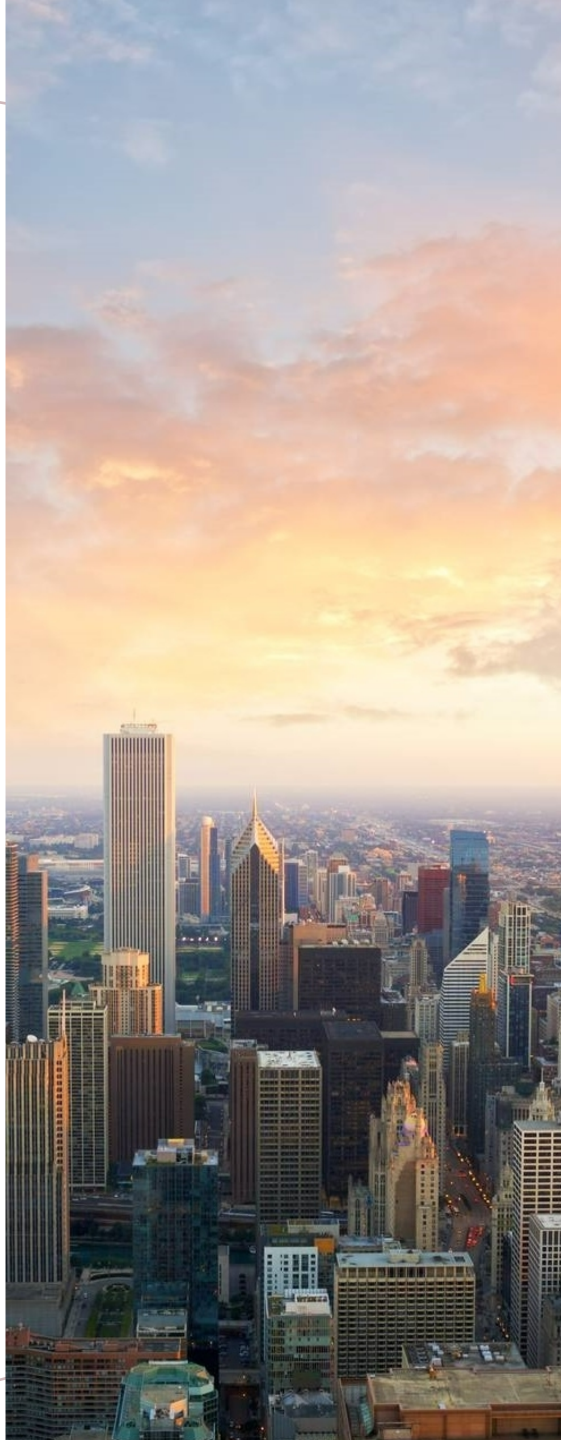
OUR STREETS:

NER APARTMENT CLASSIFICATION MODEL

Presentation By: Kwasi Brooks

1. Introduction
2. Problem Statement
3. Deep Learning Model Used
4. Challenges and Solutions
5. Data Structure and Gathering
6. Final Product

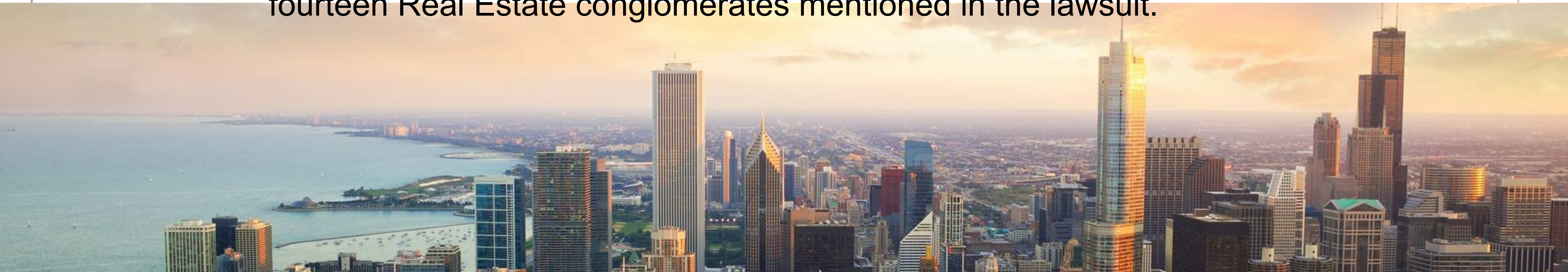


INTRODUCTION

Name: Kwasi Brooks

Occupation: Data Scientist at the DOES (Department of Employment Services)

Project Description:
Combining Named Entity Recognition (NER) and Natural Language Processing (NLP) techniques to identify properties owned or managed by entities named in the class action lawsuit against RealPage, Inc in DC. The final product allows users to input their DC address and get an output telling them whether their address is owned by any of the fourteen Real Estate conglomerates mentioned in the lawsuit.



IMPORTANCE OF THE WORK

- Washington DC is among the most expensive cities to live in in the world.
- This is not a natural occurrence of supply and demand but rather a concerted effort by real estate conglomerates and other special interest groups.
- The primary problem that we seek to address within this project is how much of the city of the District of Colombia is owned by any of the fourteen landlords included in November 1st, 2023, antitrust lawsuit against RealPage, Inc.
- How can we empower DC residents with transparency regarding their rental prices and who literally owns their streets?

WHAT'S THE MODEL: BERT

What am I using?

I am using the BERT model (Bidirectional Encoder Representations from Transformers) that has been fine-tuned for Named Entity Recognition (NER). Specifically, the code uses the BertForTokenClassification class from the transformers library, along with a pre-trained BERT model.

Bert Model: dbmdz/bert-large-cased-finetuned-conll03-english

BERT Usage Summary

1. Pre-training: BERT is pre-trained on a large corpus of text using a masked language modeling (MLM) objective and next sentence prediction (NSP). This allows BERT to learn deep contextual representations of words in a bidirectional manner.
2. Fine-tuning: My model has been fine-tuned on a dataset for NER, specifically the CoNLL-03 dataset, which contains labeled examples of named entities in English text.
3. Token Classification: The model classifies each token in the input text into categories like 'Organization', 'Person', 'Location', etc., based on the fine-tuning task.

MORE ON BERT

- The BERT model in this code is used to perform Named Entity Recognition (NER), a subtask of Natural Language Processing (NLP). NER is the process of identifying and categorizing key information (entities) in text,
- The model is loaded using the BertForTokenClassification class, which is specifically designed for token classification tasks, including NER.
- The tokenizer corresponding to this model (BertTokenizer) is also loaded to process the input text by converting it into tokens that the model can understand.
- The pipeline function is used to simplify the process of applying the NER model to the text. The pipeline handles tokenization, model inference, and post-processing.
- The pipeline is initialized with the task 'ner' (Named Entity Recognition), and the loaded BERT model and tokenizer are passed to it. This setup allows the model to take in raw text and return recognized entities.
- Extracting Named Entities:
  - The function extract_entities(text) uses the ner_pipeline to process the input text and extract entities. The output is a list of entities found in the text, each associate (e.g., organization names, locati

OVERCOMING PROJECT CHALLENGES

Variation in Address Formats:

- Addresses and building names can be written in many different formats, including abbreviations, punctuation, and varying word orders (e.g., "123 Main St.", "123 Main Street", "123 Main St").
- BERT might struggle to standardize these formats correctly without sufficient training data that reflects these variations.

Scalability and Performance

- Large Datasets: Handling large datasets with BERT can be resource-intensive, especially when running NER on long or complex inputs like addresses. Processing speed and memory usage became bottlenecks which forced other methods of data facilitation.
- Inference Latency: Since BERT models are large and computationally expensive, using them for real-time inference, especially on a GPU, can introduce latency. This might be problematic if the application requires quick responses.

Data Gathering and Normalization

- Finding information on each property conglomerates holdings was challenging and required extensive data scrubbing and sometimes manual entry into csv files.

DATA GATHERING & STRUCTURE

Defendant	Building Name	Defendant Address	Defendant Full Address	URL
GABLES RESIDENTIAL SERVICES, INC.,	Gables Dupont Circle	1750 P STREET NW	1750 P STREET, NW Washington, DC 20036	https://www.gables.com/community/528461
Equity Apartments	Alban Towers Apartments	3700 Massachusetts Avenue NW	3700 Massachusetts Avenue NW Washington DC 20016	https://www.equityapartments.com/washington-dc/catheDriveal-heights/alban-towers-apartments
JBG ASSOCIATES, LLC	13|U	13 and U STREET NW	13 and U STREET, NW	https://www.jbgsmith.com/property/residential/13U/3313312
WILLIAM C. SMITH & CO. INC.	The Oaks	1814 29th STREET SE	1814 29th STREET SE Washington, DC, 20020	https://wcsmith.com/apartments/the-oaks/

ORIGINAL CONCEPT VS FINAL PRODUCT

Original Concept

- Used NER to find Defendant Record information on public data sites in pdf and csv. (Opendata DC)
- Use NER to identify the different shell companies associated with the named entities.
- Match named entities with the properties they own or manage
- Train model on DC address csv file to help the program recognize property building names and addresses via user input

Final Product

- Use data scrubbing on Defendant official websites (Buzotto.com ect) to identify property listings
- Create spreadsheet of defendants and all addresses and properties owned or operated by them in DC
- Use pre-trained BERT model to recognize user inputs and match them to the address & defendant which owns or operates their property.

THANK YOU

Kwasi Brooks

kbrook14@umd.edu