Policy Analysis with NLP

WORLD
RESOURCES
INSTITUTE

1.1 Overview

1. Executive Summary
2. Results
3. Recommendation and Benefits
4. Conclusion

1. Executive Summary

WRI Project Goal

To better identify financing availability in forest landscape restoration policy

Challenges

- Better classify different types of text documents which begin with manual labelling rules
- Apply the labelling models to a more global scale
- Identify more dimensions of financing mechanisms using labeling models

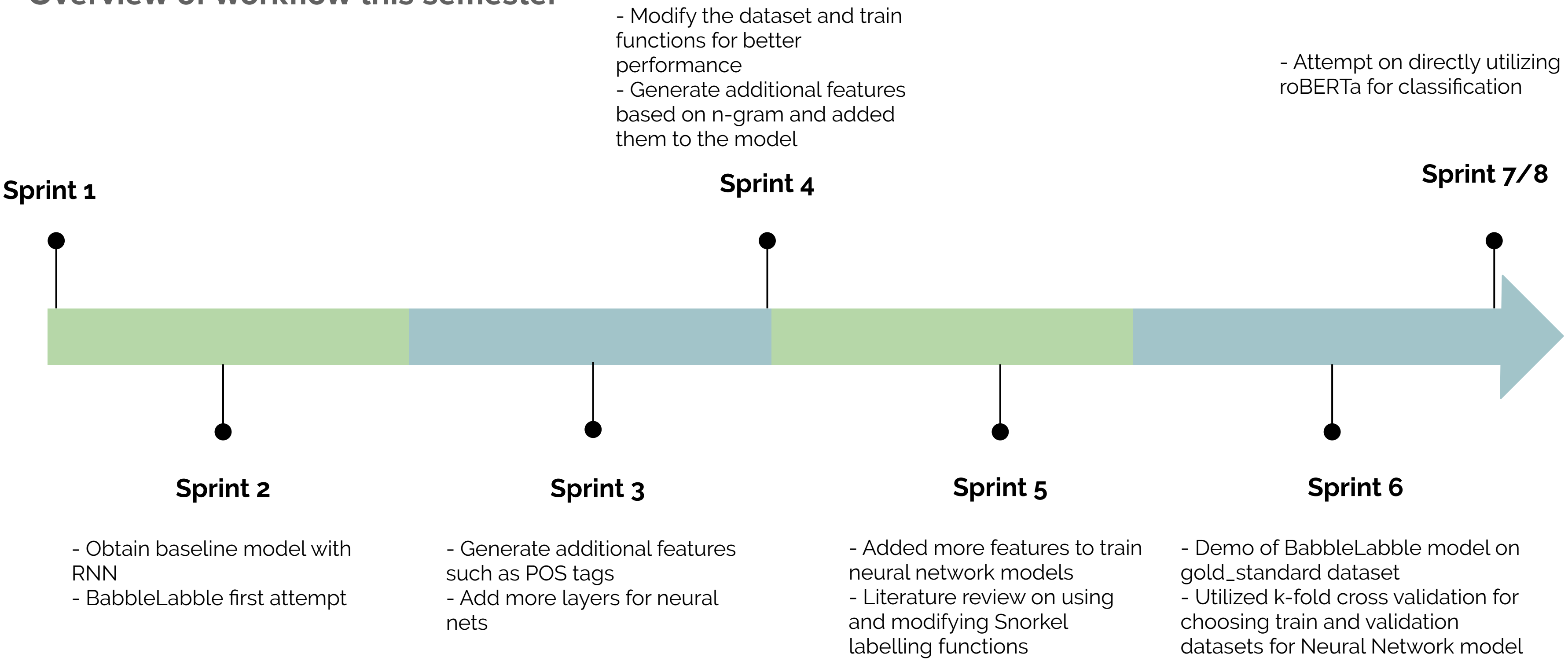
Research Questions

- How to better classify the financing incentive of relevant policies?
- How to maximize the value of current models and apply them to identify other characteristics of each financing mechanism?

Type **Audience** **Financier** **Stipulations** **Amount**

1. Executive Summary

Overview of workflow this semester



Sprint 1

Sprint 2

- Obtain baseline model with RNN
- BabbleLabble first attempt

Sprint 3

- Generate additional features such as POS tags
- Add more layers for neural nets

Sprint 4

- Modify the dataset and train functions for better performance
- Generate additional features based on n-gram and added them to the model

Sprint 5

- Added more features to train neural network models
- Literature review on using and modifying Snorkel labelling functions

Sprint 6

- Demo of BabbleLabble model on gold_standard dataset
- Utilized k-fold cross validation for choosing train and validation datasets for Neural Network model

Sprint 7/8

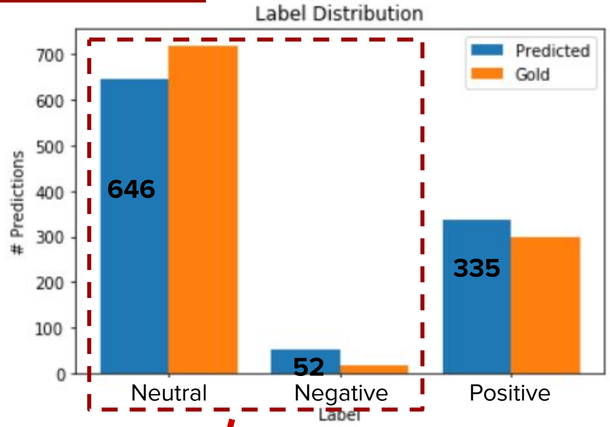
- Attempt on directly utilizing roBERTa for classification

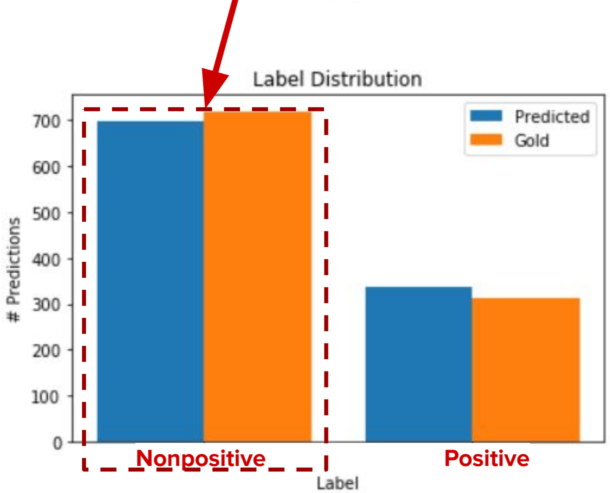
2. Results Overview

1. Data Source and Preparation
2. Labelling Model Optimization - Snorkel
3. Labelling Model Optimization - BabbleLabble
4. Feature Engineering
    a. Topic Modeling
    b. Sentiment Score and NER
    c. Summary
5. Neural Network Model

2.1 Data Source and Preparation

Gold_standard data



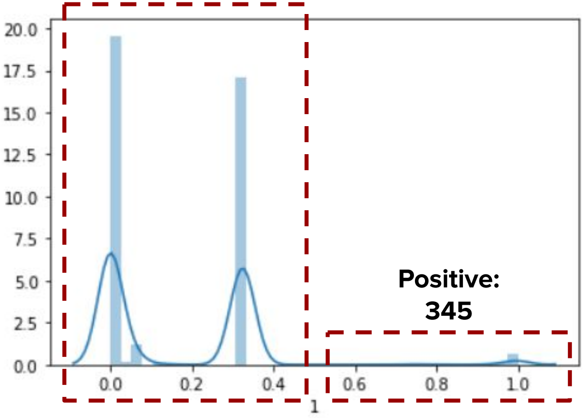


Noisy data

After converting the original gold_standard dataset to the new one with two classes, we redid the snorkel prediction, and the distribution of the noisy_proba is shown in the following plot.

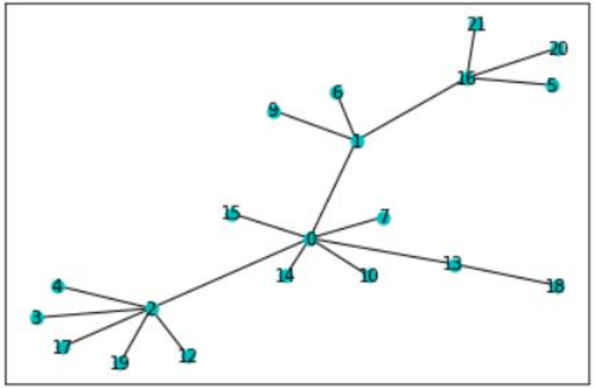
We used 0.5 as threshold to classify the results into ‘nonpositive’ level and ‘positive’ level.

Nonpositive: 10508



Positive: 345

Raw data -- policy documents



Filename of each documents:

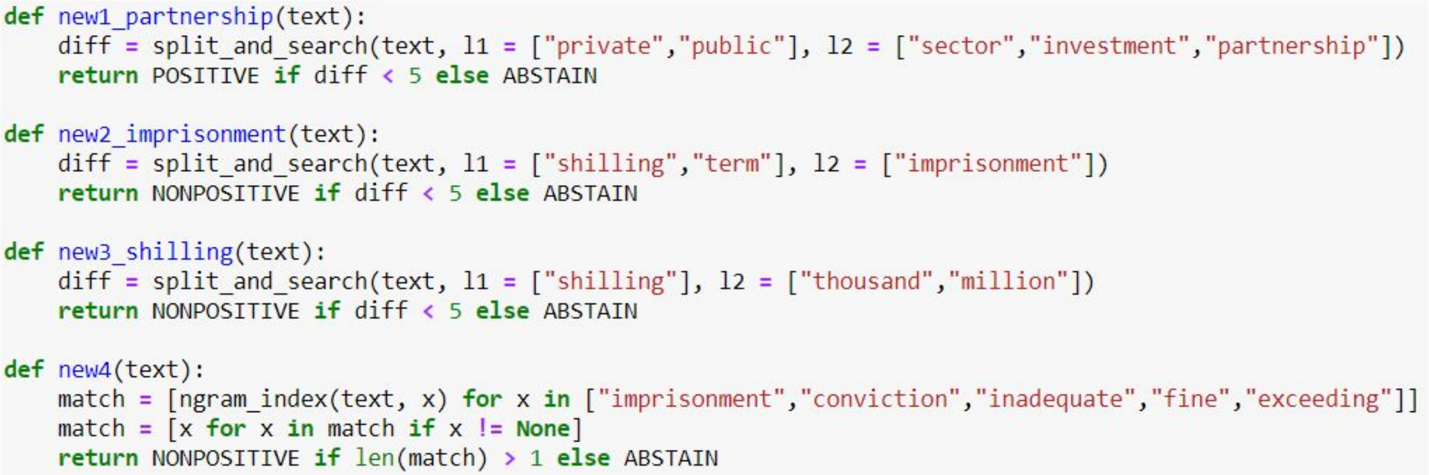
0: "Constituion.txt"
1: "Environment Management Act.txt"
2: "Forest Act – 1997.txt"
3: "Forest Landscape Restoration Strategy.txt"
4: "Malawi Government National Forest Policy - June 2016.txt"
5: "Malawi Growth and Development Strategy III .txt"

2.2 Labelling Model Optimization - Snorkel

- We looked at the unigrams, bigrams, and trigrams generated from the gold standard dataset and then created four additional label functions based on the most frequent ngram words.

- Before we added these functions to Snorkel, we firstly checked if they could significantly classify data in different classes. For example, the function new1_partnership could help to classify 14% samples in 'positive' class, the function new2_imprisonment could help to classify 83% samples in 'negative' class, and the function new3_shilling could help to classify 77% samples in 'negative' class.

- As a result, the labeling accuracy was increased by 0.024 and F1 score was improved by 0.081.



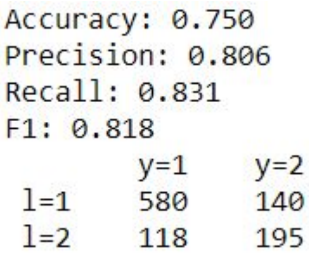
```python
def new1_partnership(text):
    diff = split_and_search(text, l1 = ["private","public"], l2 = ["sector","investment","partnership"])
    return POSITIVE if diff < 5 else ABSTAIN

def new2_imprisonment(text):
    diff = split_and_search(text, l1 = ["shilling","term"], l2 = ["imprisonment"])
    return NONPOSITIVE if diff < 5 else ABSTAIN

def new3_shilling(text):
    diff = split_and_search(text, l1 = ["shilling"], l2 = ["thousand","million"])
    return NONPOSITIVE if diff < 5 else ABSTAIN

def new4(text):
    match = [ngram_index(text, x) for x in ["imprisonment","conviction","inadequate","fine","exceeding"]]
    match = [x for x in match if x != None]
    return NONPOSITIVE if len(match) > 1 else ABSTAIN
```

Latest Snorkel Results:



```
Accuracy: 0.750
Precision: 0.806
Recall: 0.831
F1: 0.818
           y=1    y=2
 l=1       580    140
 l=2       118    195
```

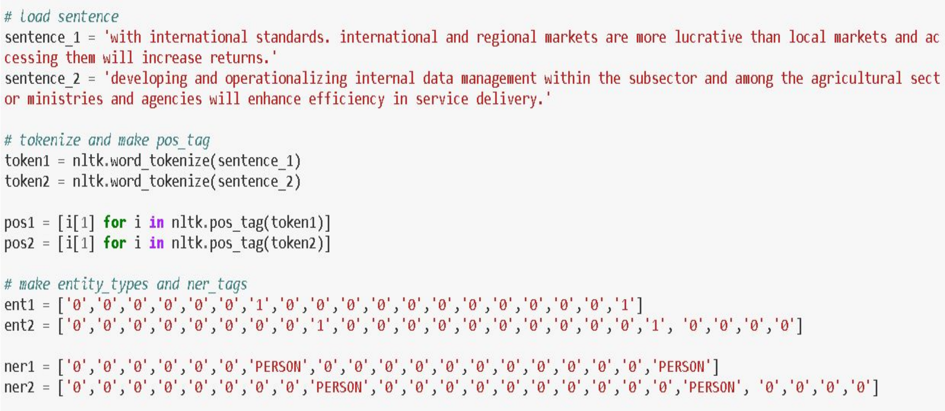
2.3 Labelling Model Optimization - Babble Labble

Challenges:

- Data structure does not match Babble Labble requirements (RelationMention)
- Create RelationMention with matching attributes

Applications:

- Implement two sample sentences from golden standard to pre-process to check the feasibility of Babble
- Reverse-engineer the data pre-processing procedure which is not provided by Babble
- Manually add required attributes to construct the relationMention structure
- Successfully generate labeling functions with explanations provided



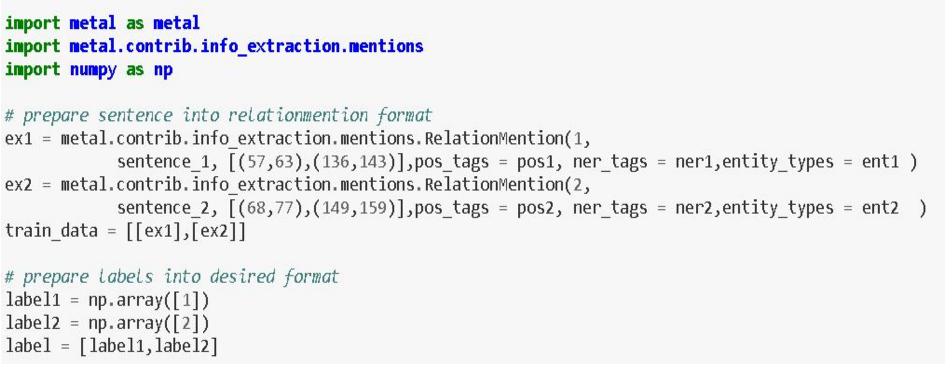
```
# load sentence
sentence_1 = 'with international standards. international and regional markets are more lucrative than local markets and ac
cessing them will increase returns.'
sentence_2 = 'developing and operationalizing internal data management within the subsector and among the agricultural sect
or ministries and agencies will enhance efficiency in service delivery.'

# tokenize and make pos_tag
token1 = nltk.word_tokenize(sentence_1)
token2 = nltk.word_tokenize(sentence_2)

pos1 = [i[1] for i in nltk.pos_tag(token1)]
pos2 = [i[1] for i in nltk.pos_tag(token2)]

# make entity_types and ner_tags
ent1 = ['0','0','0','0','0','0','1','0','0','0','0','0','0','0','0','0','0','0','1']
ent2 = ['0','0','0','0','0','0','0','0','1','0','0','0','0','0','0','0','0','0','0','1', '0','0','0','0']

ner1 = ['0','0','0','0','0','0','PERSON','0','0','0','0','0','0','0','0','0','0','0','PERSON']
ner2 = ['0','0','0','0','0','0','0','0','PERSON','0','0','0','0','0','0','0','0','0','0','PERSON', '0','0','0','0']
```

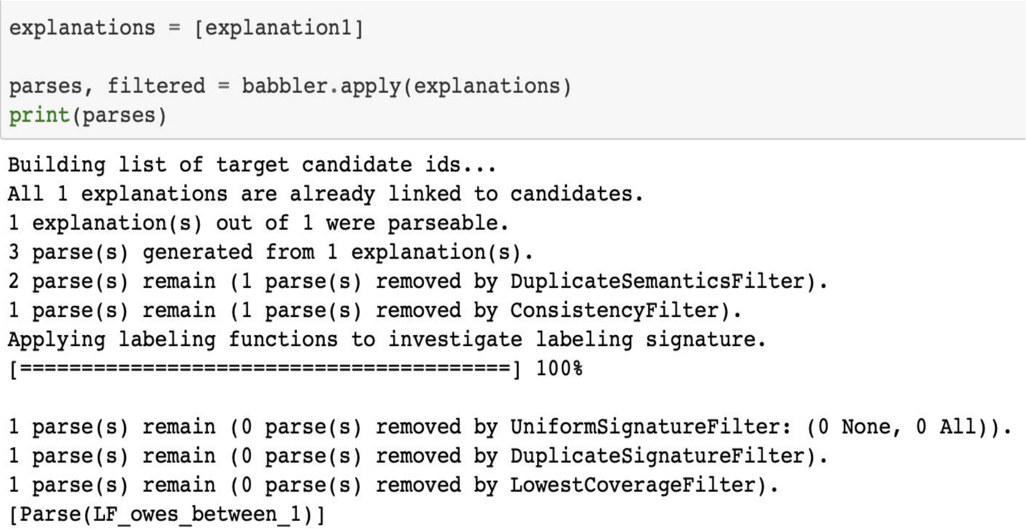


```
import metal as metal
import metal.contrib.info_extraction.mentions
import numpy as np

# prepare sentence into relationmention format
ex1 = metal.contrib.info_extraction.mentions.RelationMention(1,
            sentence_1, [(57,63),(136,143)],pos_tags = pos1, ner_tags = ner1,entity_types = ent1 )
ex2 = metal.contrib.info_extraction.mentions.RelationMention(2,
            sentence_2, [(68,77),(149,159)],pos_tags = pos2, ner_tags = ner2,entity_types = ent2  )
train_data = [[ex1],[ex2]]

# prepare labels into desired format
label1 = np.array([1])
label2 = np.array([2])
label = [label1,label2]
```

2.3 Labelling Model Optimization - Babble Labble



```
explanations = [explanation1]

parses, filtered = babbler.apply(explanations)
print(parses)
```

```
Building list of target candidate ids...
All 1 explanations are already linked to candidates.
1 explanation(s) out of 1 were parseable.
3 parse(s) generated from 1 explanation(s).
2 parse(s) remain (1 parse(s) removed by DuplicateSemanticsFilter).
1 parse(s) remain (1 parse(s) removed by ConsistencyFilter).
Applying labeling functions to investigate labeling signature.
[========================================] 100%

1 parse(s) remain (0 parse(s) removed by UniformSignatureFilter: (0 None, 0 All)).
1 parse(s) remain (0 parse(s) removed by DuplicateSignatureFilter).
1 parse(s) remain (0 parse(s) removed by LowestCoverageFilter).
[Parse(LF_owes_between_1)]
```

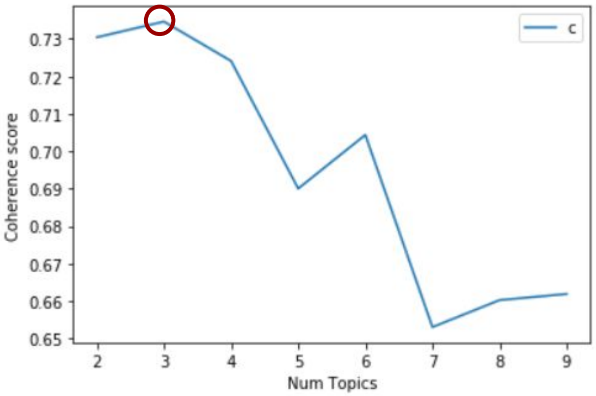
Limitation

- Sometimes explanations would be filtered out with no LF created
- Data must be in RelationMention structure, need extra data transformation process

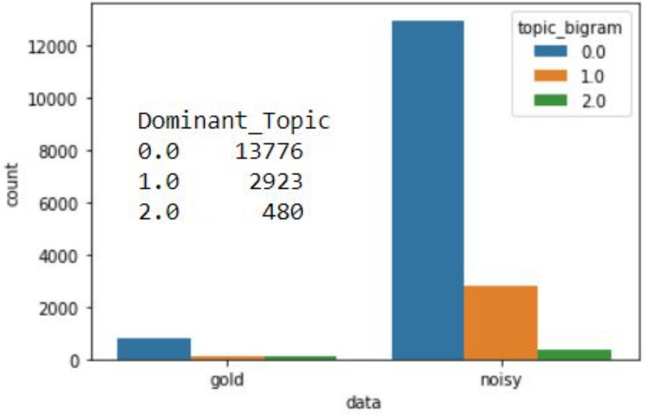
Recommendation

- Automate data transformation pipeline to construct RelationMention structure
- Write various explanations with natural language to create better training datasets and test the results
- Further Research Babble Labble codes, improve the embedded functions to better recognize the explanations

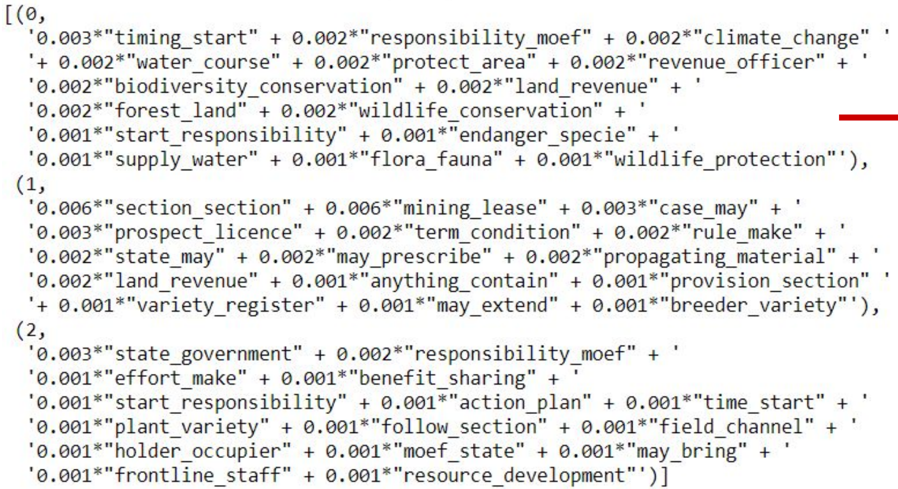
2.4 Feature Engineering -- Topic modeling



For easy understanding, we do **LDA with bigram**



```
Dominant_Topic
0.0     13776
1.0      2923
2.0       480
```



```
[(0,
  '0.003*"timing_start" + 0.002*"responsibility_moef" + 0.002*"climate_change" '
  '+ 0.002*"water_course" + 0.002*"protect_area" + 0.002*"revenue_officer" + '
  '0.002*"biodiversity_conservation" + 0.002*"land_revenue" + '
  '0.002*"forest_land" + 0.002*"wildlife_conservation" + '
  '0.001*"start_responsibility" + 0.001*"endanger_specie" + '
  '0.001*"supply_water" + 0.001*"flora_fauna" + 0.001*"wildlife_protection"'),
 (1,
  '0.006*"section_section" + 0.006*"mining_lease" + 0.003*"case_may" + '
  '0.003*"prospect_licence" + 0.002*"term_condition" + 0.002*"rule_make" + '
  '0.002*"state_may" + 0.002*"may_prescribe" + 0.002*"propagating_material" + '
  '0.002*"land_revenue" + 0.001*"anything_contain" + 0.001*"provision_section" '
  '+ 0.001*"variety_register" + 0.001*"may_extend" + 0.001*"breeder_variety"'),
 (2,
  '0.003*"state_government" + 0.002*"responsibility_moef" + '
  '0.001*"effort_make" + 0.001*"benefit_sharing" + '
  '0.001*"start_responsibility" + 0.001*"action_plan" + 0.001*"time_start" + '
  '0.001*"plant_variety" + 0.001*"follow_section" + 0.001*"field_channel" + '
  '0.001*"holder_occupier" + 0.001*"moef_state" + 0.001*"may_bring" + '
  '0.001*"frontline_staff" + 0.001*"resource_development"')]
```

The most representative sentence in topic0:

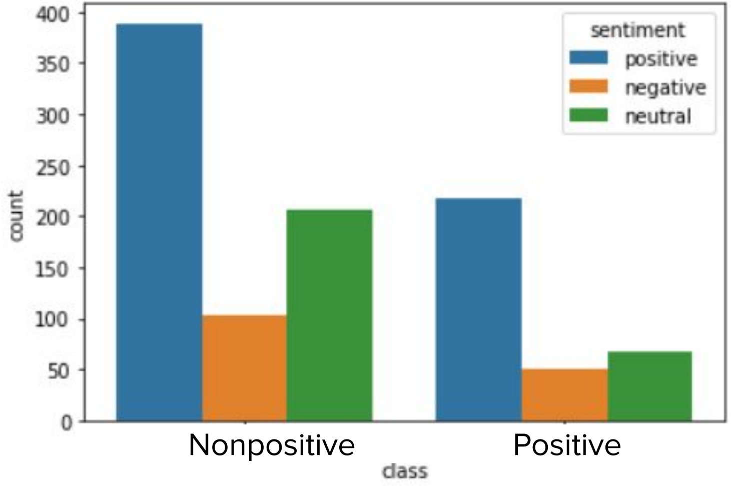
'health care subsidies for social health protection this will be achieved through the following strategies consolidating, expanding new and existing and coordinating social health subsidy mechanism for the poor with a view of achieving universal coverage;provision of free maternity services in all public health facilities;expanding coverage of health benefits to all the indigents;establishing a national social health insurance mechanism that caters for employees, employers and the informal sector with a view to gain universal coverage;reforming national hospital insurance fund naif to effectively act as a vehicle to implement the national health insurance scheme;designing a harmonized and progressive resource mobilization strategies targeting all sources of funds, both domestic and international;strengthening programming of external funding of health through improved harmonization and alignment to sector priorities and improved reporting;ensuring efficient allocation and utilization of resources; and progressively eliminating payment at the point of use of health services, especially by the marginalized. re engineering human resource for health to realize achievements in this project......

2.4 Feature Engineering -- Sentiment Score and NER

Sentiment Analysis:

We used TextBlob package, which is a high level library built over top of NLTK library and trained data on a Naive Bayes Classifier, to label our data.
In the 'positive' class, the number of sentences with positive sentiment(blue) is apparently larger than the number of sentences with other sentiments(orange + green).

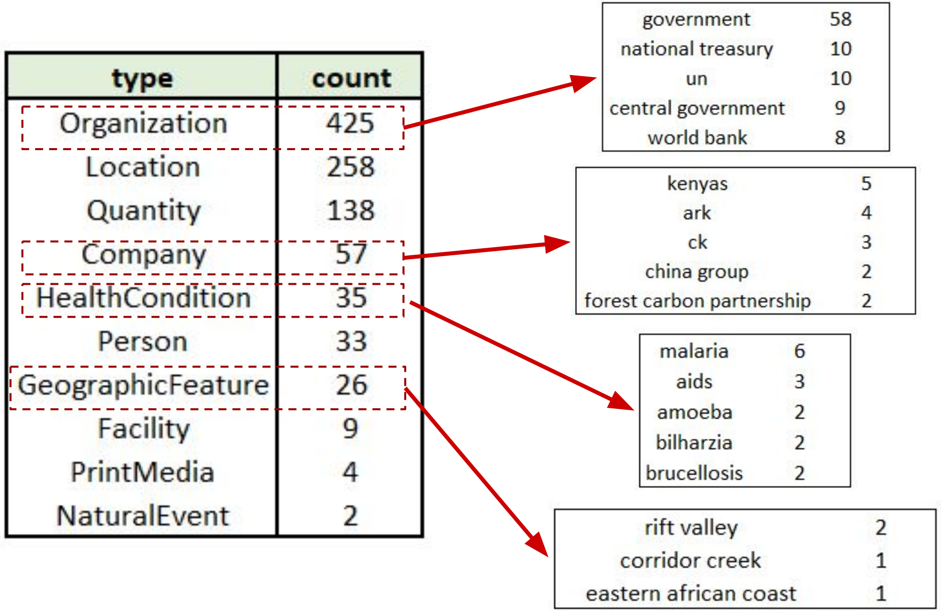
Sentiment Level Distribution on gold_standard data



Named-Entity Recognition:

We used IBM Natural Language Understanding API to extract named entity in the gold-standard data and noisy data.

“NER_results_gold_file.csv”
“NER_results_noisy_file.csv”



type	count
Organization	425
Location	258
Quantity	138
Company	57
HealthCondition	35
Person	33
GeographicFeature	26
Facility	9
PrintMedia	4
NaturalEvent	2

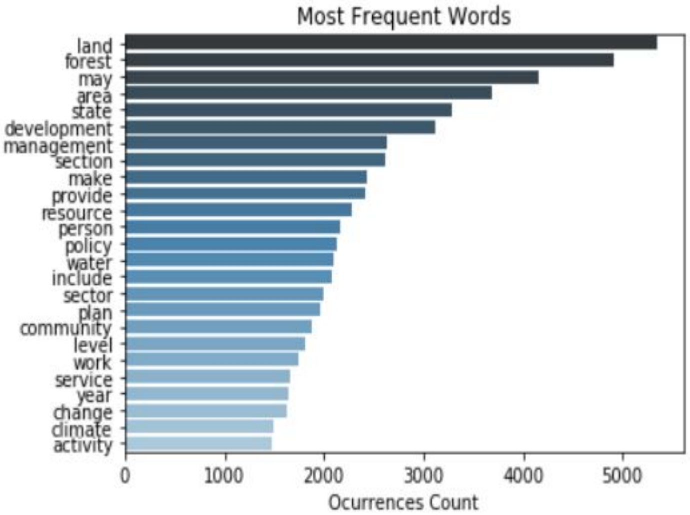
government	58
national treasury	10
un	10
central government	9
world bank	8

kenyas	5
ark	4
ck	3
china group	2
forest carbon partnership	2

malaria	6
aids	3
amoeba	2
bilharzia	2
brucellosis	2

rift valley	2
corridor creek	1
eastern african coast	1

2.4 Feature Engineering -- Summary



N-Gram: we added a unigram feature by counting whether the sentence include a certain unigram and selected the top 200 features to use.

POS tags: after inspecting the most common tags in the dataset, we decided to add the counts of POS tags 'NN', 'IN', 'JJ', 'NNS', 'DT', 'CC', 'CD', 'VB' as the new POS tag features

X0:

- TfidfVectorizer

X_base:

- results of topic modeling
- sentiment score
- number of entities

X1:

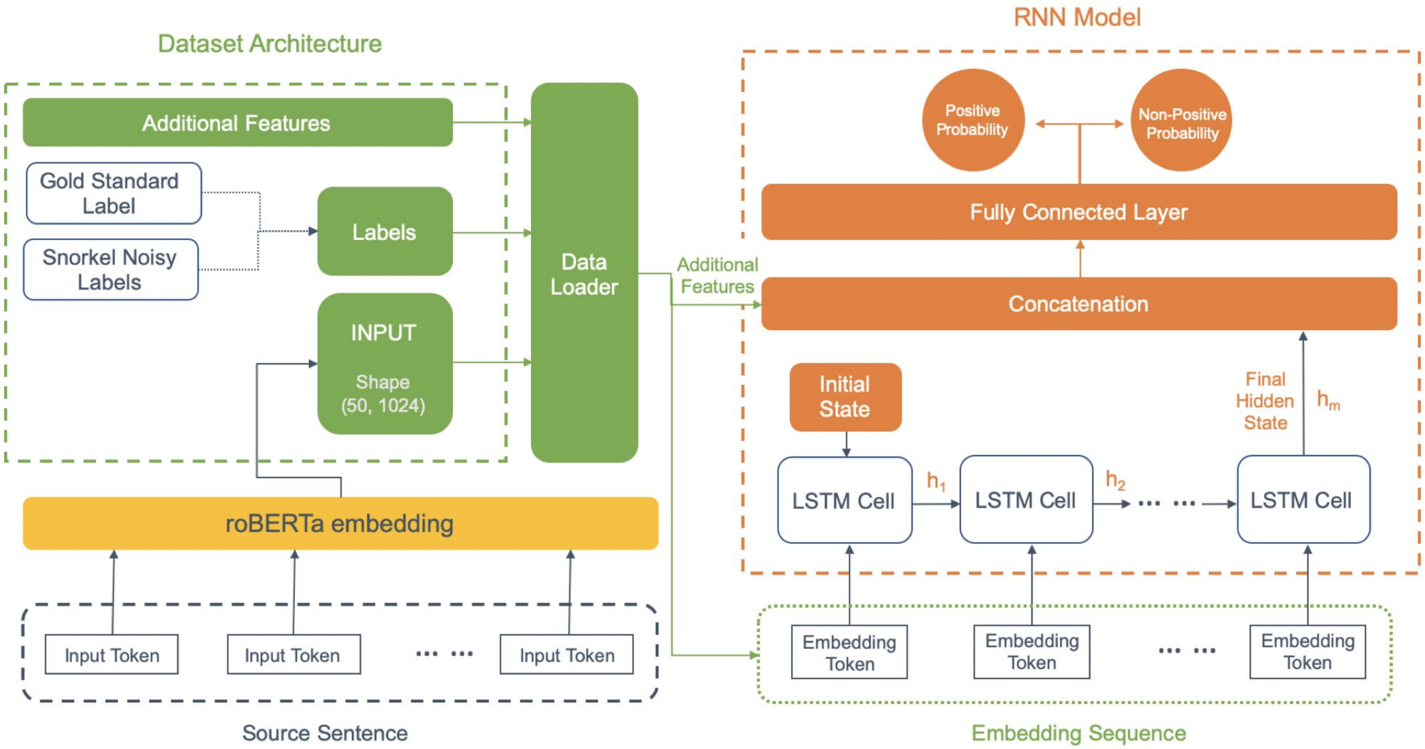
- ngram(unigram, bigram and trigram)

X2(pos tag & others):

- total number of words in the documents
- total number of characters in the documents
- total number of punctuation marks in the documents
- total number of upper count words in the documents
- total number of proper case (title) words in the documents
- Frequency distribution of Part of Speech Tags

- ***Plus: One Hot Encoding & Normalization***

2.5 Neural Network Model



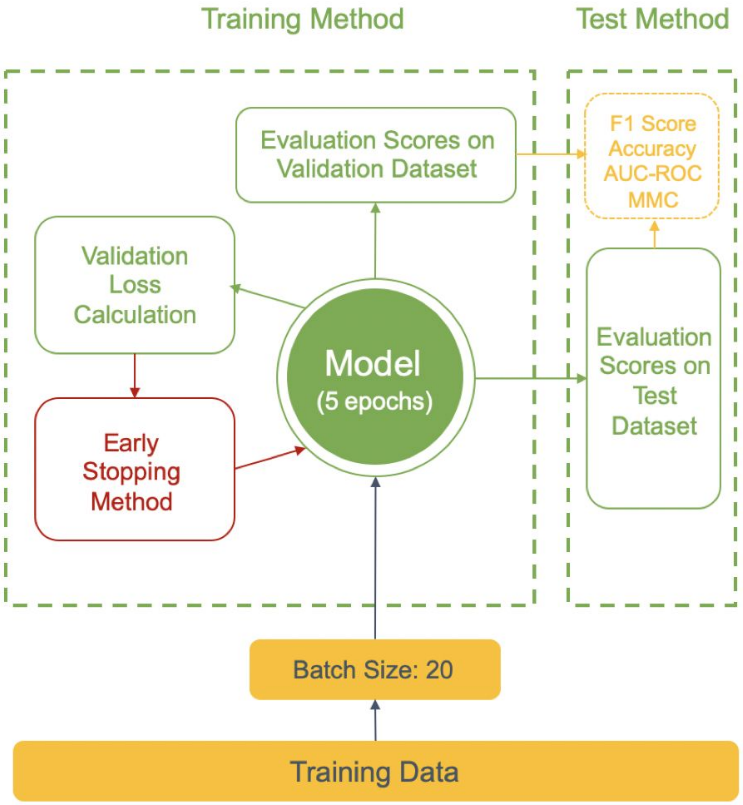
1) Model Architecture

Modified the structure of dataset and RNN model by integrating new variable for features and concatenation functions in order to incorporate additional features into the roBERTa embedding results.

2) RNN Layers

Increased the depth of LSTM layers and also tried out GRU layers but the latter didn't improve model performance

2.5 Neural Network Model



3) Training Method

- Reduced the **batch size** from 50 to 20
- Add an **early stopping** method to stop the iteration if the validation loss stops improving for 3 consecutive times since we only train the model for 5 epochs for now
- Tried **k-fold cross validation** but it didn't improve model performance
- Evaluate the final model on the **validation** dataset when using k-fold cross validation

4) Test Method

Add a test method to evaluate the model on the test dataset. Evaluation scores include:

- F1 Score
- Accuracy
- AUC-ROC Score
- **Matthews Correlation Coefficient** (good measure for unbalanced dataset)

2.5 Neural Network Model

Metrics/ Models	Baseline Model	Noisy Model	N-gram+POS+ Sentiment (batch 50)	N-gram+POS+ Sentiment (batch 20)	**N-gram+POS+ Sentiment+Topic (batch 20)**	N-gram+POS+ Sentiment+Topic+ Counts (batch 20)
F1 Score	0.509	0.4	0.383	0.488	**0.516**	0.510
Accuracy	0.58	0.63	0.62	0.644	**0.665**	0.651
ROC AUC Score	0.525	0.51	0.495	0.513	**0.536**	0.527
Matthews Correlation Coefficient		0.0165	-0.049	0.036	**0.102**	0.073

Results

- Our current model has showed an improved performance on all four metrics compared with the original noisy model and the baseline model
- The optimal model uses the following additional features (all the features are scaled):
  - N-Grams
  - POS Tags: the count of noun, preposition, adjective, determiner, coordinating conjunction, cardinal number, and verb
  - Sentiment Scores
  - Topic Class: generated using topic modeling

2. Results

Limitation

- **Unbalanced dataset**: around 98% of the noisy data are non-positive and 2% of them are positive
- Too few **label functions** for identifying positive information in the Snorkel model
- Due to the **computation and time limit**, the RNN model has only been trained on 5 epochs and we aren’t able to spend more time on exploring more complicated neural network architectures

Next Steps

- **Dataset & Snorkel**: Add more labeling functions and rules in the Snorkel model to identify positive labels based on the logic behind the human evaluation of gold standard data
- **BabbleLabble**: Feasible but need to implement data transformations for further modeling and develop new pipeline
- **Neural Nets**:
  - Use the MCC score for evaluation
  - Train the model on more epochs
  - Explore more complicated RNN
  - Optimize roBERTa by training it on policy-related documents
  - **Another possible alternative**: Text classification directly using roBERTa

3. Recommendation

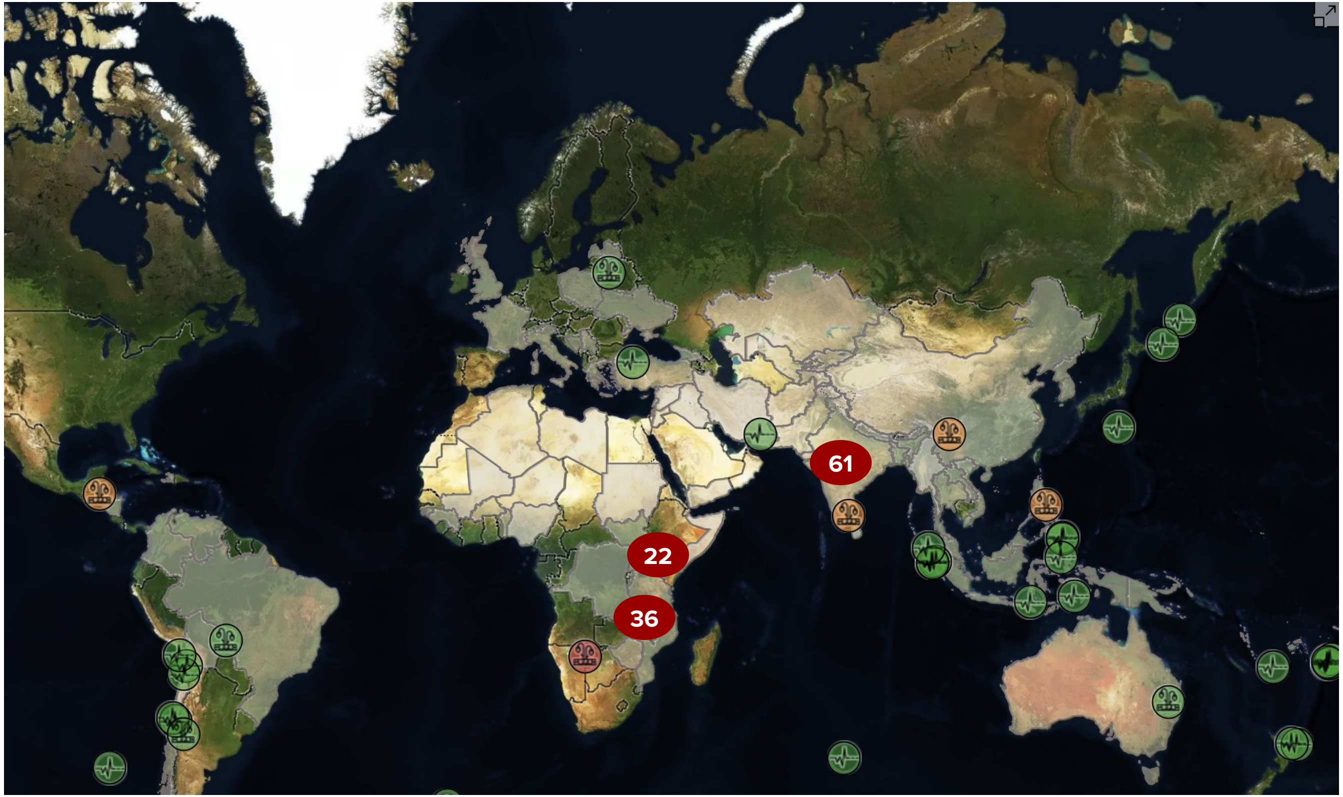
Insights generated from previous analysis:

- **Model optimization for more accurate classification**
  - Data transformation
  - Feature Engineering
  - Model training methods

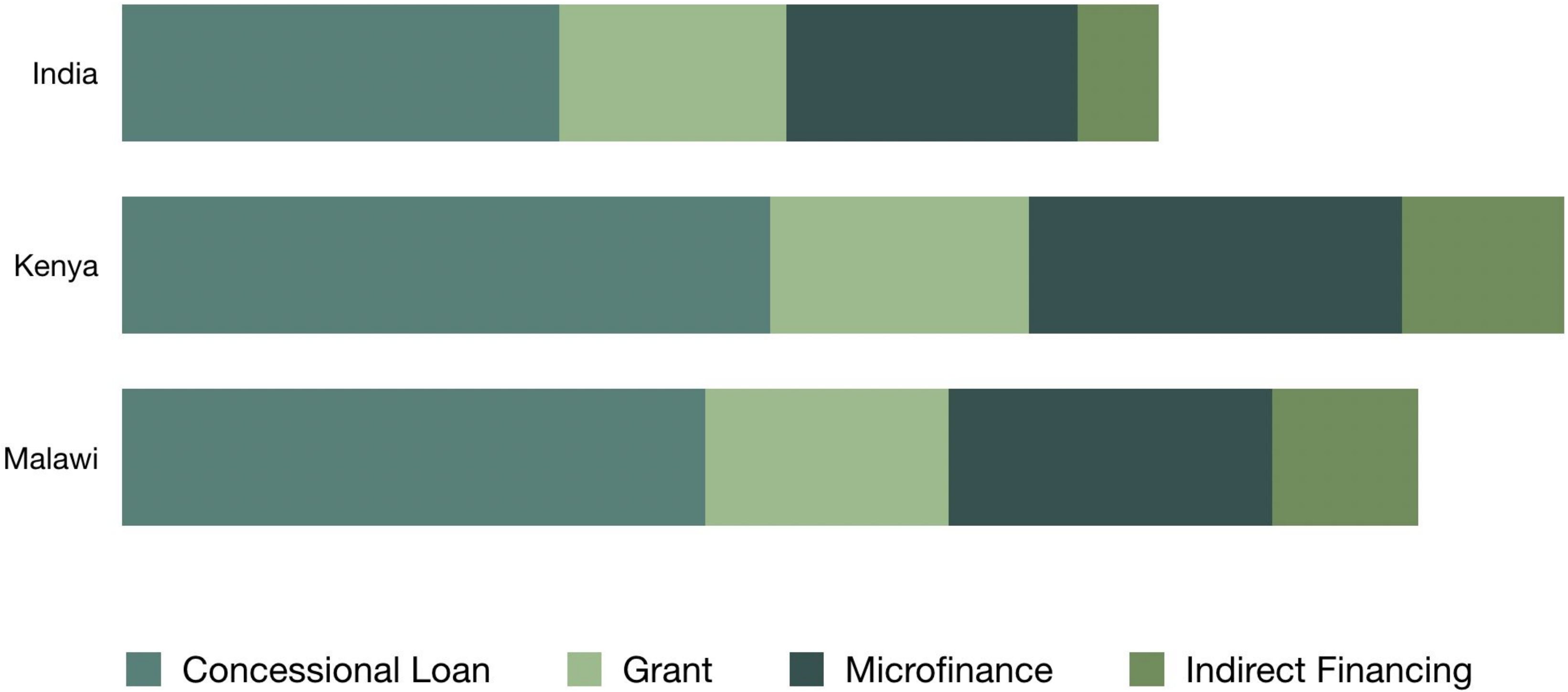
- **Global policy analysis framework development**
  - Initiate the data labeling pipeline for better document classifications
  - Identify more dimensions of financing mechanism characteristics (financier, amount, etc.)
  - Provide country-level support with more complete information on financing availability
  - Institutional research on how restoration financing mechanisms match up with subsidies for agricultural intensification

3. Recommendation - Global Overview

Number of available documents

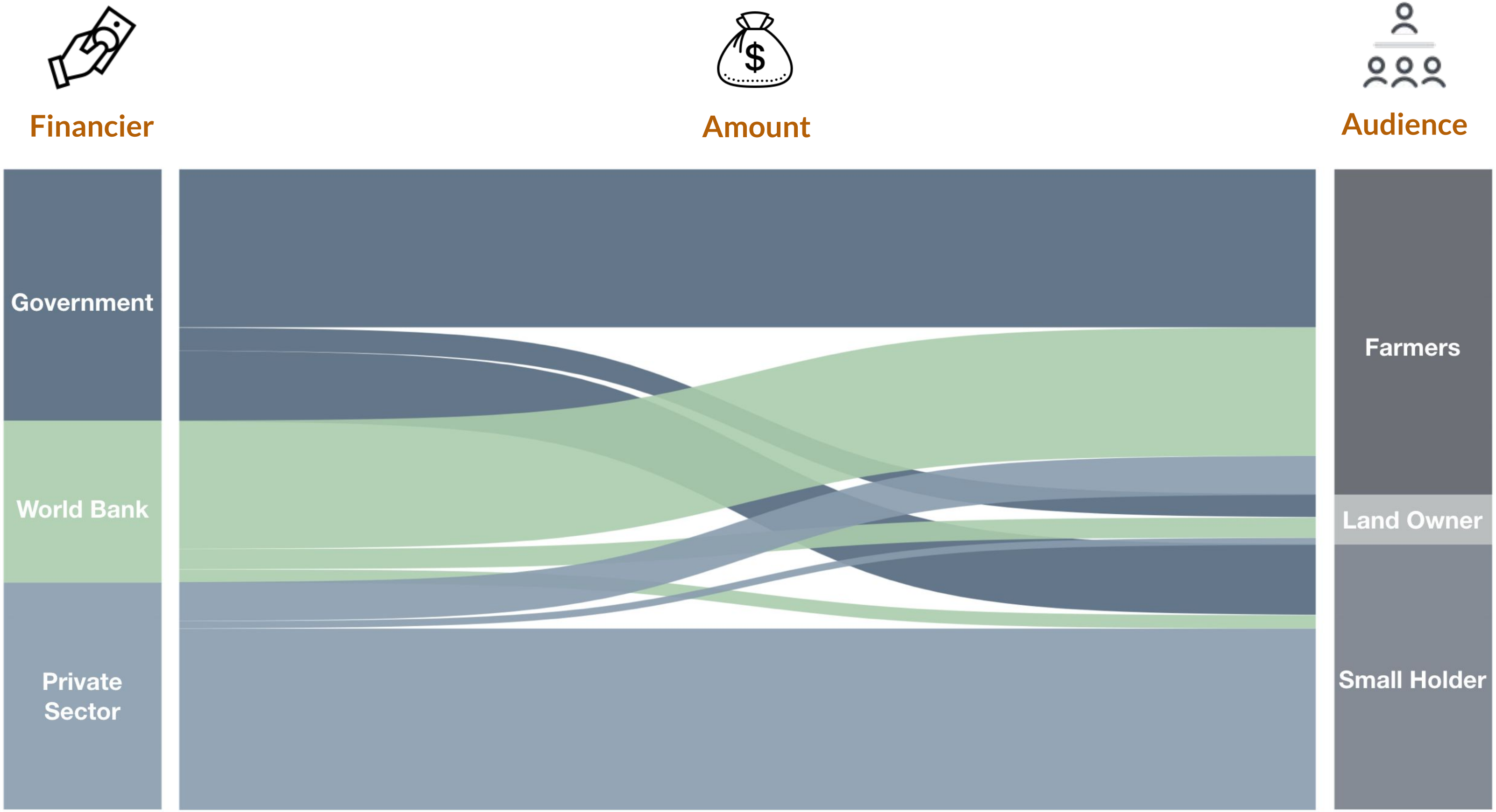


3. Recommendation - Country View



Number of different **types** of documents

3. Recommendation - Financing Flow



3. Recommendation - Financing Flow

Solution:

The gap between the governmental policies and its country’s audience

4. Conclusion

- **Analysis** for the project
- **Improvements** and the application of results
  - Labelling Model Optimization - Snorkel
  - Labelling Model Optimization - BabbleLabble
  - Feature Engineering
  - Neural Network Model