

**Project Brief: Kiva Loans**

**Dr. Stephen Thomas**

**January 20, 2022**

# Introduction

You will use Python to build a machine learning model to predict whether a Kiva loan request will default. You will describe their modeling journey and outline which techniques worked and which techniques did not work for you.

# The Case

Kiva Microfunds is a non-profit that allows individuals to lend money to low-income entrepreneurs and students around the world. Since starting in 2005, Kiva has crowd-funded millions of loans with a repayment rate of around 98%.

At Kiva, each loan request includes both traditional demographic information on the borrower, such as gender and location, as well as a personal story, because Kiva wants lenders to connect with borrowers on a human level. For example, consider the personal story for a borrower named Evelyn:

*Evelyn is 40 years old and married with 3 kids. She is in the Karura Hope women group and her life has been changed by the first KIVA loan she received last year which she is completing this quarter. Before she received the loan, she used to sell 9 litres of milk daily to local residents. After receiving the loan she bought iron sheets, five cement packets, one lorry of sand, some ballast and animal feed for her cows and improved her cow shed. Today she sells a daily average of 40 litres of milk to the Kiamba Dairy cooperative society, which is affiliated to the Kenya Cooperative Creameries at a cost of USD 0.28 per litre. Her daily farming has really grown. Evelyn intends to buy another dairy cow and a tank of water for home consumption and for her cows. She intends to repay in monthly installments.*

Despite her uplifting story, and her previous successful loan, Evelyn defaulted on her next loan of 900 USD. Her lenders lost their money, and Evelyn is no longer allowed to use Kiva.

You are tasked to predict this unfortunate default, and others like it. If defaults can be predicted beforehand, then the risk to potential lenders is reduced, and loan resources are given to borrowers with the highest likelihood of repayment.

You can find further details of this case by reading the following resources:

* [Hacking the Language of Loan Defaults](https://smith.queensu.ca/insight/content/hacking_the_language_of-loan-defaults.php). Smith Insights.

# The Data

Kiva makes the history of all previous loan requests and associated information publicly available, including the borrower's demographic data, the industry sector of the borrower, the requested loan amount, and whether the loan eventually defaulted.

For this project, I have extracted a labeled training dataset of 6,138 completed loans, of which 49% were makred as “defaulted”. I have removed all loan information (demographics, industry, loan amount, etc.) except the textual loan description and the target.

I have also extracted a testing dataset of 664 additional loan requests in which the target will not be provided to you.

The data can be found at the following links:

* [kiva\_train.csv](https://drive.google.com/file/d/1dzzVbgHphbCf7kvq9IKiIhwzmxPbuH4s/view?usp=sharing)
* [kiva\_test.csv](https://drive.google.com/file/d/1EVWfyqQOd_W2uTKrr4JTD2iFrEZHoOHT/view?usp=sharing)

# Your Mission

Using the labeled training data (*kiva\_train.csv*), you are to build a model to predict whether a loan request will default (i.e. *defaulted=1*) given the description (i.e., *en\_clean)* of the loan request.

You will use your model to predict default of the held-out test set (*kiva\_test.csv)*. You will submit your predictions via email to [stephen.thomas@queensu.ca](mailto:stephen.thomas@queensu.ca) to see how well you did. Your predictions will be measured using Macro F1-Score.

You are to share your code (for example, using the [Google Colab](https://colab.research.google.com/) platform).

**Rules**

* You may not use any other data sources/sets, other than what is provided in *kiva\_train.csv*. While it is possible to find more information about each of the loan requests in question (e.g., from build.kiva.org, or from other Kiva loan data dumps that you may find on Github or other), please do not do so.
* You may use any preprocessing and cleaning steps you deem appropriate.
* Your predictions must be fully automated, i.e., no humans-in-the-loop.
* You must use machine learning. i.e., no hand-crafted rules.
* You may use any Python package to build your ML model, including (but not limited to) scikit-learn, xgboost, spaCy, textblob, Gensim, Pytext, Transformers (from Huggingface), Flair, Allen NLP, and fastText.
* You must try at least one shallow ML model (e.g., TF-IDF and XGBoost) and one pre-trained LM (e.g., BERT, GPT-J, T5).

**Presentation**

Please create a 15-minute presentation that outlines your modeling journey. The presentation should outline the various techniques (cleaning, preprocessing, feature engineering, ML algorithm types) that were attempted, and whether they helped or did not help the overall model performance.

The presentation should target an audience of technically-savvy professionals and data scientists (i.e., Management Analytics and CS professors) with the primary goal of conveying what you learned during the competition process. It is OK to assume the audience is familiar with terms and ideas learned in this program (e.g., embedding, RNN, XGBoost, bag of words, stop words, etc.).

**Deliverables**

* A file that contains the predictions of the held-out test set
* A link to (or copy of) your coding solution
* A 15-minute presentation as described above