Predicting If A Kiva Loan Posting Will Expire

Project Final Report

Dylan Connor
Carnegie Mellon University
Pittsburgh, PA
dmconnor@andrew.cmu.edu

Kaung M. Khin
Carnegie Mellon University
Pittsburgh, PA
kkhin@andrew.cmu.edu

1 INTRODUCTION

Kiva is a non-profit organization that organizes and facilities micro-funding to support entrepreneurial efforts in underdeveloped regions living in poverty. As part of that mission, they work with local partners in high poverty regions to identify people looking for relatively small loans to promote business essentials. Ideally, these loans are paid off in a timely fashion, and investors can reinvest that same money to another person who needs it.

In this endeavor to connect people through lending to alleviate poverty,[12] Kiva must successfully vet opportunities for people that are of higher risk of defaulting, so they can communicate that information to willing investors. Additionally, it can be important to identify what characteristics highlight those that will not get the funding they are requesting, so that they can either promote them more effectively or choose not to pick them up to begin with. While we would like to delve into factors that identify defaults, the data available does not provide the granularity required for this analysis.

It was these considerations that caught our eye the most when we identified the Kiva dataset on Kaggle in promotion of data science for good. We were excited at the prospect of data science for good, helping to better understand the micro-funding landscape. With the above in mind, we identified the following major ML task:

Expiration Classification: The primary classification task is to identify loans that will expire. That is, loans that will go up on the site, but fail to be fully funded. If Kiva can accurately identify if a request is going to go unfunded, then they can abandon the request or provide additional marketing to give it the extra push it may need to get funded.

2 DATA UNDERSTANDING AND PREPARATION

Our primary data set is the Kaggle Kiva, Data Science for Good dataset.[11] The data provided in the competition is relatively clean data in form of four csv files.

The first contains instances of individual loans including dates, locations, category, number of lenders, the amount loaned, and the amount paid back (671,205 instances).

The second file contains multidimensional poverty index data by region (1009 regions) as well as latitude and longitude. This data can be merged onto the loans dataset using country. There are inconsistent region data that make matching the two files more granularly than at the country level extremely hard. For this analysis, we have chosen to ignore the sparse region specific MPI in favor of the country specific MPI, which we collected from University of Oxfords Oxford Poverty & Human Development Institute.[1]

The third file contains records that match the loans by ID and identify regional theme IDs and regional themes (779092 instances). The last file has Loan Theme ID by country and by region as its primary key. It contains regional data related to loan themes and partner ids (15736 instances). Without cleaning and matching regions manually, this file is hard to meaningfully merge with the loan data due to inconsistent and non-overlapping region names For this analysis, we have grouped on Partner ID and Loan Theme ID, summed the count and amount and averaged the latitude and longitude.

In addition to the Kaggle data, we identified another data source. Kiva has a RESTful API that allows certain additional requests of the data, including a data dump of current data that is readily available.[3] From this source, we were able to get the status field (among many other fields) that forms the foundation of our classification task. Unfortunately, in the current data dump provided by the API, there are no loans classified as defaulted. There is data available from the wayback machine that does identify some defaults but does not overlap with the timeframe of the data provided by Kaggle. For that reason, we are focusing on prospective loan postings that expire, without being fully funded.

Our final dataset for the machine learning task of classifying relies on: loan amount (int), activity (categorical with 264 labels in total we kept all that contained more than 0.5% of the total loans and labeled the rest as Other) , sector (categorical with 15 labels), country (categorical with 78 countries), currency (categorical with 47 labels), term in months (int), repayment interval (categorical with 3 labels), original language (categorical with 5 labels), days to expiration (int), year (categorical with 4 labels), month (categorical with 12 labels), MPI (float), female and male counts (int number of females and males in the requesting part) and partner id (categorical with 198 labels).

3 METHODOLOGY

Our work streams are divided into three primary tasks. The first task was feature selection. The second, was relevant preprocessing to prep the data for modeling. Finally, was the actual modeling and classification task.

Team 22, May 4th, Pittsburgh, PA
2018.

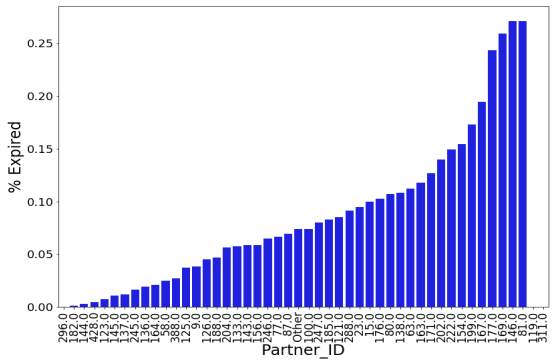

Figure 1: % Expiration by Partners

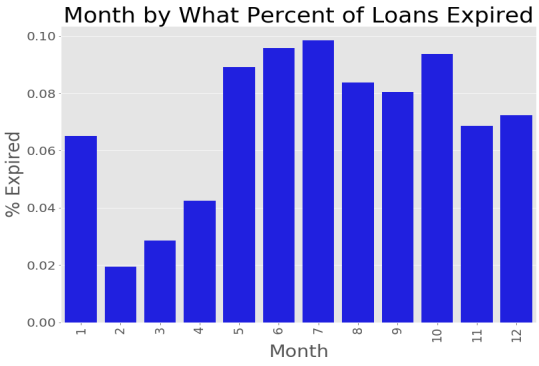

Figure 3: % Expiration by Month

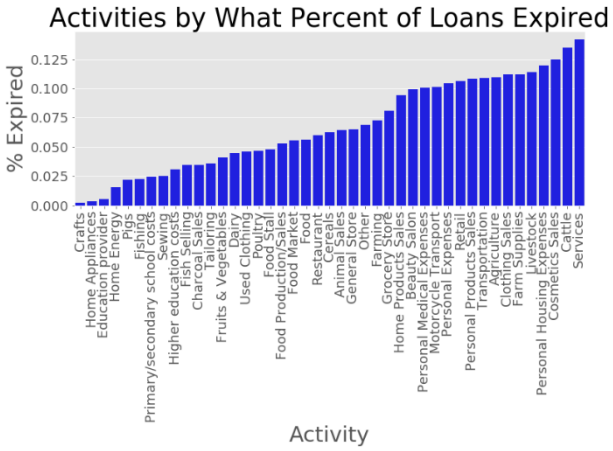

Figure 2: % Expiration by Activites

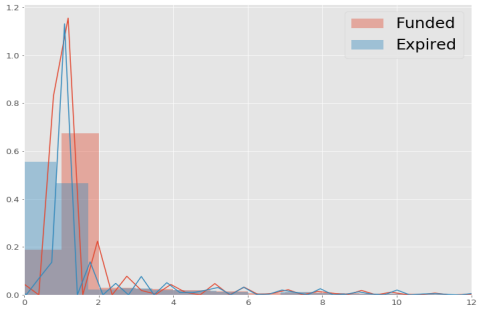

Figure 4: % Expiration by Activites

Table 1: Variables by Unique Counts

Variable	Unique Count
activity	44
sector	15
country	78
currency	61
partner id	48
repayment interval	3
male count	40
year	4
month	12

3.1 Feature Selection

For the first task, we ended up with the features mentioned above mostly by being over inclusive and keep all fields that seemed like they could have any predictive power in our analysis. We generally only removed fields that had sparse amounts of data. To validate that the various fields we left in had explanatory value on our decision variable, we considered and analyzed various graphs that highlighted variance over the decision variable (see Figures 2, 3 as example below). Additionally, we reconsidered variables in an iterative process after running initial models. The major preprocessing steps included:

- Data Wrangling
- One-hot encoding all categorical variables
- Normalizing the data
- Dealing with the unbalanced dataset

The data wrangling for this project mostly related to getting all of the datasets merged properly and deciding what rows and columns would be dropped for sparsity reasons. From a starting point of 671,205 rows, ultimately 52,932 were dropped. This left the final data set with 618,813 rows, of which 38,879 were expired and 579,934 were funded.

For the one hot encoding, we simply created dummy fields for every unique value in our categorical variables. For activities and Partner IDs, any factor that had over 0.5% of the loan data was kept. Anything below this number was encoded to Other. The categorical variables with their final counts are as follows:

3.2 Preprocessing

For the normalization we used min-max scaling across the data set. With fields such as loan amount have a large range compared to many other fields, it was important to normalize for algorithms that are affected by unnormalized feature spaces.

Since our final population had only about 6% classified as expired, we needed to deal with the unbalanced nature of the dataset. Our primary means of doing so was through SMOTE. After applying the SMOTE algorithm, we were left with 927,894 observations. Additionally, we considered modeling methods that would be less effected by the unbalanced data.

3.3 Modeling

For the classification task, our goal was to first get a baseline model running. We ran a Nave Bayes model on the unbalanced data, as our baseline. See Figures 20 and 19. This provided a reasonable baseline for our analysis. It had many false positives, highly overpredicting the minority class. However, it does a reasonable job, with an F1 score of .86.

After getting a baseline model working, we started to address the imbalance issue as well as try different models and increase our hyperparameter tuning. We analyzed the results using Synthetic Minority Over-sampling Technique (SMOTE) [7] as well as Adaptive Synthetic Sampling Approach for Imbalanced Learning (ADASYN) [10] as an under/over sampling technique in the hopes of improving our metrics. We chose this data level approach of rebalancing the class distribution because it is usually more versatile.[9]

3.4 Text Description Analysis

Additionally, separate from the full feature space analysis, we attempted a neural network based analysis on the description field that was also available in the data. In this analysis we first cleaned the data by lemmatizing, removing stop words and separating the words. We then visualized a word cloud of the most frequently represented words in each classification category, we can see in Figure 5 that most funded posts were related to school, businesses and even working capital. This is in contrast to the words we see for the unfunded posts in Figure 6, most of these words seem related to livestock with words such as fertilizer, cow and chicken.

It is also important to note that we can see in Figure 7 that most of our data comes from non-English speaking countries. As such most of the descriptions have been translated from their native language to English probably by the partner. This resulted in over 10,000 records being duplicated which placed a heavy burden on the analysis since each description was not longer than a few sentences as well. We found that because of the inherent biases baked into the text description, we ultimately decided not to include the text analysis into our models as it did not perform well and might even skew the performance of our models down.

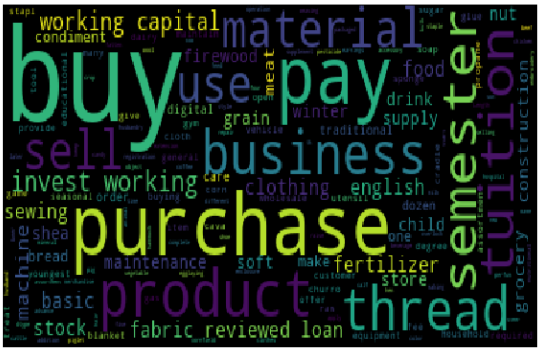

Figure 5: Word Cloud of Funded Posts

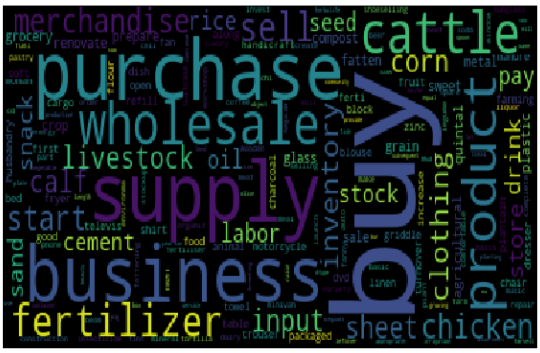

Figure 6: Word Cloud of Expired Posts

4 EVALUATION AND RESULTS

Since the data set is inherently geographical, we started with a small exploratory data analysis exercise to visualize the locality of the data we were working with. In Figure 7 below, every yellow circle indicates at least 2000 loans that were funded in the data set and every red cross indicates at least 50 loans that expired in the data set. We see that there are locality clusters in areas such as south South-East Asia, Latin America and Coastal Parts of Africa.

For the classification task, when we were cross-validating and evaluating our results, we considered several metrics. We had very unbalanced data and were particularly interested in increasing true positive rate, while not letting the false positives get too high. Intuitively, this made the most sense given what we set out to do. We see not identifying loans that will ultimately expire as the costliest component in the confusion matrix. If Kiva can identify these loans, they can either waste less resources on supporting the loan or employ specific targeted marketing tactics to help ensure securing the loan. Metrics aside, when we were looking at a confusion matrix, what we wanted to see was high true positive without unreasonably high false positive.

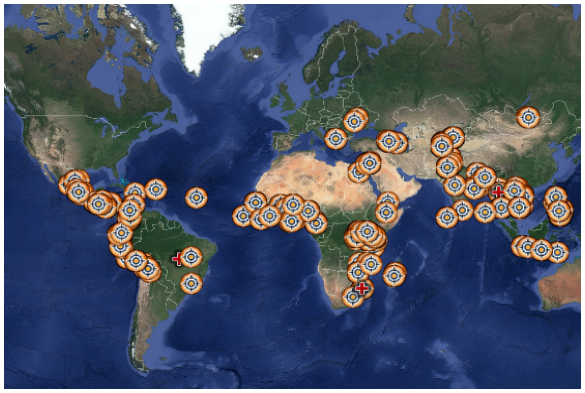

Figure 7: Global Map of Expirations

		Predicted Label	
		0	1
True Label	0	97184	18803
	1	3720	4056

Table 2: Confusion Matrix of Naive Bayes

With this in mind, we cross-validated using F1 score. As a metric of precision and recall, F1 score considers the balance discussed above quite well. Additionally, we considered the ROC curve and precision-recall curve for each of our models. We considered ROC because of the convex-hull nature of the metric where we can choose an optimal classifiers based on AUC.[8] We also utilized the F1 metric to measure the performance of our individual models instead of accuracy because the imbalanced nature of our data set makes the classifiers try to predict no positives to maximize accuracy. Weighted F-1 score is therefore used as it has been proven to be a cost sensitive metric that is suitable for imbalanced data sets.[6]

For performance evaluation we used a train-test split of 80/20 of our original data. For cross-validation purposes we generally used 3-5 folds depending on the computational intensity of the model.

Since the effectiveness of SMOTE on our modeling was of considerable concern to us we ran several other unbalanced analyses and found random forest to be the model that achieved the best results (See Table 7). As random forest would become one of the primary models for our analysis, we used this unbalanced model as a comparison point as well.

In identifying a baseline, as mentioned above, we considered an unbalanced Nave Bayes. It achieved an AUC of .79 and an F1 score of .86.

After identifying a benchmark, we moved forward with strategies to balance the dataset appropriately. We looked into and tested both SMOTE and ADASYN methods. Results were generally similar and slightly favored SMOTE so our

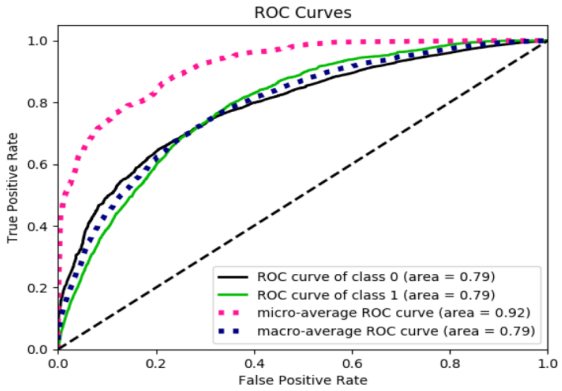

Figure 8: ROC Curve of Naive Bayes

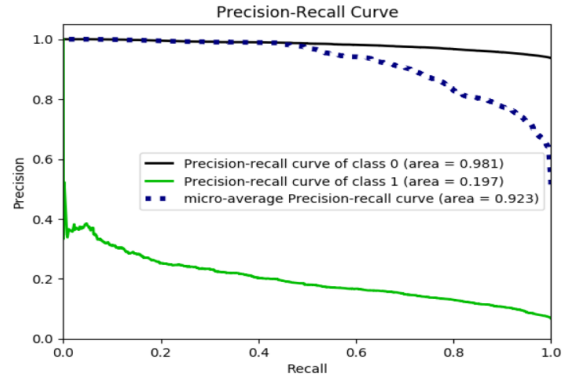

Figure 9: Precision-Recall Curve of Naive Bayers

		Predicted Label	
		0	1
True Label	0	114776	1211
	1	5692	2084

Table 3: Confusion Matrix of Random Forests

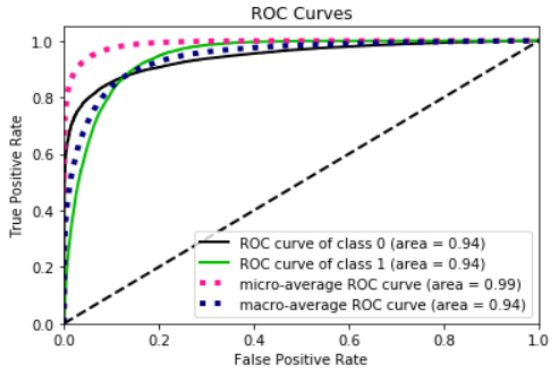

Figure 10: ROC Curve of Random Forests

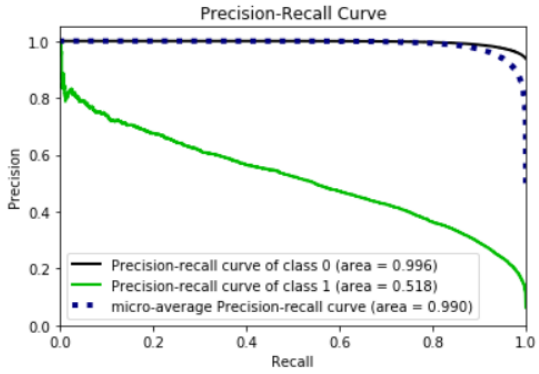

Figure 11: Precision-Recall Curve of Random Forests

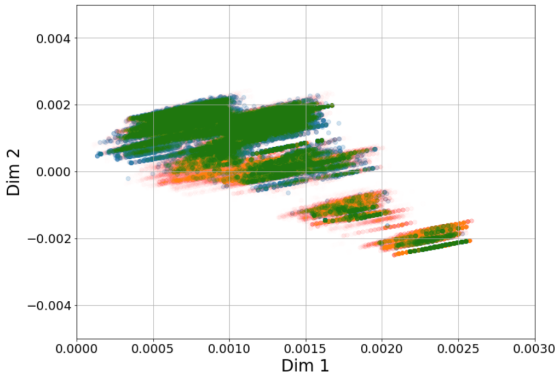

Figure 12: PCA Plot of Normal Data

final results have all been performed with SMOTE as the method for synthetic sampling.

SMOTE upsizes the train data from 495,050 up to 927,894. The special impact of the SMOTE algorithm can be seen in a side-by-side comparison of a 2D PCA run on the non-SMOTE and SMOTE data. As seen below, the minority class in blue is greatly upsized leading to considerably more dense overlap of the classes (causing the green color). The PCA analysis also makes it apparent that the data is not nearly separable in the 2D space which highlights the inherent difficulty in our task to identify true positives (People won't get funded).

We ran extensive cross-validation across many models (see results in Table 7), but ultimately found Random Forest and Neural Network based models to provide the best results. With our SMOTE random forest, we got the results seen in Figure 14 and Figure 15.

When we first looked at the confusion matrix, we were very happy with these results. It identified nearly 6,000 of the 8,000 expired loans, with false positives below 10,000. Compared to the non-SMOTE result, this looked much more like what we were trying to achieve with our primary task. However, when we analyzed the ROC and precision-recall

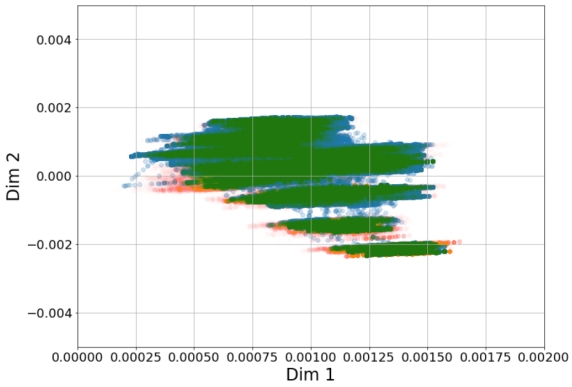

Figure 13: PCA Plot of Data after SMOTE

		Predicted Label	
		0	1
True Label	0	106523	9464
	1	1908	5868

Table 4: Confusion Matrix of Random Forests (SMOTE)

		Predicted Label	
		0	1
True Label	0	107008	8979
	1	1905	5871

Table 5: Confusion Matrix of Normal Random Forests with .175 Threshold

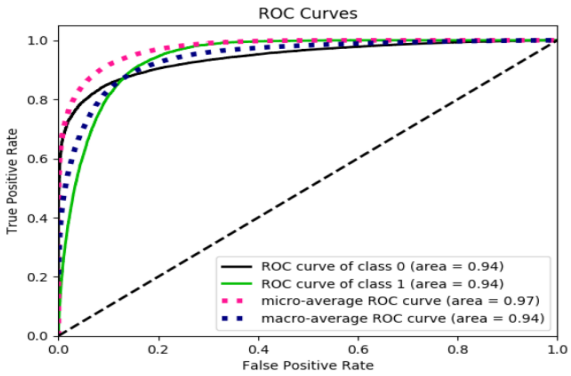

Figure 14: ROC Curve of Random Forests (SMOTE)

curves, it was clear that the two models were nearly identical in how they predicted.

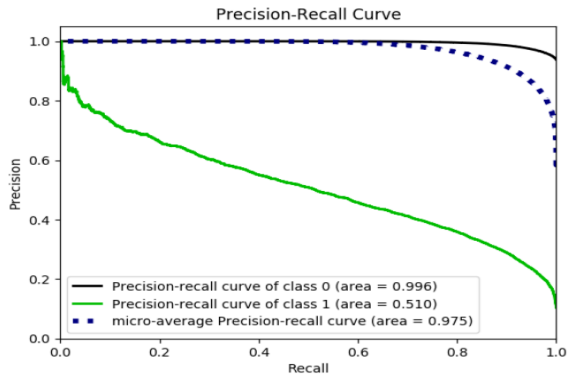

Figure 15: PR Curve of Random Forests (SMOTE)

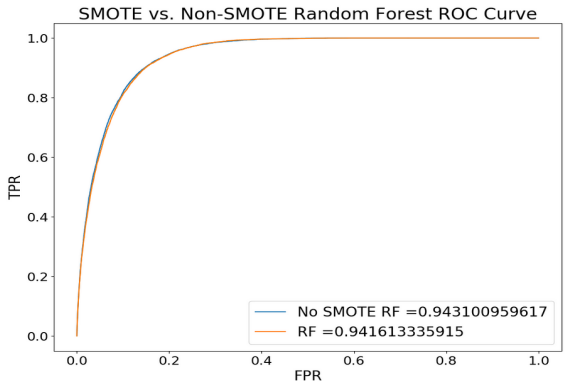

Figure 16: ROC Curve of SMOTE Data vs Normal Data

The SMOTE results merely shifted the threshold around. With the SMOTE results, the precision-recall tradeoff was more where we wanted to see it at the 50% threshold. The unbalanced data saw similar results with a threshold of 20% (see Table 5). Therefore, SMOTE is not achieving much beyond a simple probability threshold that can be easily manipulated on the user side.

When analyzing the features with highest importance in our model, we see in Figure 17 that the top features lean towards the numeric values. This was very interesting to observe since many of the numeric values had quite similar density curves for funded and expired loans. However, this is at the aggregate, so broken out more granularly as we can see in Figure 18, there must be significant variability in different segments of the population.

However, it makes sense that the numeric variables would be weighted highly since the tree can split on them $n-1$ times. Looking at the next ten variables, we can see some of the categorical fields that are making a big impact.

In addition to our random forest, neural nets preformed quite while on this data set. The scikit learn classifier provided the following results in Figure 20 and Figure 19.

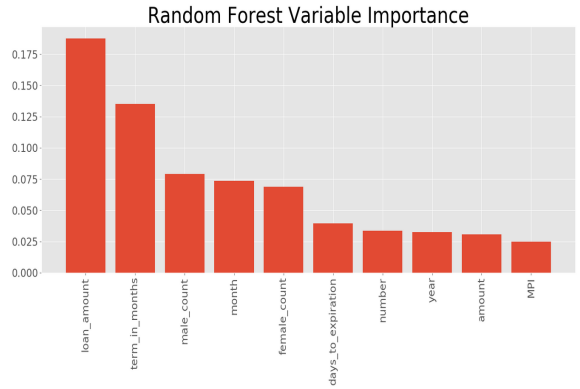

Figure 17: Random Forest Variable Importance

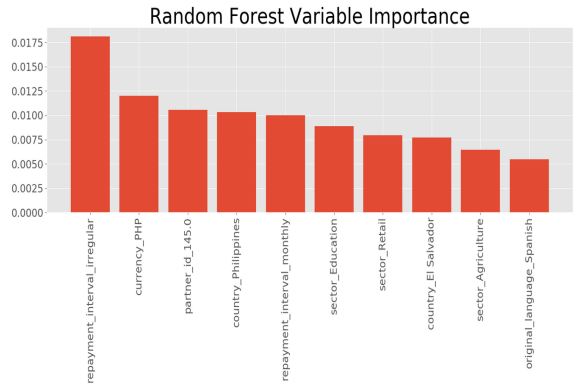

Figure 18: Random Forest Variable Importance

Predicted Label

True Label		0	1
	0	103429	12558
	1	1655	6121

Table 6: Confusion Matrix of MLP Classifier

During the analysis, we found it interesting how different models picked up very different components in the feature space to predict on. This made us curious if soft weighted mixtures of different models could meaningful benefit from each other. Meaningfully cross-validating this was prohibitively costly given the time constraints. However, as proof of concept, we took three models that had been appropriately cross validated (Neural Network, Logistic Regression and Random Forest) and plotted the F1 score and AUC of various weighted average combinations of the different models. Our final results can be seen in Table 7.

While we tried our best to grid search for our parameters for most of our models, we did face a computational

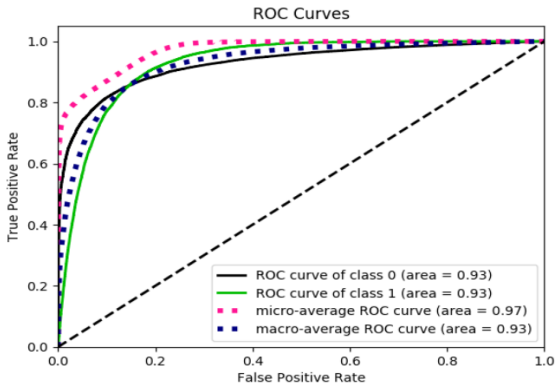

Figure 19: MLP Classifier ROC

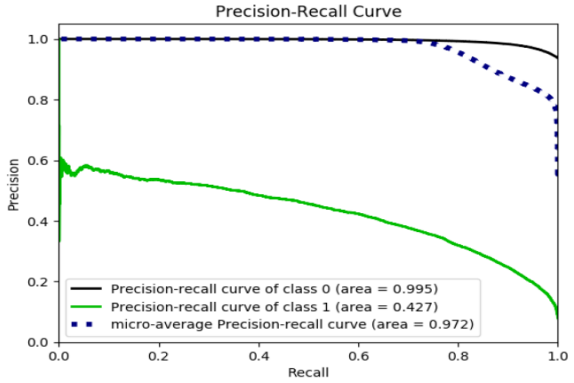

Figure 20: MLP Classifier PR Curve

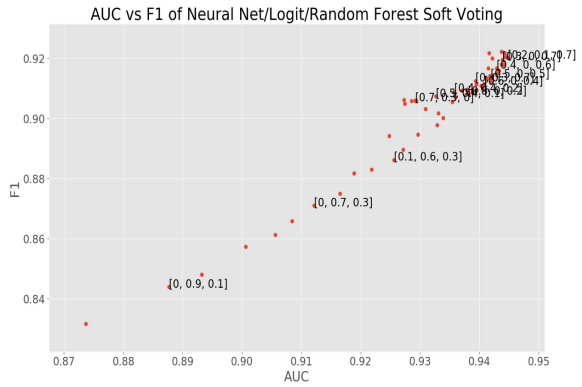

Figure 21: Ensemble Model Soft Voting Weights

hurdle in which for example Random Forest grid search for hyper parameter tuning took over 12 hours on an Amazon Web Services instance. As such we tried our best to get the best possible hyper parameters by doing randomized search instead of just grid searching. This has been known to approximate the best results of the grid search.[4]

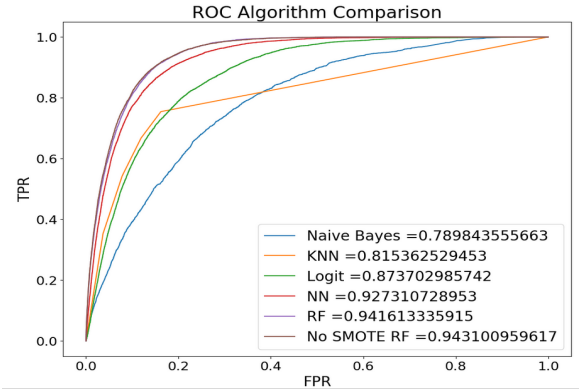

Figure 22: Algorithm Comparison by AUC ROC

5 CONCLUSION AND NEXT STEPS

Overall, we are happy with the results of the top performing models. A model such as this could be implemented within Kiva or by its member partners to meaningful identify loans likely to expire and respond appropriately.

Additionally, this analysis highlighted a couple key takeaways:

(1) SMOTE issues with categorical feature space. Overall, SMOTE failed to have a meaningful impact on most of the models we tested. We suspect that this is due to having so many categorical variables. Since SMOTE creates synthetic points via a nearest neighbors style imputation, it loses value when the feature space has high dimensionality.[5]

(2) The importance of threshold metrics in identifying similarity and variation in model results. A confusion matrix is easy and intuitive to read but may not be telling the whole story. Considering threshold metrics such as ROC AUC and precision-recall can help identify when models are similar but predict to a different threshold. This can be extremely valuable for model comparison.

(3) Difficulties of text generation when text is not independent (filtered through funneled translation mechanisms). While there has been some research into generative text models, it has been mostly focused on Natural Language Processing and relies on a trained model.

As next steps we would consider the following:

(1) Perform a more thorough soft voting analysis with cross-validation and deeper dive into what are the similarities and differences in the feature space that each model is picking up on.

(2) Continue to try to build out the text analysis piece with some more complex generative text models and n-grams. The hope here would be that even if we cannot make this model as robust as the feature model, we could extract something of value from the text that could flow into the feature model.

Table 7: Final Cross-validated Results

Model	Hyperparamters	F-1 Score	ROC-AUC	PR AUC
Naive Bayes	$\alpha = 0.1$	0.76	0.78	0.20
kNN	N = 4	0.91	0.78	0.27
l_2 Logistic Regression	C = 1000	0.83	0.86	0.27
Random Forests	Max Depth = 36, No of Estimators = 193	0.93	0.94	0.52
Random Forests (SMOTE)	Max Depth = 50, No of Estimators = 200	0.92	0.94	0.51
MLP Classifier	Learning Rate = Adaptive, Layer Sizes = (100, 50, 20), Activation = Relu	0.93	0.93	0.43

(3) Spend a little more time on interpretability. It was not the main focus of our task with this project, but from both a business sense and a new feature selection sense, this could help a great deal in communicating and improving results.
(4) Spend more time trying to tie in meaningful geographic characteristics. When we started, we saw this something that would really help the feature selection. However, we tried to match region names with a global index of millions of cities and regions with little success. [2] The messiness of the 12,000+ individual regions that are identified in the loan data made it hard to make good use of given our time constraints.

REFERENCES

[1] [n. d.]. http://ophi.org.uk/multidimensional-poverty-index/global-mpi-2017/mpi-data/
[2] [n. d.]. Free World Cities Database. https://www.maxmind.com/en/free-world-cities-database
[3] [n. d.]. Kiva API. https://build.kiva.org/
[4] James Bergstra and Yoshua Bengio. 2012. Random Search for Hyper-parameter Optimization. *J. Mach. Learn. Res.* 13, 1 (Feb. 2012), 281–305.
[5] Rok Blagus and Lara Lusa. 2013. SMOTE for high-dimensional class-imbalanced data. *BMC Bioinformatics* 14, 1 (22 Mar 2013), 106. https://doi.org/10.1186/1471-2105-14-106
[6] Nitesh V. Chawla. 2005. Data Mining for Imbalanced Datasets: An Overview. In *The Data Mining and Knowledge Discovery Handbook*, Oded Maimon and Lior Rokach (Eds.). Springer, 853–867. http://dblp.uni-trier.de/db/books/collections/datamining2005.html#Chawla05
[7] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Oversampling Technique. *J. Artif. Int. Res.* 16, 1 (Jun 2002), 321–357.
[8] Tom Fawcett. 2006. An Introduction to ROC Analysis. *Pattern Recogn. Lett.* 27, 8 (June 2006), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010
[9] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera. 2012. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42, 4 (July 2012), 463–484. https://doi.org/10.1109/TSMCC.2011.2161285
[10] Haibo He, Yang Bai, Edwardo A. Garcia, and Chengchao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (2008), 1322–1328.
[11] Kiva. 2018. Data Science for Good: Kiva Crowdfunding — Kaggle. https://www.kaggle.com/kiva/data-science-for-good-kiva-crowdfunding
[12] Terry Waghorn. 2013. Premal Shah: Loans That Change Lives. https://www.forbes.com/sites/terrywaghorn/2013/11/04/premal-shah-loans-that-change-lives/#750b949a2b07