



## Text Analytics for a Real Estate Service Provider

- Reduced operational costs by 65%
- Time taken per document to plot boundaries reduced from 2 hours to 5 mins



## Customer Background

The customer is a real estate and infrastructure consulting services provider. Their expertise allows their customers (real estate property owners – both public and private) make better informed decisions about infrastructure projects and cut costs in project implementation. They help efficiently design and provide services to real estate property owners in the U.S.



### Tools

The solution as well as the application was fully built in Python using several libraries. Here is an exhaustive list of libraries that were used:

#### **OCR and Preprocessing** -

Pytesseract, OpenCV, Tabula, xPDF, Align, Poppler, Tesseract

#### **Regular Expression** – re

LSTM & CRF - pycrfsuite, tensorflow, GATE, keras, docsanno

#### **Plotting** - ezdxf, svg,

**Web Application** - boto3, flask, msal, Flask, AWS S3, Postgres, uWSGI, Nginx



### Business

Text Analytics



### Domain

Real Estate



### Key Highlights

- Automated the process of traverse file creation reducing the manual time to almost zero.
- Reduced man hours from 4800 to 960 hours per month.
- 100% text extraction accuracy achieved.

# Business Requirement •



The boundary description of the plots for deed and lease documents is written in Metes and Bounds format. Although still prevalent, it is in an archaic verbose format which is not amenable to be used for modern drawing software like ArcGIS, CAD etc. These tools require input in a shorthand format often called traverse files. The conversion from Metes and Bounds to traverse format is generally done manually, involves a lot of effort and takes many hours to generate the required input file. Another pain-point was the length of deed documents – they can run in hundreds of pages. And finding these plot descriptions was like finding a needle in haystack

## Challenges

- Metes and Bounds deeds are mostly present as PDF files that pose the following challenges:
  - Scanned copies of deeds (often with poor lighting).
  - Rotated text.
  - Highlighted (with marker) text.
  - Watermark in the background.
- Converting such PDFs to text files lead to significant upfront data loss and OCR error.
- Multiple types of curve components which can occur randomly without a fixed pattern. Hence, writing a logical rule is near to impossible.
- The length of deed documents and sparse presence of text of interest i.e. plot description.



## Objective

- To automate the process of Metes and Bounds to traverse file conversion and plotting.
- To create an application to upload PDFs, extract traverses and review the same, along with the ability to create and view plots drawn using the traverse rules extracted.

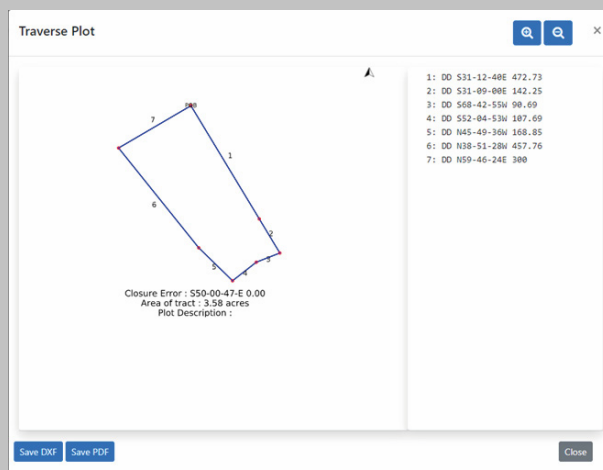


## Solution Overview

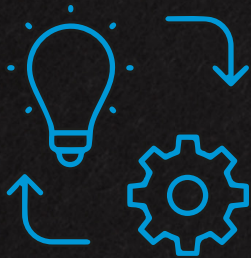
Using teX.ai, the following solution was designed

- A solution to extract the different plot components – direction and distance for lines and radius, chord angle etc. for a curve.
- This solution was built using two methods:
  - one using simple regular expressions.
  - second using custom entity detector using LSTM and CRF models.
- The solution was extended to provide plotting capabilities for the extracted plot boundaries.
- The plot drawings can be saved as DXF files which can be opened and edited in CAD.
- To consume this solution, a web application was created with functionalities to edit the extraction results in case there was error and to plot the final boundary.

**Sample plot diagram which is automatically generated post extracting coordinates from input deed document**



# Approach & Implementation •



A solution was created using OpenCV and Deep Learning-based methods to pre-process PDFs (rotation, highlighting correction, etc.) and convert them to text. Using teX.ai, two approaches were implemented to automate the conversion of the text file to traverse file.

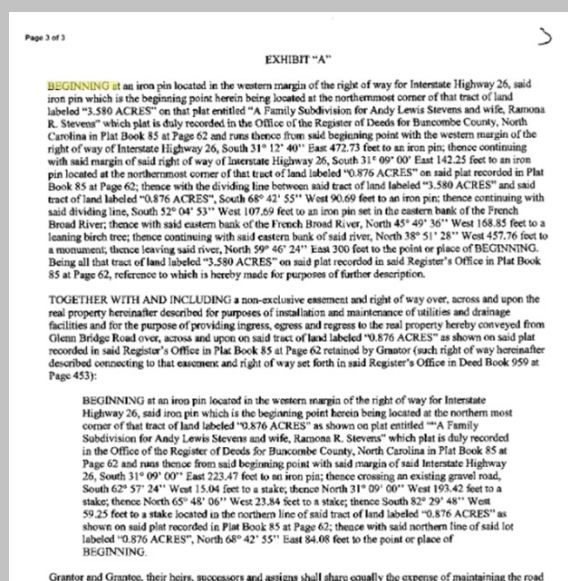
**Regular Expression** - Identify curve components and their values by finding the occurrence of keywords using regular expression patterns e.g. Chord Direction or Non-Tangent for a Non-Tangential Curve.

Although quite effective, this approach needed manual intervention to define rules.

Moreover, defining a comprehensive set of rules with high accuracy is tricky and prone to manual errors. Hence, the teX.ai team came up with another method by using Conditional Random Field.

We implemented Machine Learning based tagging using Conditional Random Field.

## Input



# Output

	Part	Direction(DMS)	Distance	Radius	Angle	Arc_Length	Left_Right	Description
+		POB						
+	DD	S31-12-40E	472.73					
+	DD	S31-09-00E	142.25					
+	DD	S68-42-55W	90.69					
+	DD	S52-04-53W	107.69					
+	DD	N45-49-36W	168.85					
+	DD	N38-51-28W	457.76					
+	DD	N59-46-24E	300.0					

## Business Impact



Creation of augmented AI page to compare the extraction results with actual document ensured 100% accuracy in extraction with just minutes of manual intervention.



Reduction of operational costs by nearly 65% due to automation.



Drastic reduction in time taken per document to plot boundaries – from 2 hours to 5 mins – 30x reduction in time taken per document.



Reduced man hours from 4800 to 960 hours per month.

**Get in touch to see how  
teX.ai can help you!**



[info@tex-ai.com](mailto:info@tex-ai.com)



+1 (888) 207 5969 (Toll-free)



[www.tex-ai.com](http://www.tex-ai.com)



Suite 100, 19925 Stevens Creek Blvd,  
Cupertino, CA — 95014