**Bachelor of Technology - 1 Project**

on

# Data Driven Legal Reforms

Project Code: P2

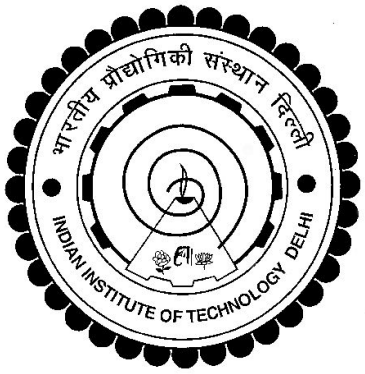
## Submitted by


Harsh Vardhan Rai
2015ME10101

Utkarsh Singh
2015ME10686


## Supervised by

Prof. Nomesh Bolia





Department of Mechanical Engineering
Indian Institute of Technology Delhi


November 2018

# Abstract

Previously, students have worked on creating a very large dataset of court case summaries through web crawling. The process is still ongoing, but data of more than 68% of all the districts has been scraped. Earlier work was focused on identifying anomalies on a host of variables associated with the case disposal time, such as its Mean Disposal Time, Zero Day Disposal rate, Tail Bounds, etc. and developing an aggregate measure of the same. Existing probabilistic distance algorithms were surveyed to develop a better aggregation metric. The list of case types across all the various districts were clustered to develop one holistic, standard list. Attempts were also made to develop predictive models that predict the time a case would take, given information regarding the case, such as case act, year of filing, court of filing, etc. Multiple family of models - Gradient Boosted Trees, Random Forests and Multilayer Perceptrons - were trained on different datasets, and analysed for accuracy. Continuing on this work, we extend the above analysis to the cluster level to understand data at a more high level, and develop and train better models in an attempt to improve on the prediction accuracy given by previous models.

# Acknowledgements

We would like to thank Sansiddh Jain, M.Tech. student from CSE department for the fruitful discussions and valuable guidance. Sansiddh has worked on this project previously as part of his B.Tech. Project.

# Contents

# 1. Introduction

*The following project is a part of the Faculty Interdisciplinary Research Project (FIRP) family, with the primary investigators being Prof. Nomesh Bolia from the Mechanical Engineering Department, and Prof. Mausam from the Computer Science Department.*

A large database of court case summaries from Indian district courts has been scraped. The scraping is ongoing and as of now, 420 of the available 610 districts have been scraped. The raw data is extremely large and is of the order of 16 TB in size. The summary of a case contains information including but not limited to, registered case act and case section, registration and disposal date, number of hearings and corresponding hearing dates, name and address of petitioners and respondents and the name of the judge. The information is scraped from http://services.ecourts.gov.in, and a snippet of the information on the website is given below:

**Chief Judicial Magistrate**

**Case Details**

| | | |
|---|---|---|
| Case Type | : C R Case | |
| Filing Number | : 1000201/2013 | Filing Date: 30-10-2013 |
| Registration Number | : 201/2013 | Registration Date: 30-10-2013 |
| **CNR Number** | **: ASTN04-000533-2013** | |

**Case Status**

| | |
|---|---|
| First Hearing Date | : 29th November 2013 |
| Next Hearing Date | : 24th September 2018 |
| Stage of Case | : Evidence before charge |
| Court Number and Judge | : 6-Additional Chief Judicial Magistrate, Tinsukia |

**Petitioner and Advocate**

1) Smti Rishmita Sarkar

**Respondent and Advocate**

1) Sri Dulal Sarkar

**Acts**

| Under Act(s) | Under Section(s) |
|---|---|
| Indian Penal Code | 406 |

**History of Case Hearing**

| Registration Number | Judge | Business On Date | Hearing Date | Purpose of hearing |
|---|---|---|---|---|
| 201/2013 | Judicial Magistrate 1st Class, Tinsukia | 29-11-2013 | 30-12-2013 | Hearing on Petition |
| 201/2013 | | 30-12-2013 | 27-01-2014 | Hearing on Petition |
| 201/2013 | Sub-Divisional Judicial Magistrate(Sadar),Tinsukia | 27-01-2014 | 21-03-2014 | Hearing on Petition |
| 201/2013 | Sub-Divisional Judicial Magistrate(Sadar),Tinsukia | 21-03-2014 | 31-03-2014 | Hearing on Petition |
| 201/2013 | Sub-Divisional Judicial Magistrate(Sadar),Tinsukia | 31-03-2014 | 12-05-2014 | Hearing on Petition |
| 201/2013 | Sub-Divisional Judicial Magistrate(Sadar),Tinsukia | 12-05-2014 | 20-06-2014 | Hearing on Petition |
| 201/2013 | Sub-Divisional Judicial Magistrate(Sadar),Tinsukia | 20-06-2014 | 01-08-2014 | Hearing on Petition |

*Figure 1 : Snippet of Information on website*

Such detailed information offers a lot of possibilities. Defining the Disposal Time of a case as the time (in days) between the First Hearing Date and the Final Decision Date, one can visualise this metric over different levels of granularity to understand how much time district courts take to close a case. Zero Day Disposal rate, which is defined as the percentage of cases disposed during the day of their first hearing, is also a very useful metric. Other meaningful metrics such as Tail Bounds and Standard Deviation can also be analysed. Such an analysis can be used for anomaly detection which can give more meaningful insights on the nature of the data. In addition, the information available to us can also be used to develop predictive models that can predict the Disposal Time of a given case.

For this BTP, we focused on the following things:

- Scraping and databasing case summaries from as many districts as possible. With data from as many as 610 districts available on the website, it was important to scrape data from all the districts as soon as possible. We started with data from 263 districts at the start of the semester, and we were successfully able to scrape and enter the data from 157 more districts into the database, taking up the tally to 420.

- Extended the existing case type analysis to the cluster level, and augmenting the visualisations to the project website - https://www.cse.iitd.ac.in/dair/courtanalytics/. This has been done for metrics such as Mean Disposal Time, Zero Day Disposal, Tail Bounds for 3 months, 6 months, 1 year and 3 years.
  The 16 clusters are:
  - Civil Act
  - Civil Appeal
  - Civil Arbitration
  - Civil Application
  - Civil Case
  - Civil Petition
  - Criminal Act
  - Criminal Appeal
  - Criminal Arbitration
  - Criminal Application
  - Criminal Case
  - Criminal Petition
  - Sessions Case
  - Special Case
  - Small Cause Case
  - Other

- Cleaning of the available case acts information. The case acts information, as the way it was scraped from the website, was not available to us readily in a usable format. The comma separators, that were meant to distinguish between the section numbers of the Act under which the case was filed, were either missing, or were replaced with arbitrary characters. To clean this information, a dynamic programming algorithm with some initial assumptions was designed.

- Create better models, which, instead of predicting the disposal time of a case based on only the initial available information, also incorporates the information gained as the case progresses - thereby incorporating information regarding number of hearings taken place, individual time gap between hearings, and purpose of each individual hearing. Such a model would be applicable in the real world to a much larger extent, as we would not only be able to predict the time that fresh cases would take, but also cases that have been going on for a while.

If given an opportunity to continue on this project in BTP-2, we would continue our work on developing these advanced models that are expected to perform much better than previously tested models. The data such as Purpose of Hearing and Case Acts are too granular to use as it is. Better approaches are needed to be developed for making better use of this information. Additionally, implementing these models to incorporate all these additional information remains a challenging problem. Some key questions that arise - How to represent these information as features for training our predictive models? How should the model architecture be designed so that we are able to provide all necessary information to it without redundancy?

# 2. Literature Survey

A short review of the predictive models that have been tried previously:

- **Random Forests**[1]: Random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set - a single decision tree learns patterns in the training set which are restricted to that set alone, they are not patterns one finds in the generally across all instances of such kind of data. Random Forests, by counter this tendency of decision trees first by inhibiting the tree from using all the features to learn, and secondly by learning multiple trees.

- **Gradient Boosting**[2]: Gradient Boosting is another method that is used to create an ensemble of trees, the purpose of which, again, is to mitigate the tendency of individual decision trees to overfit the training data. Gradient Boosting in the realm of classification is typically only used when the number of classes is very low; if the number of classes is high, it takes a significant amount of time to train the ensemble, and people prefer to use random forests in that case.

- **Multilayer Perceptrons**[3]: A multilayer perceptron (MLP) is a deep, artificial neural network. It is composed of more than one perceptron. They are composed of an input layer to receive the signal, an output layer that makes a decision or prediction about the input, and in between those two, an arbitrary number of hidden layers that are the true computational engine of the MLP. MLPs with one hidden layer are capable of approximating any continuous function.

  Multilayer perceptrons are often applied to supervised learning problems: they train on a set of input-output pairs and learn to model the correlation (or dependencies) between those inputs and outputs. Training involves adjusting the parameters, or the weights and biases, of the model in order to minimize error. Backpropagation is used to make those weigh and bias adjustments relative to the error, and the error itself can be measured in a variety of ways, including by root mean squared error (RMSE).

These models did not perform well - the highest reported accuracy was about 50%. Some of the possible reasons behind this performance are discussed below:

1. **Sequence Data:** Given the fact that a court case entails multiple sessions of hearings, which are generally non-uniformly spaced, we can interpret the problem of predicting the disposal time of a case as predicting when the last hearing will happen. For predicting that, information on the hearing just before the last hearing will be very useful. Going backwards this way, it is realised that this is exactly what Sequence Modelling is - given the sequence of observations, we want to predict the next observation. Here, the observation is the hearing date. When the model predicts that the sequence has ended, that will give us the disposal time of the case.

2. **Model Features:** In previous work, the features that were used for training the model were the information on the State, District, Court, Type of Case and the Year of Filing the case. The fact that despite a lot of hyperparameter tuning these models couldn't deliver, implies that some useful features may be missing. Some features that could provide additional context could be the information on Case Acts, the dates of hearing and purpose of each hearing.

3. **Models under consideration:** Even if we work with the features that were used for the predictive models in previous work, the problem is that these models predict one absolute value. However, it may be the case that for a particular state, district, court, case type, and year of case filing 2 different cases had different disposal times. This information cannot be represented by any of the above models, as the model would only predict a single value. In such a scenario, what we would like is the probability distribution over the disposal times, given the set of input parameters. This may not give direct results, but can be useful in drawing very useful inferences, such as for given input features, is a disposal time of 100 more likely than that of 200? This is what discriminative models do.

# 3. Project objectives and work plan

## 3.1. Problem Motivation

India is the world's largest democracy. Judiciary is a critical pillar of this democracy. Since independence, the functioning of our judicial system has not been able to keep up with the requirement of the citizens of this country. The judicial system is highly overburdened with over 20 million cases currently pending in district courts of which 10% have been pending for over 10 years. There is a large shortage of judges. India has one judge for every 73000 people which is an average 7 times worse than in the United States. The bulk of the cases that the judiciary handles is done through their lowest tier of execution i.e. the district courts.

Data Analysis and Artificial Intelligence can help with making this system better. Descriptive analysis such as performance monitoring and anomaly detection can help identify superlative performances for further investigations and perhaps emulate the model being used elsewhere to get better results. Predictive analysis such as predicting expected disposal time of the case or the next hearing date of the case can help people set expectations from the judicial system. Prescriptive analysis such as scheduling and allocation methodologies can help improve the performance of the courts and achieve better satisfaction for the public from the system.

## 3.2. Project Objectives

From previous works, it has been concluded that predictive models like Gradient Boosted Trees, Random Forests, Multilayer Perceptrons do not work very well on the data that we have. One reason for this could be that these models don't make use of the sequential nature of the data - it would be much easier to predict the disposal time, or the next hearing date of the case, if we make use the sequential nature of the past observations.

In this project, we aim to build a new set of predictive models which attempts to learn this sequence in the data, and also incorporate new methodologies and features from the available case summaries for more accurate predictions of:

1) Disposal time, given a new case and its features
2) Next hearing date, given an ongoing case and its history
3) Disposal time, given an ongoing case and its history

Previous models used features like State information, District information, Case type, Year of case filing, and Disposal time.

We plan to build new models which uses additional features (such as Case Acts, Dates of each Hearing, Purpose of each Hearing) including the earlier ones and make use of different models such as LSTMs, Hidden Markov Fields, and Conditional Random Fields which are successfully being used in sequence modelling tasks. We will also explore some more discriminative models like Mixture Density Networks that don't build on sequence modelling, but are expected to work better than the earlier models.

## 3.3. Methodology

The method employed to obtain final useful data involves the following steps:

1) **Data Procurement**: Raw data is procured from http://services.ecourts.gov.in. For each case the website provides numerous details like date of filing, date of disposal, intermediate hearing date, hearing details, case type, acts under which the case has been filed, litigant, defendant, judge presiding and court of filing. This data from the website is scraped in the form of a text file by an automated script. The data is available on the website for 610 districts. As of now, data from 420 districts have been successfully scraped.

2) **Databasing:** The useful features are extracted from the raw data and entered to a database hosted on a server. The database is structured properly and allows multiple scripts to enter the information to it in parallel. There are eleven partitions of the database, each of them storing information from three different states. Currently the database hosts information from 420 districts.

3) **Final useful data:** For the purposes of getting usable data for our models and computing metrics, we have scripts that fetch the relevant data from the database and write them into CSV files that can be directly used.

Thus, there exists a pipeline where we have the scraping scripts running in the background, downloading the raw case data every minute. Once that is done, a separate script is run to enter the information from that district into the database, so that the data is both backed up and more structured. Once that is done, we can obtain this structured data in the form of CSVs that can be used for our analyses.

# 4. Work Progress

## 4.1. Theory

A short description on the models that we have explored, or intend to explore in future work (if given the opportunity to work on it in BTP-2):

1) **Mixture Density Networks (MDNs)[6]:** MDNs are discriminative models used to learn the distribution $p(\mathbf{y} \mid \mathbf{x})$. They use the outputs of a neural network to parameterise a mixture distribution. A subset of the outputs are used to define the mixture weights, while the remaining outputs are used to parameterise the individual mixture components. The mixture weight outputs are normalised with a softmax function to ensure they form a valid discrete distribution, and the other outputs are passed through suitable functions to keep their values within meaningful range.

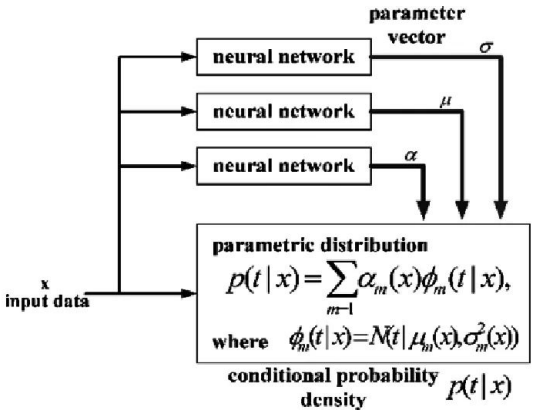



*Figure 2 : Structure of Mixture Density Networks[7]*

2) **Long Short Term Memory (LSTM)[5]:** They are a special kind of Recurrent Neural Network, that are able to retain past information and are able to use it as a context for making predictions along with the recent information. This makes this model more powerful than the general RNNs and also give better accuracy, especially on time series data. LSTMs perform very well on problems like Speech Recognition, which is itself a very complex task. Since for a particular case the observations and the states can be thought of as a discrete time series data, LSTMs can be a suitable predictive model for our objective.

3) **Hidden Markov Fields (HMFs)**: They are generative, graphical model of a joint probability distribution between the observations and the states. It consists of an undirected graph $G = (N, E)$ in which the nodes $N$ represent random variables and $E$ represents the edges. Markov Network (undirected graphical model) is a set of random variables satisfying markov property (memoryless property) described by an undirected graph.
   Mathematically, the joint probability distribution is given by

$$p(\mathbf{y}, \mathbf{x}) = \prod_{t=1}^{T} p(y_t|y_{t-1})p(x_t|y_t)$$

   Where **x** is the observation sequence and **y** is the state sequence.

4) **Conditional Random Fields (CRFs)[4]:** CRFs are discriminative models used for making predictions on sequential data, which is something even HMFs can do. However, unlike Hidden Markov Fields, which assume that the current observation is independent of the previous observations, Conditional Random Fields does away with the joint probability distribution between the observations and the states, and instead models the conditional probability of the state sequence **y**, over the observations **x**.

   Mathematically, CRFs are formulated as:

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{x})} \exp\left(\sum_j \lambda_j F_j(\boldsymbol{y}, \boldsymbol{x})\right).$$

Where

$$F_j(\boldsymbol{y}, \boldsymbol{x}) = \sum_{i=1}^{n} f_j(y_{i-1}, y_i, \boldsymbol{x}, i),$$

   With $f_j(y_{i-1}, y_i, x, i)$ being a feature function, $\lambda$ is the vector of parameters to be learnt and $Z(x)$ is a normalising factor.

## 4.2. Problem Description

Given case summaries, where each case is characterized by the following parameters :

1) Case Type
2) Case Act
3) Dates of Hearings
4) Purpose of Hearings
5) Year of Filing Case
6) Court Information
7) District Information
8) State Information

We have to predict the disposal time or the next hearing date for the given case. As it was observed that models like Random Forest, Multilayer Perceptron and Gradient Boosted Trees did not generalise well, our task was to do better on this problem with predictive model that are expected to be more suited for the above task, such as Mixture Density Networks, LSTMs, Hidden Markov Fields, Conditional Random Fields.

## 4.3. Data Procurement

The data is procured from the government website https://services.ecourts.gov.in.

Several challenges that keeps inhibiting the process of data procurement:

- The website (from where data is procured) often goes offline for several days, during which scraping stops altogether.

- The connection to the server of the website is not reliable, and several kinds of exceptions often arise. Tight measures of exception handling had to be employed to ensure reliability.

- On several instances, the table of cases used to be extremely huge, and scraping consumed multiple days, severely slowing down the entire process. Efficient random sampling methods were devised, to ensure samples are small, yet representative enough of the complete data.

- The website underwent constant changes, and the code had to constantly be reworked to update according to the new website.

- Running multiple scraping processes for downloading large volumes of data in parallel would often cause slowing down of scraping process, thereby negating the purpose of multiple processes. Efficient methods were thought of and incorporated to ensure faster and seamless downloads.

- Every time new parameters are used to query on the website for obtaining data, a Captcha problem needs to be solved. We use Google's Tesseract-OCR for this purpose.

As the website requires some information to be entered for querying the data, we use the web automation library, Selenium, that automates the entering of all the parameters without human intervention, query for the data, and finally download the obtained data.

Most recent figures of data procurement are as follows:

- Raw data for 420 districts out of 610 has been scraped.
- All of the useful data from these 420 districts have been entered into the database.

## 4.4. Cleaning of Case Acts Information

Some Case Acts information in the database was found to be bad. The comma separators, that were meant to distinguish between the section numbers of the Act under which the case was filed, were either missing, or were replaced with arbitrary characters. We couldn't come up with a definite approach, but a heuristic approach to clean this was used which may not be very dependable.

For example, if the sections under which the case was filed were "443A,212,121,33", then it is available in the database as '443adr212121wqw33'. One way of obtaining the original string was to maximize the number of 3 digit case acts as we observe that these section numbers are most likely to occur. This heuristic uses dynamic programming.a

One other heuristic that we are trying to build will try to segregate the section numbers by maximizing P(x), where x is a given valid segmentation of section numbers and P(x) is the probability that the valid segmentation x occurs. Since the length of string is generally less than 15, the number of possible valid segments are small. Therefore, we can find x for which P(x) is maximum by checking over all possible values of x, while taking into account the relationship between sections and the case acts, and then populating the database with this computed x. We may require some professional advice to gain little knowledge about different case acts and chances of a pair of section number and case act occurring together.

## 4.5. Cluster Level Data Analysis

Having mapped the case types across all districts to 16 distinct clusters in order to capture the information given by a case type at a higher level, and standardising it across all districts, it became important to also understand the behavior of cases across these new clusters in the form of metrics such as Mean Disposal Time, Zero Day Disposal rate and Tail Bounds. This will be published on the project website - https://www.cse.iitd.ac.in/dair/courtanalytics/. It includes visualisations of these metrics on the Indian map to get an idea of how different districts and states rank in comparison to each others based on these metrics. These rankings can also be viewed in a tabular form for better readability.

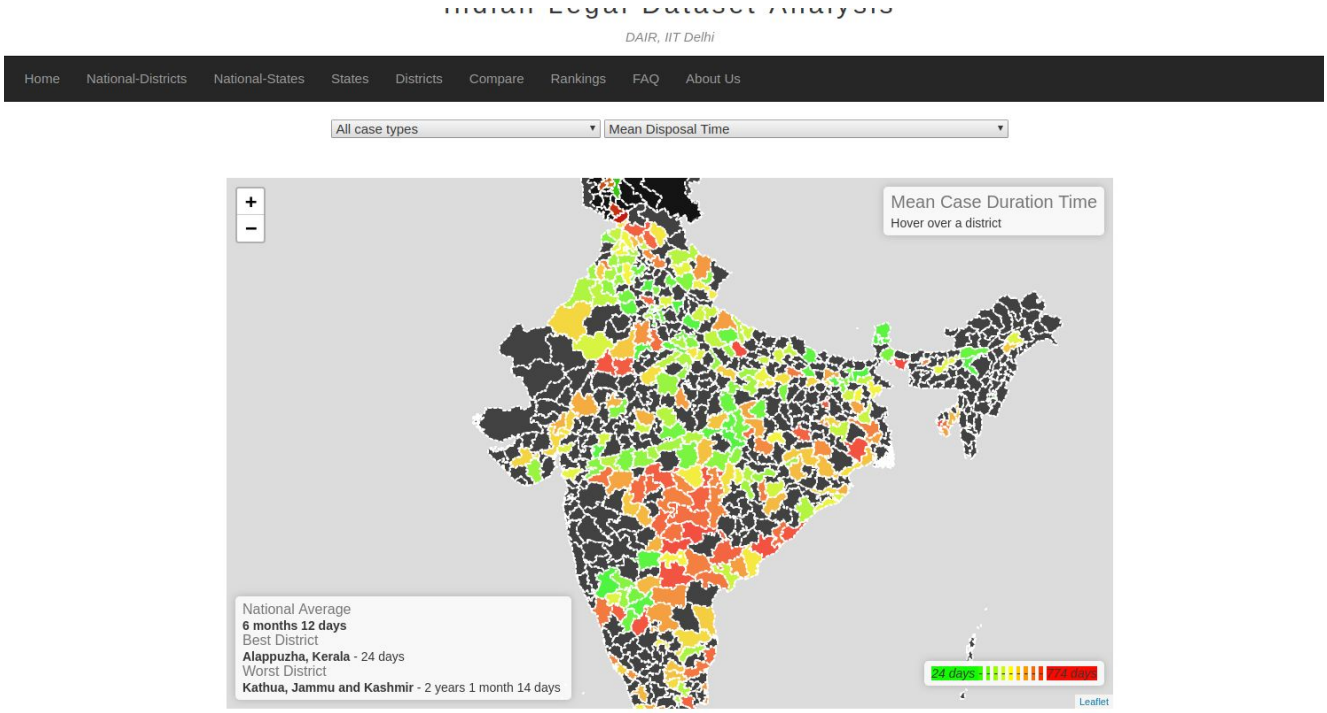



*Figure 3 : Snippet of project website*

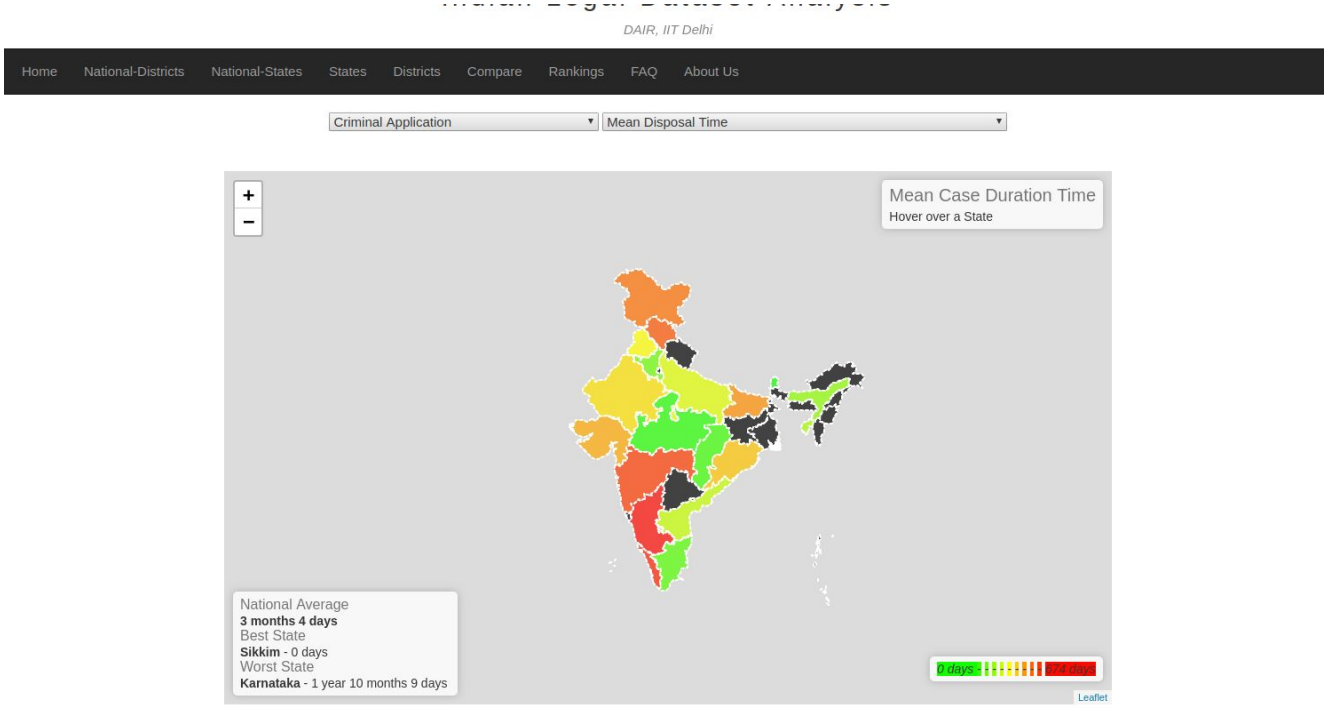



*Figure 4 : Snippet of Information on website*

## 4.6. Mixture Density Networks (MDNs)

The unison between the traditional neural network and the mixture model part is achieved by using the log-likelihood of the linear combination of kernel functions as a loss function of the neural network. By choosing a mixture model with a sufficient number of kernel functions, and a neural network with a sufficient number of hidden units, the Mixture Density Network can approximate any conditional density $p(\mathbf{y} \mid \mathbf{x})$ as closely as desired.

The representation graph of the Mixture Density Network model is as follows:

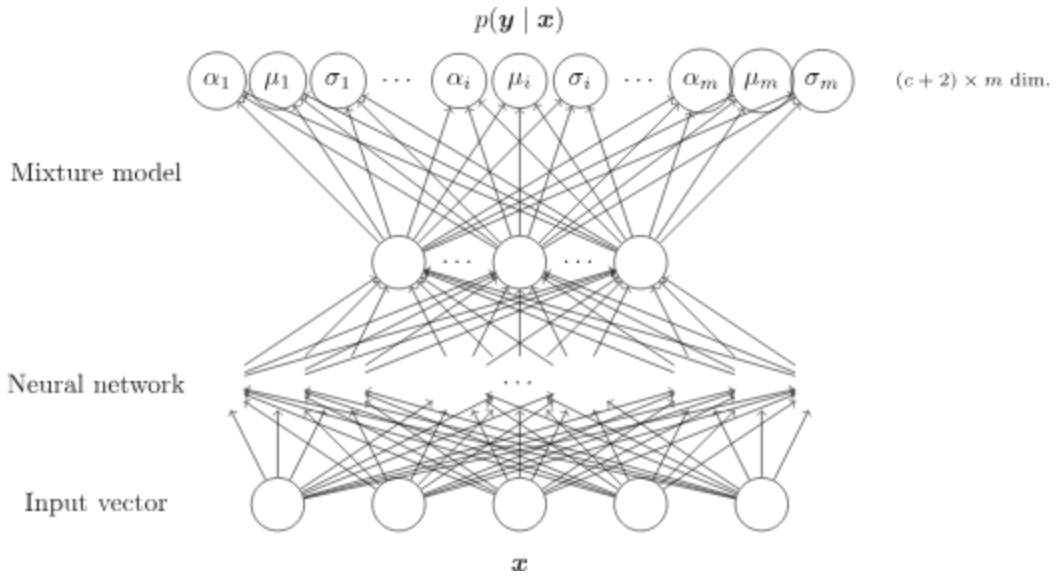



*Figure 5 : Representation graph of MDN*

For our model, the features corresponding to ‘x’ are the State information, District information, Court information, Case Type information (in the form of cluster) and the Year of Filing. The purpose of choosing these features was to observe how this model will behave, as compared to the models that were tested before, as the old models also used the same features.

Given the value of these features, we can now obtain the probability that a case will take ‘y' months of time for disposal.

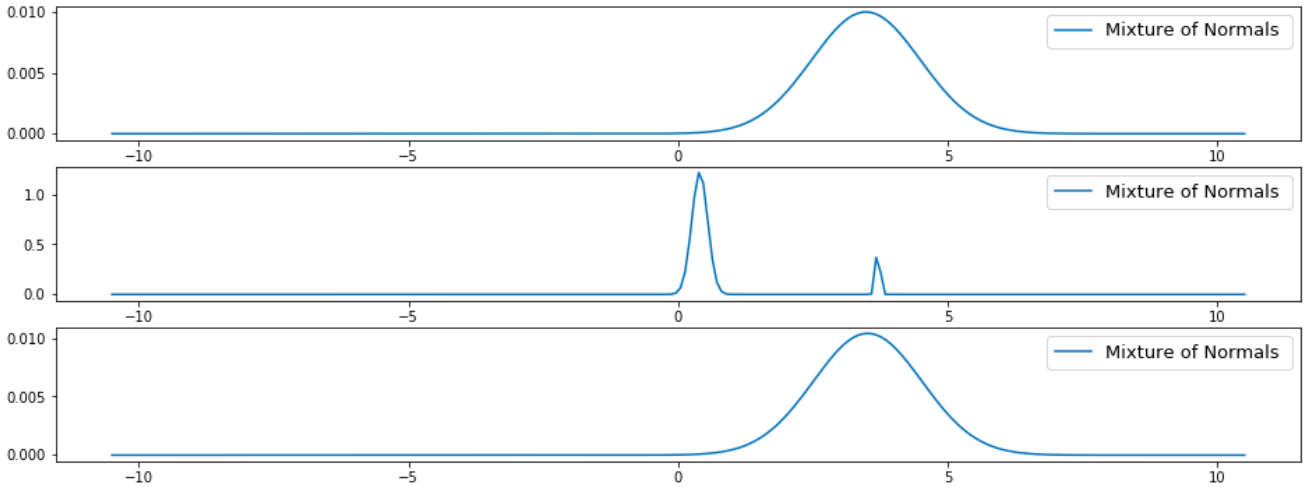



*Figure 6 : Probability distribution of disposal time (in months) for various set of feature values*

Each of the plot above shows the distribution of the random variable ‘y’ (no of months for the case to dispose) for 3 set of feature values from Sikkim. The distribution model here uses 3 Gaussian Normal distributions to capture the overall distribution of a particular case.

The parameters of the distribution are predicted by a neural network with the following architecture.

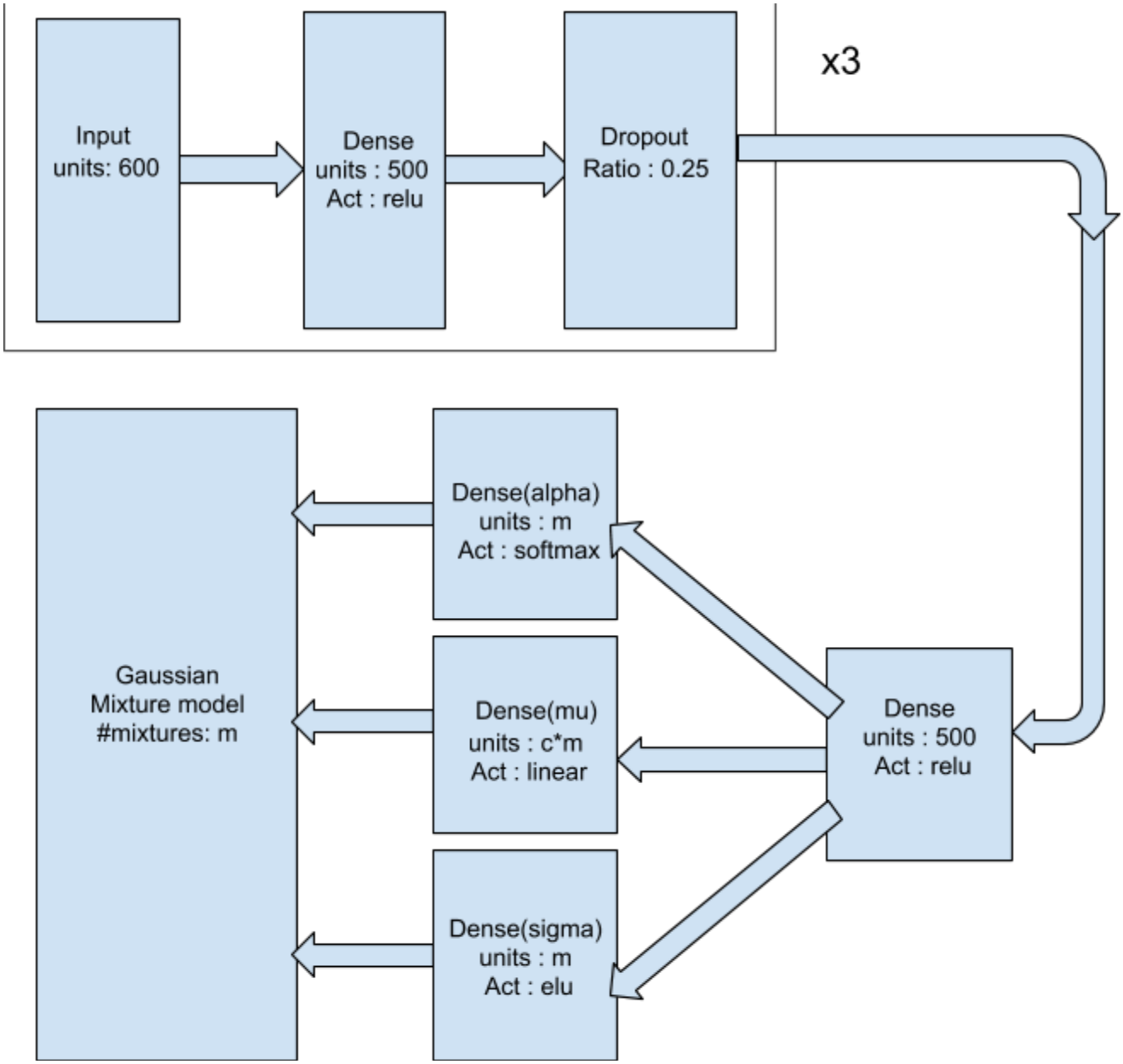



*Figure 7 : Block diagram of MDN model architecture*

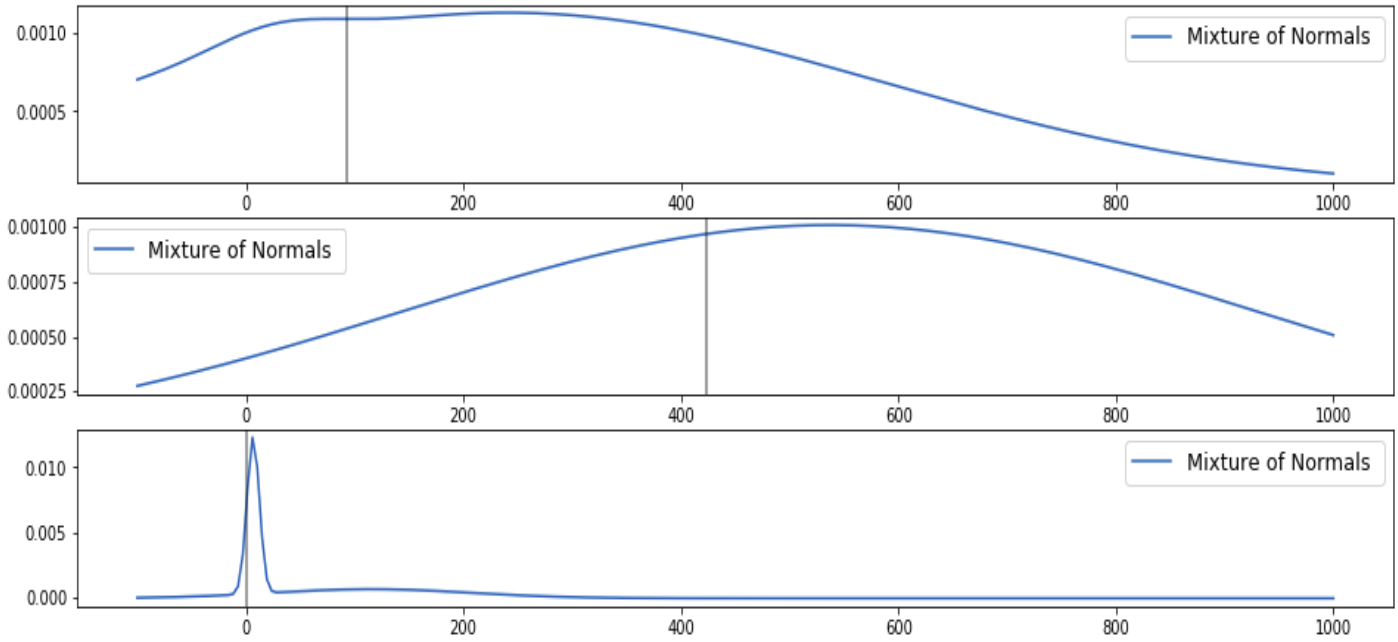



*Figure 8 : Probability distributions with ground truth values*

The above shown plots represents the probability distribution of the following (State; District; Court; Cluster; Year of Filing) quintuple respectively:

1) (Maharashtra; Shahada; Civil Judge Junior Division, Samudrapur; Criminal Application; 2015)

2) (Punjab; Bathinda; Chief Judicial Magistrate, Taluka Court, Talwandi Sabo; Criminal Application; 2017)

3) (Madhya Pradesh; Mandleshwar; Civil Court Barwaha; Civil Case, 2015)

The vertical lines in the plot represent the ground truth value. Examples were taken from the training data. The distribution is constituted by three gaussian curves among them one dominates over the other.

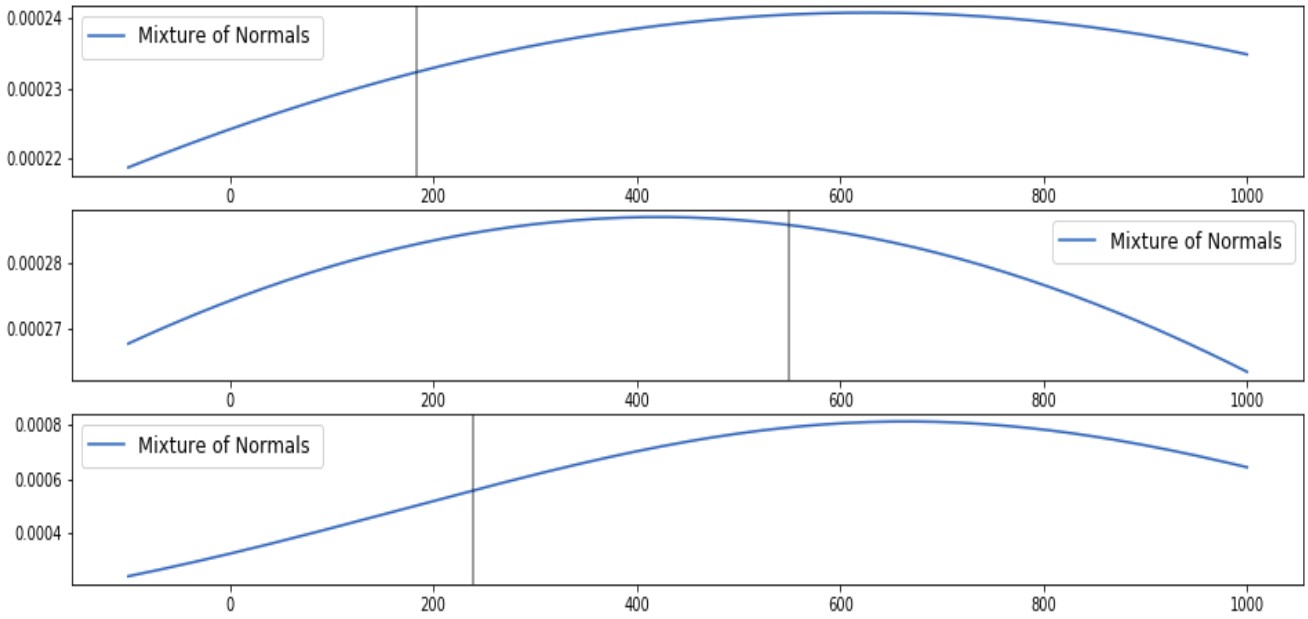



*Figure 9 : Probability distributions with ground truth values*

The above shown plots represents the probability distribution of the following (State; District; Court; Cluster; Year of Filing) quintuple respectively:

1) (Bihar; Kaimur; DJ Div. Bhabhua; Criminal Appeal; 2013)
2) (Haryana; Sirsa; Chief Judicial Magistrate; Sirsa; Civil Act; 2015)
3) (West Bengal; Kolkata; Chief Judge, City Sessions Court, Calcutta; Civil Appeal; 2015)

The vertical lines in the plot represent the ground truth value. Examples were taken from the test data this time. The ground truth value is reasonably closer to the mode of the distribution. Considering the limitation of features available, MDN gives a reasonable approximation of the disposal time and the probability that the case will dispose before a given period.

## 4.7. Long Short Term Memory (LSTM)

While we did think about how to make use of LSTMs theoretically, it remained a challenge as to how to feed the features to the model. We do not want to give one-time information (such as state, district and court information) at every time step to the network, as that will be computationally expensive and will only slow the learning.

One possible approach that came out of our discussions was to encode the information of the hearing dates into a 128 dimensional vector (the vector dimension can be a hyperparameter for the model), which can act as features along with the other general case information (State, District, Court, Cluster) to train a Multilayer Perceptron. The idea behind this was to represent the hearing dates information in some way that could be used for learning a model to predict the disposal time. That is, we will learn the embedding of the hearing dates. This model is also expected to represent the claim that the more we know about the past hearing dates, the better the model can tell us about the disposal time.

We are in the process of implementing the model right now, and it is expected to get completed by the final presentation. This model, however, won't be useful for predicting the next hearing date. For that we may need more contextual information such as nature of hearing, but that data is far too noisy, and just like the case types, may have to be clustered.

## 4.8. Results and Discussion

- Case summaries from 157 districts have been scraped since the beginning of this BTP. We expected to complete at least 200 districts but this mark was missed due to challenges discussed earlier.

- New data for all the districts has been successfully entered into a database that allows user to conveniently query for any type of information they may need.

- Case Acts information in the database was found to be bad and we couldn't come up with a definite approach but a heuristic approach to clean this was used which is not dependable. A better heuristic has also been proposed.

- Mixture Density Networks are able to represent the probability distribution over the disposal time given the features reasonably well, and can be thought of as a more general model compared to the models tried earlier like MLPs and Random Forest.

- A predictive model incorporating LSTMs for learning sequential nature of the data has been proposed. It is to be seen how it will fare.

- A pipeline had been successfully created and implemented, such that whenever we download and incorporate new information to our database, we can re-compute all the metrics and add it to the website by simply running a few scripts.

- Data sometimes 'disappears' from the source website. This hampers scraping. While scraping scripts uses tight exception handling to solve most of the issues faced, this is something that we cannot do anything about.

- New metrics for analysis at the cluster level have been computed for the project website.

# 5. Conclusion and Further Work

- As exhibited by the results from Mixture Density Networks, probabilistic modelling using discriminative models seems to be a promising direction. As CRFs are also discriminative models that are more complex than MDNs, we expect a better performance from them.

- Cleaning of section number information from Case Acts will be essential as it will provide a contextual information for a particular case summary, now that we have brought the case summaries to a cluster level, which is at a very low level of granularity.

- An architecture for modelling the next hearing date using LSTMs without the problem of redundant features at every time step need to be thought of. One possible solution might be to learn the initial state of the network - that the initial state of the network would depend on the common features (namely state, district and court).

# 6. References


- [1] - https://en.wikipedia.org/wiki/Gradient_boosting#Gradient_tree_boosting
- [2] - https://en.wikipedia.org/wiki/Random_forest
- [3] - Multilayer Perceptron - https://skymind.ai/wiki/multilayer-perceptron
- [4] - Hanna M. Wallach. “Conditional Random Fields: An Introduction”. Technical Reports (CIS), 22.
- [5] - http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- [6] - Alex Graves, Generating Sequences With Recurrent Neural Networks, arXiv:1308.0850 [cs.NE]
- [7] - https://www.researchgate.net/figure/The-Structure-of-the-Mixture-Density-Network_fig1_226666520