of words in p_a. p_a is the project text for $a = 1, 2, \dots,$ the total number of project title and abstract texts. Furthermore, q_{ia} indicates the number of the times q_i appears in p_a, and w_a indicates the number of words in p_a. N is the total number of project title and abstract texts, and u_i is the total number of occurrences of q_i in N project title and abstract texts. [1]

For the binary classification models, we used the existing labels of "line" and "continuum" as targets and used an 80/20% train/test split. We offer an examples of both a "line" and a "continuum" title and abstract in the appendix.

B. *Line Project Measurement Discussion*

As mentioned previously, each "line" project may consist of numerous measurements across any of the ten bands, placed at different frequencies. It is important to note that a single project can consist of both line and continuum measurements. To account for this, we categorized a project as "line" if it had at least one line measurement. Otherwise, we classified the project as a "continuum" project. The data consists of 3,628 accepted "line" projects that account for 44,232 unique measurements. Figure 1 shows the distribution of measurement count per project.

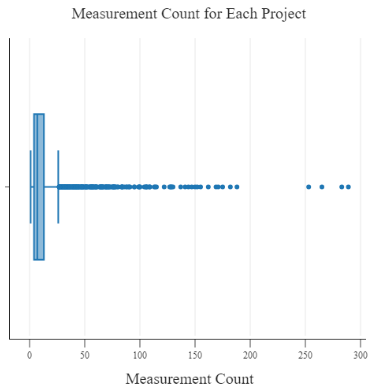

Fig. 1: Measurement Count for Each Project. 75% of projects have 13 measurements or less.

In Fig 1., we see outliers in the distribution of the number of measurements across projects. These outlier projects can often be attributed to specific uses of ALMA that are outside the scope and scale of our process. Therefore, we removed these projects to prevent them from having too much influence in model training. We identified outliers as projects with more than 26 measurements (3rd Quartile + 1.5 * Interquartile Range).

Measurements in the raw data are defined by a frequency lower-bound and upper-bound. From these, we engineered a median frequency, being the mid-point between the high and low frequency for each measurement. This serves as a simple way to define measurements by a single value, with the assumption that a measurement is defined by its center, as opposed to its range. The distribution of “line” project median frequencies is shown in Figure 2.

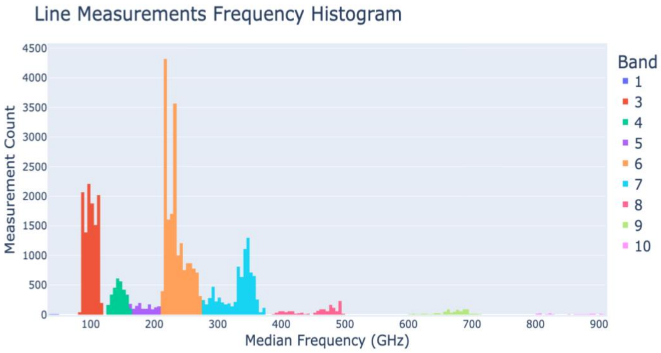

Fig. 2: Distribution of measurements defined by median frequency with color for band. Most measurements are below 500 GHz and occur in bands 3, 6, and 7.

By counting the number of measurements (defined by median frequency) that fall into each band, we found that 81.54% of projects only have measurements within one band. Nonetheless, one project has measurements in 6 different bands. So that such projects do not disproportionately influence the predicted frequency ranges, we removed projects that have measurements in more than 2 bands. Additionally, we removed projects that have incorrectly formatted band information. It is worth noting that there are no measurements in band 2.

These cleaning efforts reduced the data to 3,178 “line” projects and 23,482 measurements. Finally, we used 80% of these projects (2,383 projects with 17,638 measurements) to train, and 20% of the data (795 projects with 5,844 measurements) to test the *band prediction* and *measurement prediction* models.

IV. Related Work

The increasing popularity of BERT has seen it applied to many text processing tasks, especially in comparison to more “traditional” machine learning models. For instance, Santiago González-Carvajal and Eduardo C. Garrido-Merchán compare “BERT against the traditional TF-IDF vocabulary fed to machine learning algorithms” [2], similar to our approach for the *line/continuum classifier* task. In their work, González-Carvajal and Garrido-Merchán compare BERT to various TF-IDF-based machine learning approaches on a binary sentiment analysis task, and find that BERT performs significantly better than the other approaches, achieving 93.87% accuracy, compared to Logistic Regression’s 89.49% accuracy. This is notable, as for our binary classification task, Logistic Regression outperforms base BERT and is only just behind Sci-BERT in accuracy. We believe this is due to the technical nature of the astronomical research proposals imposing a limitation on the model accuracies. The texts are heavy in numerical measurements and units, molecules, and abstract naming conventions of galaxies and other celestial bodies that are uncommon if not non-existent in texts outside the astronomy community. Thus we believe our research shines a light on the importance of building a corpus to train and adapt BERT to specifically astronomical texts.

With respect to the *band prediction* and *measurement prediction* approaches, our work is similar to the idea of using neural networks for learning data distributions. However, we have not found literature involving the application to a distribution that does not follow a specific statistical distribution with hyperparameters that the network can learn. Recall Figure 2, which shows that our data does not follow any typical statistical distributions.

V. Line Continuum Classification

A. Logistic Regression

As a baseline model for comparison to the more in-depth neural networks, we first employ a simple Logistic Regression in the *line/continuum classifier* task. For this model, the input text is processed using the methods outlined in “Text Processing”. When assessed with the testing data, this approach had an accuracy of 86.02% using a cutoff of 0.5 to make hard classifications into “line” and “continuum”. Table I shows a comparison of all models on this binary classification task. It is notable that despite its simplicity, the Logistic Regression model performs comparably across all metrics, when compared to the transformer models we introduce later. This makes it a compelling model for this task, as it is especially quick and requires a fraction of the storage compared to the other models in this section.

B. Multi-layer Perceptron

For comparison to a simple neural network, we trained a Multilayer Perceptron (MLP) using the TF-IDF vectorized text described in “Text Processing”. The MLP model is a type of neural network that is especially suitable for classification tasks where inputs are mapped to outputs through multiple layers of neurons. The MLP architecture we use has 20 hidden layers, each containing 64 nodes. The deep structure allows the network to learn the complex patterns in the data. For each of the hidden layers we use a Rectified Linear Unit (ReLU) activation function to introduce non-linearity into the model, making it capable of learning more complex relationships in the data. We use a dropout rate of 20% to each layer during the training to prevent overfitting, which randomly ignores a subset of features during each iteration. For the output layer we use a sigmoid activation function to produce a probability indicating the likelihood of a project being a ”line” project. We trained this model over 1000 epochs with an early stopping implementation

TABLE I: Logistic Regression Performance Comparison

Model	Accuracy (%)	Line			Continuum		
		Recall (%)	Precision (%)	F1 (%)	Recall (%)	Precision (%)	F1 (%)
Logistic Regression	86.20	93.93	89.69	91.76	51.52	65.38	57.63
Multilayer Perceptron	75.28	72.74	96.08	82.80	86.67	41.45	56.08
BERT	86.09	93.66	89.78	91.68	52.12	64.66	57.72
SciBERT	86.64	93.12	90.79	91.94	57.58	65.07	61.09

based on validation loss to optimize learning without overtraining. We used binary cross-entropy for the loss function as it effectively handles binary classification tasks, while the Adam optimizer ensures efficient updates to network weights. The model achieves an accuracy of 75.28% on testing data using a cutoff of 0.75 to make hard classifications into “line” and “continuum”. The MLP model performs the worst on this binary classification task, though this appears to be due to the network overfitting to the training data. Figure 5 in the Appendix shows this overfitting. It is important to reiterate that we implemented early stopping to serve the best model according to validation accuracy for comparison in this section. We believe that this overfitting is ultimately due to the high number of input features, being the entire TF-IDF vectorization of the corpus, whereas selecting important features as inputs may help this architecture generalize better.

C. BERT Models

Finally, we evaluate BERT and Sci-BERT models on the *line/continuum classifier* task. BERT stands for Bidirectional Encoder Representations from Transformers and is utilized to understand the contextual relationships between words in a project's text. By leveraging a pre-trained model, BERT can effectively predict the category of project based on the sequence of words in both project title and abstract. Specifically, we used a bert-base-uncased pre-trained model and integrated a final classification layer to predict whether project texts fall under either “line” or “continuum”. As opposed to using TF-IDF as the input, BERT has its own tokenizer that processes text inputs into a format suitable for the model, generating token ids, segment ids, and attention masks. The max sequence length is set at 512 tokens.

To ensure robustness in model training and evaluation, we partitioned the dataset into training, validation, and testing segments. We used the AdamW optimizer, as it is known for its efficiency in handling sparse gradients and adaptive learning rate capabilities. Furthermore, we implemented a learning rate warm-up phase that starts at 5e-5 and gradually increases the learning rate, followed by a linear decay to fine-tune model responses. We found a batch size of 16 was ideal for our model, balancing computational demands and model performance. We trained our model for 5 epochs with an early stopping mechanism that based on validation loss to prevent overfitting and ensure efficient training. Early stopping is crucial in machine learning training processes to halt training when the validation loss does not improve, thereby saving computational resources and preventing the model from learning noise in the training data.

We also utilized SciBERT (allenai/scibert scivocab uncased [3]), a variant of BERT specifically pre-trained on a large corpus of scientific texts, to enhance our binary classification of project types into ”line” or ”continuum”. SciBERT's architecture is adapted from BERT and includes specialized token, segment, and position embeddings that are ideal for understanding scientific jargon and concepts, which are critical in analyzing our dataset comprising of technical project proposals. The model is integrated with a sequence classification layer atop the base BERT architecture, facilitating the direct prediction of binary labels from text inputs. We employed SciBERT's tokenizer to preprocess text data into a suitable format. For the sake of fairness between models, we parameterized SciBERT exactly the same as BERT.

The BERT model demonstrates solid performance in classifying project types, achieving an overall accuracy of 86.09%. It shows particular strength in identifying ”line” projects, with a recall of 93.66%, precision of 89.78%, and an F1 score of 91.68%, indicating its efficacy in capturing most ”line” projects accurately. For ”continuum” projects, BERT has a recall of 52.12%, precision of 64.66%, and an F1 score of 57.72%. These metrics suggest BERT performs adequately, but with room for improvement in identifying ”continuum” projects, likely due to the dataset's imbalance.

Comparatively, SciBERT, which is adapted specifically for scientific texts, achieves a slightly higher overall accuracy of 86.64%. For ”line” projects, SciBERT's metrics are slightly lower than BERT's, with a recall of 93.12%, precision of 90.79%, and an F1 score of 91.94%. However, it shows a more balanced performance for ”continuum” projects with a recall of 57.58%, precision of 65.07%, and an F1 score of 61.09%. This indicates that SciBERT may be more attuned to the nuances of scientific language, enhancing its ability to classify ”continuum” projects more effectively than BERT.

Both models are robust, yet SciBERT's slightly superior performance in identifying ”continuum” projects suggests its specialized training on scientific literature provides a slight edge in handling classifications within this more specific domain. This comparison highlights the importance of model choice depending on the specific requirements of the classification task and the nature of the text data being analyzed.

VI. Band Classification

A. Multinomial Naive Bayes

To predict the probability a project includes measurements within a specific band, we implemented Multinomial Naive Bayes, a machine learning algorithm popular with text classification. Formally,

$$P(c|d) = \text{argmax}_{c \epsilon C} \left(P(c) \cdot \prod_{1 \leq k \leq n_d} P(t_k|c) \right), \quad (2)$$

where c is the given band $(1, 3, 4, 5, 6, 7, 8, 9, \text{ or } 10)$, d is the TF-IDF vectorized project text, n_d is the number of tokens in d, t_k is a term occurring in the vectorized project title and abstract text. [4] Since we removed outlier projects in more than two bands, the goal of this approach is to identify the two bands that are most likely to be of a wide range of interest for each project.

In Fig 3, we display the distribution of bands by measurement count in the training data. As shown in Fig. 3, the distribution of measurements within bands is not uniform. To account for this imbalance, we specified the prior probabilities of a measurement corresponding with each band using the following formula:

$$w_j = \frac{n}{(C * n_j)} \text{ for } j = 1, 3, 4, 5, 6, 7, 8, 9, \text{ or } 10, \quad (3)$$

where w_j is the weight for each band, j indicates which band, n indicates the total number of samples, C indicates the total number of bands (9), and n_j indicates the total number of instances of band j. [5] This helps the model optimize accuracy for less common bands. As mentioned previously, we processed the text using the steps in “Text Processing”.

The output of this model is a vector of predicted probabilities that the project corresponds with each band. We measure the accuracy of this model’s results in two interpretations: one where we treat *band prediction* as a multi-class classification problem, where we classify a project according to its most probable predicted band, and one where we treat the predicted vectors as a probability distribution we want to “learn”. In the case of the multi-class classification interpretation, the Multinomial Naive Bayes model achieved 47.31% accuracy on test data. For the interpretation where the predicted vectors are treated as probability distributions, we evaluate performance on the mean squared error between a ground-truth vector generated from the proportion of measurements a project has placed in each band to the corresponding predictions. This model achieves a mean squared error of 0.148 for this task. Table II shows performance for all models on the two interpretations of the *band prediction* task.

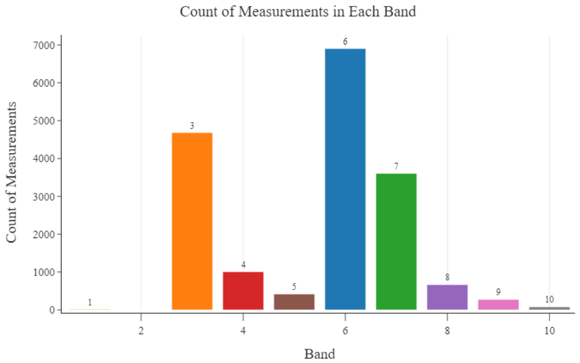

Fig. 3: Count of Measurements in Each Band. Bands 3, 6, and 7 are notably the most common bands across all measurements in the training data. Meanwhile, there are little to no measurements in bands 1 and 10. There are no measurements in band 2.

B. BERT

BERT (Bidirectional Encoder Representations from Transformers) demonstrates a noteworthy proficiency in classifying text into measurement frequency bands, as evidenced by its accuracy and mean squared error (MSE) on the task. With an accuracy rate of 50.06% in hard classification, BERT outperforms both Multinomial Naive Bayes and SciBERT in determining the most likely measurement frequency band for a given scientific text. This suggests that BERT’s deep transformer architecture is effective for this multi-class classification problem. As done for the Naive Bayes approach, we use the argmax operation to select the band with the highest probability as the predicted class from the full output vectors. Furthermore, BERT’s MSE of 0.0041252 signifies its capacity to predict not just the most likely class, but to also provide reliable probability distributions across all bands. This is essential for scenarios where understanding the likelihood of each band is as important as identifying the most probable one. BERT’s lower MSE, compared to Multinomial Naive Bayes, indicates its superior capability in estimating these distributions accurately. Instead, Multinomial Naive Bayes is trained to perform hard classification and is not optimized for predicting probability distributions. When translating these metrics into practical terms, BERT’s predictions are, on average, approximately 6.4% off the true

TABLE II: Comparison of performance between Multinomial Naive Bayes, BERT and SciBERT on the *band prediction* task. All models have similar accuracy, though BERT and SciBERT have notably lower MSE. We believe this is due to the transformer models having the flexibility to learn the distribution of the data explicitly, whereas Multinomial Naive Bayes is specified for classification tasks.

Model	**Accuracy (argmax hard classification)**	**MSE**
Multinomial Naive Bayes	47.31%	0.148074
BERT	50.06%	0.0041252
SciBERT	49.81%	0.0041395

probability values, which is a modest deviation given the complex nature of the data. BERT's nuanced performance is likely attributed to its comprehensive pre-training on a vast corpus of text, allowing it to generalize well across different domains, including scientific texts. This makes BERT a robust choice for applications where accurate and detailed predictions of measurement frequency are required.

C. Sci-BERT

As mentioned previously, SciBERT is a specialized version of BERT that has been pre-trained on a corpus of scientific texts. While it utilizes the same fundamental transformer architecture as BERT, SciBERT is enhanced with a vocabulary that is particularly rich in scientific terms. It shows nearly comparable hard classification accuracy to BERT with 49.81%, indicating its specialized training has equipped it to understand and process scientific texts effectively. While SciBERT's accuracy is marginally lower than BERT's, its MSE of 0.0041395 is nearly identical to that of BERT, underscoring its precision in generating probability distributions. This close MSE value to BERT suggests that SciBERT's pre-training on scientific texts makes it almost equally adept at accurately predicting the probability distribution for different bands in scientific text classification tasks. In practical terms, SciBERT's slightly larger MSE implies that its predictions for the probability that a given project falls within a specific measurement frequency band are only marginally off than BERT's when considering the average probability. It is important to note that SciBERT trains almost two times as fast as the BERT Large model we used. This is a major advantage to using SciBert and proves its efficacy at predicting scientific texts.

VII. Measurement Prediction

In order to achieve our ultimate goal of providing measurement recommendations, we cluster the measurements from all of the projects using Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [6]. HDBSCAN is an extension of Density-Based Spatial Clustering of Applications with Noise (DBSCAN), an unsupervised machine learning algorithm that generates estimated clusters in the data based on the density of the data. DBSCAN is primarily parameterized by a reachability distance, ϵ, around each point and the minimum number of points (*min_cluster_size*) within ϵ that quantifies "dense". While DBSCAN is a useful starting point for generating clusters the static value of ϵ can generate clusters that are too large to be of value to a researcher submitting a proposal, due to the wide distribution of measurements. In contrast, HDBSCAN adjusts ϵ to account for varying densities across the input data. This is ideal for our purposes as it generates clusters that are representative of the overall distribution of measurements while highlighting areas where many measurements are made.

The parametrization for HDBSCAN we used was a *min_cluster_size* of 20 measurements to generate a cluster. Across all the projects, measurements that are not matched to a cluster are labeled "-1" to indicate noise. This allows us to reduce the number of possible measurements from 17,638 to a much more manageable range of 343 clusters. For each project we engineer a ground truth vector of length 343 representing each possible cluster. The values of this vector are dependent on the number of measurements that the specific project had in that cluster. We subsequently applied normalization to this vector by scaling its elements such that their sum equates to 1. Essentially this vector represents the probability distribution across the clusters for each project. Figures 6 and 7 in the Appendix give a visual representation of these clusters across the entire frequency range.

A. BERT Experiments

The first model we used was BERT Large Uncased. Similar to the band probability distribution, we compare the accuracy of our models using mean squared error (MSE). This effectively allows us to compare the performance of different models to one another, as well as achieve interpretable results later using the root mean squared error (RMSE). Ultimately this large BERT model achieved a MSE of 5.8381e-05, while SciBERT achieved a very similar MSE of 5.8415e-05. This confirms our previous analysis that these models performed virtually the same for our use case. However, it is critical to note that once again SciBERT trained more than two times faster than the large BERT model.

These MSE values are orders of magnitude smaller than the MSE obtained from the band distribution prediction, but to have an accurate comparison we must consider the target size it is relative to. The average probability across the 343 vectors is $\frac{1}{343} = 0.0029$. Converting the MSE to the original probability unit by computing the root mean squared error we get 0.0076. Evidently, the RMSE is over two times the size of the average probability. This proves that the HDBSCAN model performs much worse than the band distribution prediction which had a RMSE of 0.064, half the size of the relative average probability of $\frac{1}{9} = 0.1111$. This significant increase of error in the measurement distribution prediction is expected due to the significant increase in possible targets.

Despite this poor performance relative to the band distribution prediction, it is still important to note that this BERT model exhibited signs of learning. The distribution of the 343 clusters was captured rather accurately, but simply did not represent the extremes of the actual values. For example, if the probability of a cluster was high the model always under-predicted it and if it was low the model over-predicted it. Evidently there was an aspect of regularization occurring that was forcing the predicted values to be constrained when compared to their true extremes. This is a sign that the models were under fitting our data and that more training should be done. Due to resource constraints only four epochs of training were completed, but increasing this along with varying the learning rate could result in a model that more accurately

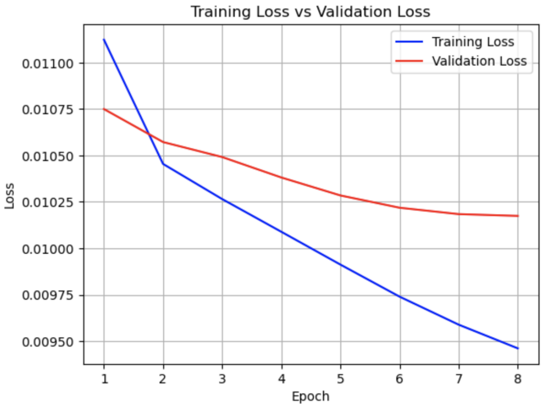

Fig. 4: Loss graph of the BERT model shows the model is over-fitting on training data and under-fitting on validation data

represents the extremes of our data. These results can be seen both in Figure 4 above, representing the training loss graph, and Figure 8 in the Appendix, representing the average of cluster predictions versus the average of the true distributions.

VIII. Conclusion

In essence, all the models we explored are viable options for classification problems, including the binary and multi-class problems. Due to the very similar results in these scenarios, using simpler models like the logistic regression are preferred due to their ease of implementation. However, for more complex problems like predicting probability distributions, transformer models outperform simpler models by far and should be used despite their elaborate setup and long run-times. while both BERT and SciBERT are capable performers for predicting the band and measurement frequencies of scientific texts, SciBERT's training on a specialized corpus offers it as a compelling alternative to BERT, particularly for those in the scientific community who require an algorithm attuned to the nuances of their literature. As well, SciBERT has training times almost twice as fast as the large BERT model with very similar final results, making it the ideal model for this use-case.

IX. References

References

[1] W. Scott, "Tf-idf from scratch in python on a real-world dataset." [Online]. Available: https://towardsdatascience.com/tf-idf-for-document-ranking-from-scratch-in-python-on-real-world-dataset-796d339a408

[2] S. González-Carvajal and E. C. Garrido-Merchán, "Comparing BERT against traditional machine learning text classification," *CoRR*, vol. abs/2005.13012, 2020. [Online]. Available: https://arxiv.org/abs/2005.13012

[3] I. Beltagy, K. Lo, and A. Cohan, "Scibert: A pretrained language model for scientific text," 2019.

[4] "Naive bayes text classification." [Online]. Available: https://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html

[5] K. Singh, "How to improve class imbalance using class weights in machine learning?" [Online]. Available: https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/

[6] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," *Advances in Knowledge Discovery and Data Mining. PAKDD 2013. Lecture Notes in Computer Science()*, vol. 7819, no. null, pp. 16–172, 2013. [Online]. Available: https://doi.org/10.1007/978-3-642-37456-2_14

Appendix

A. Example of a Line Project Title and Abstract - Project 2011.0.00236.S

Title: The Dynamics of Massive Starless Cores. Abstract: Progress towards resolving a decade-long debate about how massive stars form can be made by determining if massive starless cores exist in a state of near virial equilibrium. These are the initial conditions invoked by the Core Accretion model of McKee and Tan (2003). Alternatively, the Competitive Accretion model of Bonnell et al. (2001) requires sub-virial conditions. We have identified 4 prime examples of massive (=50 Msun) cores from mid-infrared (MIR) extinction mapping (Butler and Tan 2009, 2011) of Infrared Dark Clouds. We have found spectacularly high deuterated fractions of N2H+ of 0.5 in these objects (Fontani et al. 2011). Thus N2D+(3-2) becomes an excellent tracer of the kinematics of these cold, dark cores, where most other molecular tracers are thought to be depleted from the gas phase. ALMA Cycle 0 Compact Configuration Band 6 observations probe this line on scales from 9" down to 2.3", well-matched to the structures we see in MIR extinction. Sharing a 5 hour track between single pointing observations to each of the 4 cores, we have the sensitivity and uv plane coverage needed to measure the kinematics of these structures and thus determine whether or not they are near virial equilibrium.

B. Example of a Continuum Project Title and Abstract - Project 2011.0.00101.S

Title: Shedding Light on Distant Starburst Galaxies Hosting Gamma-ray Bursts v9.

Abstract: Studies of distant starburst galaxies hosting gamma-ray bursts (GRBs) offer unique insights into extreme star-forming regions during early epochs. We propose to carry out a pilot program to observe the 345 GHz continuum from the host galaxies of GRB021004 and GRB080607 at z ¿ 2 with ALMA. The selected targets show contrast examples in the host galaxy population in the observed neutral gas surface mass density in front of the GRB birth site. The host galaxy of GRB080607 exhibits a large gas surface mass density of 400 Msun pc-2, including a large molecular gas column density in the afterglow spectrum. In contrast, the host galaxy of GRB021004 exhibits ionized ISM and complex velocity field in the afterglow spectrum. Both hosts have been identified with associated stellar light in late-time HST images and have constraints for the ISM metallicity from afterglow absorption-line measurements. In addition, the early-time afterglow spectra of the GRBs have revealed the presence of strong Mg II absorbers at z = 1.5. We aim to obtain a deep sub-mm map of the fields around the two GRB host galaxies with a 5-simga limit of 0.5 mJy in the 345 GHz waveband. This sensitivity limit is an order of magnitude improvement from previous single-dish observations of these fields that yielded null results. We expect that the proposed observations will allow us to resolve the extragalactic background light in the sub-mm and to constrain the dust luminosity of these luminous GRB host galaxies. The proposed pilot program will offer important insights into both the progenitor environment and the contribution of dusty starburst galaxies to the GRB host population at $z > 2$. It will also allow us to examine the dust luminosity of strong Mg II absorbers in the foreground.

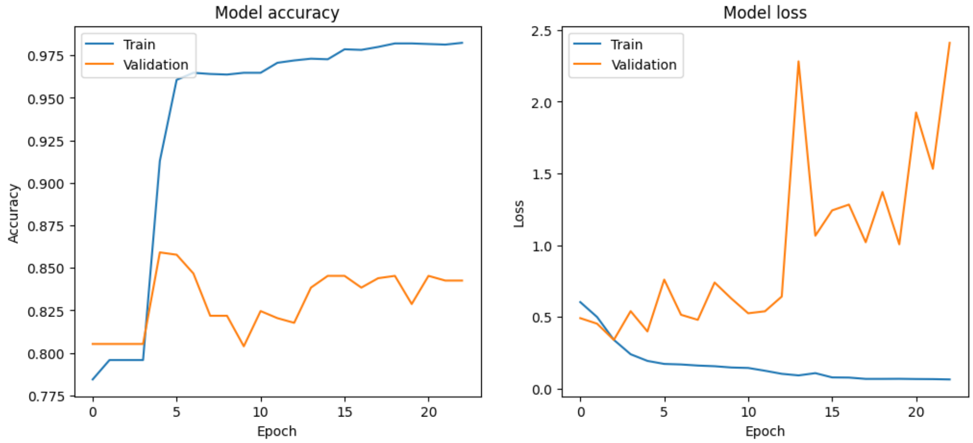

Fig. 5: Graphs showing the training and validation accuracy and loss for the Multilayer Perceptron on the *line/continuum classifier* task. Despite adjustments and hyperparameter tuning the model consistently overfits to the training data, while validation and test loss suffer. It is important to note we implemented early stopping to serve the best performing model according to validation loss for the comparison to the other models on this task.

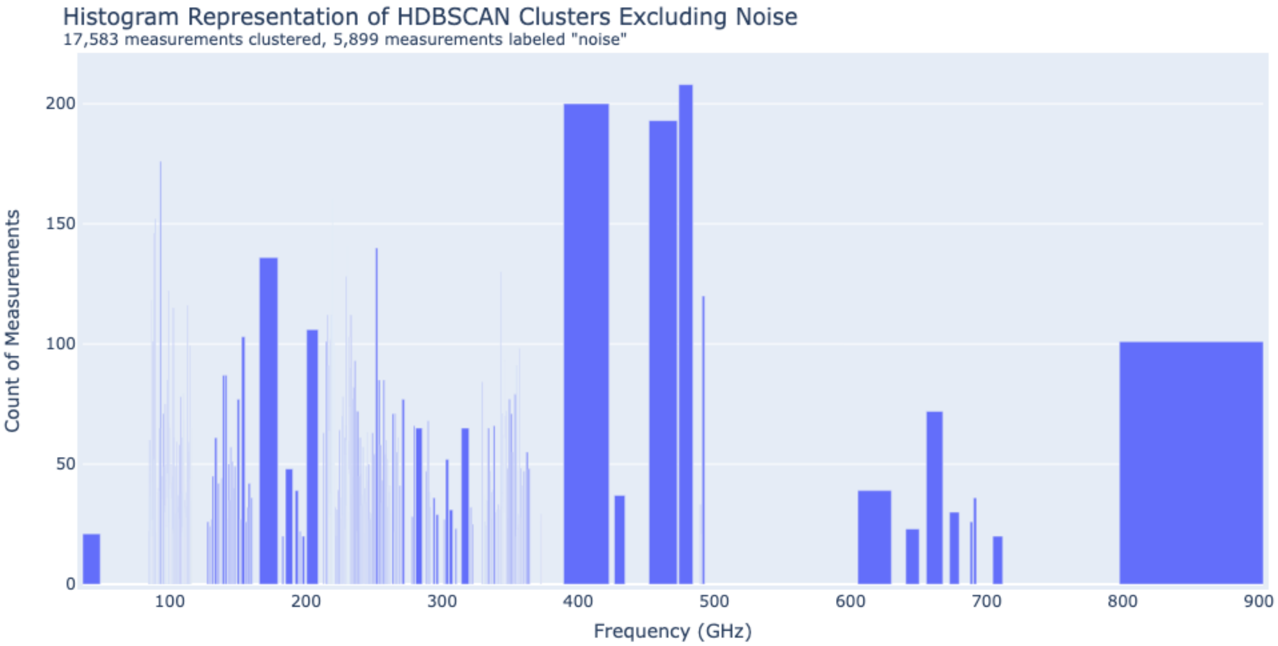

Fig. 6: Inspection of measurement clusters from Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN). This plot excludes “noise” measurements not assigned to a cluster. Histogram bin width (and similarly cluster width) is defined by the minimum and maximum median frequency assigned to each cluster. The mean cluster width is 1.229 ± 6.402 GHz. The mean count of measurements in clusters is 51.412 ± 31.924 measurements. These clusters provide a reduced set of targets to predict while maintaining a distribution representative of the raw measurement data.

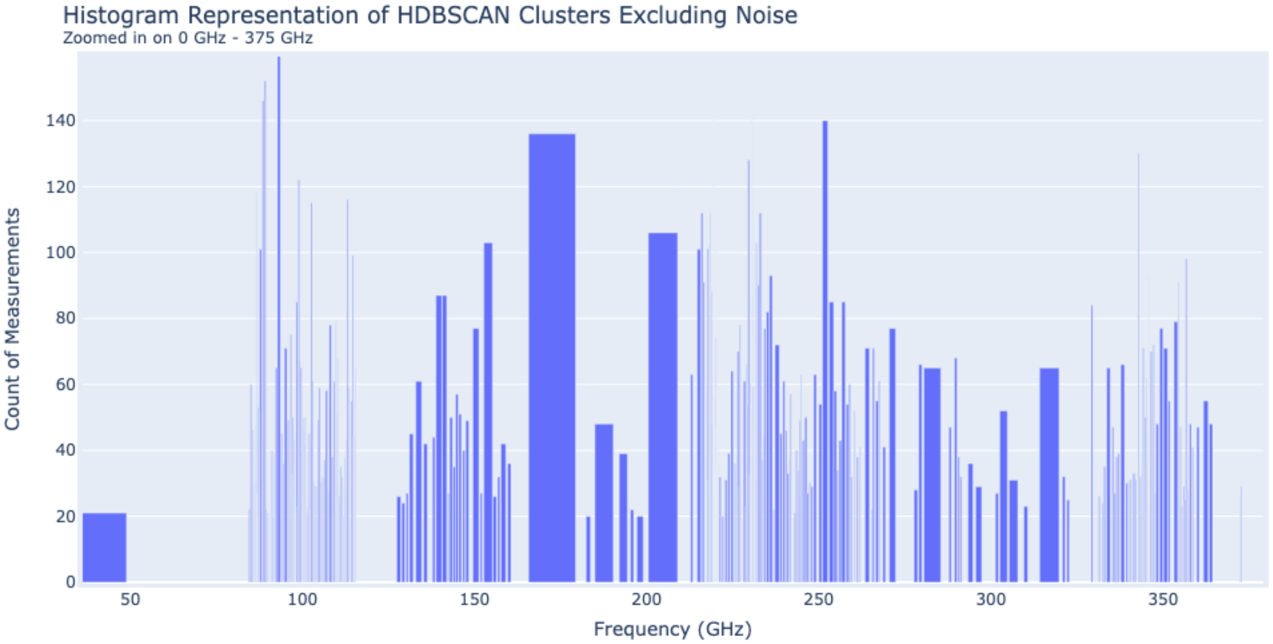

Fig. 7: Inspection of Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) clusters zoomed in to show many “narrow” measurement clusters in the range of 0 - 350 GHz.

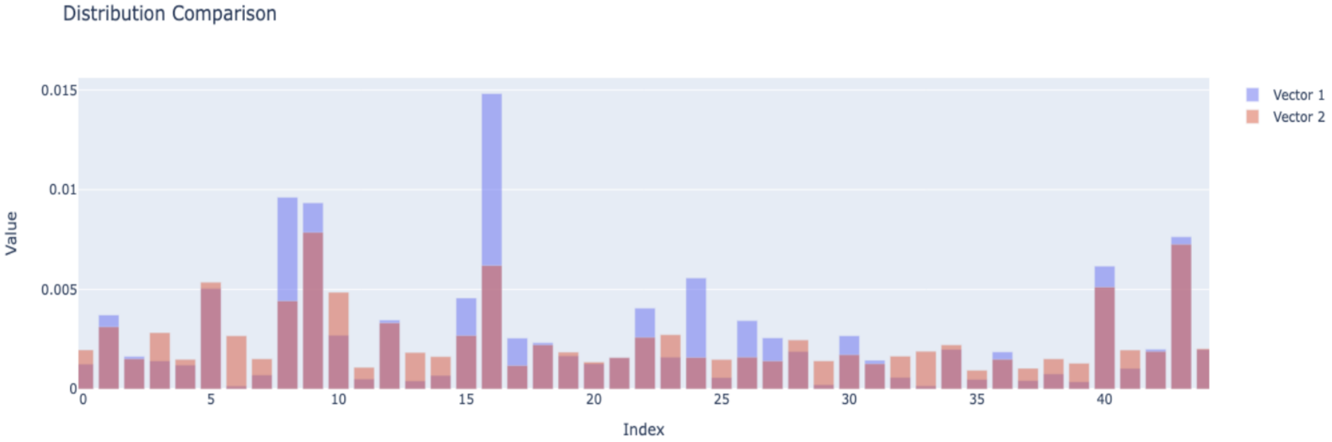

Fig. 8: Bar graph showing comparison of average ground-truth HDBSCAN distribution (Vector 1, blue) to predicted distributions from BERT (Vector 2, red). The X-axis “Index” refers to HDBSCAN cluster index with the Y-axis “Value” representing the proportion of measurements in that cluster. Note that our model generally under-fits the data, predicting low proportions when the ground truth is high, and predicting high proportions when the ground truth is low.