Exiger - Entity Type Classification

AI Studio Final Presentation

Break Through Tech New York @Cornell Tech
December 13th 2022

Our objective is to develop a machine learning model that can quickly classify whether an entity from Exiger’s dataset is a person or company.

Our Goal

1. Create accurate and efficient features that can be implemented in our model.
2. Group the different languages into language groups. Ex: Latin, CJK.
3. Split data for equal distribution of languages in both test and training set.
4. Test different models like Decision Tree, Random Forest, Logistic regression ect.
5. Create a model with high precision, recall and f1 score above 90%.
6. Overall, our product needs to differentiate between Person and Company entities quickly, efficiently, and accurately.

Business Impact

- Clients of Exiger frequently don't distinguish between names of people and companies. This is a challenge because exigers' product strategies vary depending on whether they are working with a person or a business.
- Updating the current rules based system to a machine learning model will make improvements and upkeep easier for the developers who work with this data

Our Approach

September

Data Preprocessing & Business Understanding

Evaluating the dataset provided by Exiger and determining our plan for the model

October

Feature Engineering

Developing features for the dataset in order to increase dimensionality and model accuracy

November

Feature Engineering

Continuing our feature development and preparing for model selection & training

December

Modeling

Model training, testing and evaluation

Resources We Leveraged

HYPEROPT

StackExchange

stack overflow

eCornell

kaggle

Forebears

scikit learn

WIKIDATA

Data Preprocessing

Our Data Set

- Given to us by Exiger
- CSV File: Entity_Type_Detector_Data_Set.csv
- There are 10000 entities in total . 5717 person entities and 4282 Company entities.
- Two columns: entity name, entity type(Person or Company)
- In multiple languages

Entity Name	Entity Type
Mudzumwe, Joubert	Person
Walter Joel Martinez Vasquez	Person
Zimmermann, Lena Maria	Person
Sim, Sambath	Person
泰納國際水果連鎖經營（北京）有限公司	Company
ချိုချိုသန်း	Person
Solarvat EOOD	Company
Vargas Rojas, Gerardo de los Angeles	Person
Cater Inn	Company
أحمد بن فهد بن سلمان بن عبد العزيز آل سعود	Person
Vicuña Veliz, Ronaldo Alberto	Person
Marinov, Yuriy Hristov	Person
PAMELA MARS	Person
China Railway First Group (Mongolia) Co.,Ltd	Company
အေးမြတ်မွန်	Person
Abu Shawish, Mohammed Mohammed Abul-Aziz	Person
Garcia da Veiga, Emanuel Antero	Person
210所	Company
Pasargad Fartak Aryan	Company
Matrix Internatinal Logistics, Inc.	Company
Ludwig Leung	Person
CÔNG TY CỔ PHẦN KINH DOANH VÀ PHÁT TRIỂN BÌNH ...	Company
Al Tahahwi, Ahmed Abdul Aal Bayoumi	Person
Emma Williams	Person
Guarded	Company
EGL Albania	Company

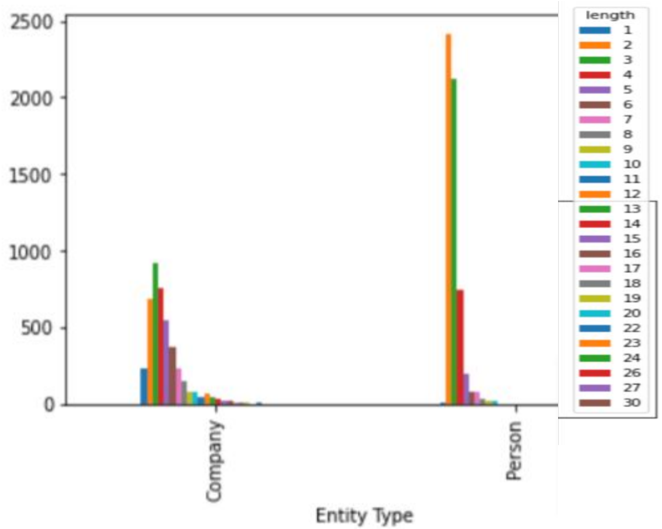
Feature Engineering - Language Detection

- Many iterations before we settled on alphabet detect
- Detects the alphabet being used NOT the language
- 17 separate alphabets
- We could see the split of different alphabets in the dataset
- This feature is critical to making sure many other features work effectively
- This also ensured we could split the data set properly

	Entity Name	Entity Type	langs_ad
6782	MƏMMƏDOV,Zaur	Person	LATIN
3690	ОШ МАМЛЕКЕТТИК УНИВЕРСИТЕТИ МЕКЕМЕСИ	Company	CYRILLIC
2494	Aeropuerto de Santa Isabel	Company	LATIN
132	KCRAM	Company	LATIN
5338	马瑞云	Person	CJK
6626	青柳真	Person	CJK
1956	Outer Islands Development Corporation (OIDC)	Company	LATIN
2131	National Life Insurance Company	Company	LATIN
6736	いむらひでや	Person	HIRAGANA
8803	Shihab Reza	Person	LATIN

LATIN	7867
CJK	924
CYRILLIC	308
HANGUL	277
ARABIC	256
DEVANAGARI	52
SINHALA	36
HEBREW	36
MYANMAR	34
ARMENIAN	33
THAI	33
GEORGIAN	33
GREEK	31
LAO	31
HIRAGANA	28
KATAKANA	13
KATAKANA-HIRAGANA	7

Feature Engineering- Length Feature

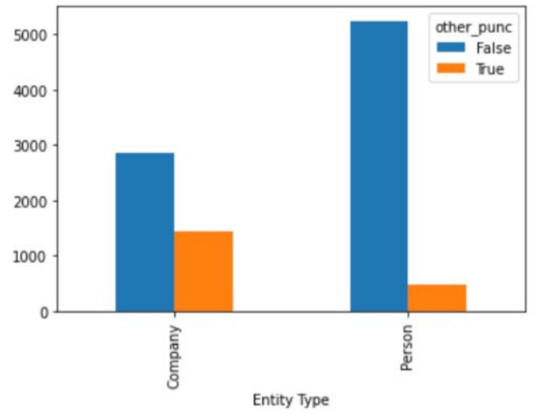


- This feature shows how long the Entity name is
  - We decided to measure length in words or characters depending on the language
- For people the most common length 2
- There are almost no people names with a length greater than 10
- This feature is helpful for determining if an entity is a person since there are some distinct characteristics of people entities in terms of the length of the entity

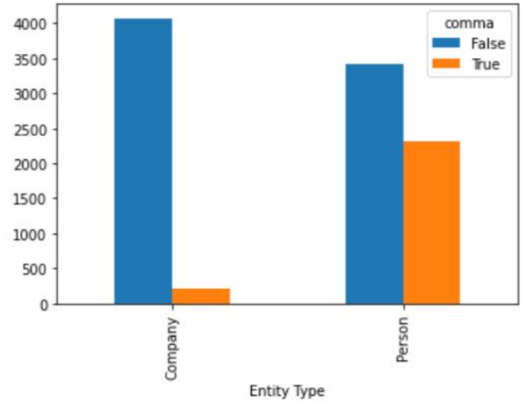
2317	JS Air	Company	LATIN	2
6171	李向东	Person	CJK	3
412	Santa Maria Energias Renováveis S.A.	Company	LATIN	6

Feature Engineering- Comma and other punctuation feature

- These features determine which entities have a comma present, and which have other punctuation
- Why do we make this distinction?
  - In latin languages commas appear much more frequently in name entities
  - Company names tend to have more other instances of punctuation (eg period or dash)
- As mentioned people and company entities have different patterns when it comes to punctuation appearing in the entity name which makes this feature helpful



Other punctuation



Comma

Feature Engineering- Contains Number Feature

John 99 Smith
Vs.
John Smith

- This feature identifies whether there is a number in the Entity Name.
- 83 entities in the dataset contain a number in the name
- This feature is beneficial because entities with numbers will most likely be companies.

Feature Engineering- Contains Company Suffix or Prefix Feature

John Smith Inc.
Vs.
John Smith

- This feature uses a list of company suffixes/prefixes, such as “Limited”, “Inc.”, “Co.”, in multiple languages to recognize company entities.
- 1887 entities in the dataset contain a company prefix or suffix in their name
- This feature is beneficial because it can help with the issue of a company name being a person’s name
- One of the most common words in our dataset was company suffixes and prefixes

Conjunction/Stopword Feature

	Entity Name	Entity Type	dup	conj_pres
624	Société Nationale d'Exploitation des Transport...	Company	False	1
5943	Amir Mansour Borghei	Person	False	0
5423	Cadmael Pech	Person	False	0
924	Aby Technical & Traning	Company	False	1

This feature identifies conjunction words from 39+ languages from within the entity

- spaCy Stopwords Package

The code inputs an entity name as a string and then outputs

➔ 1 if there is a conj. word

➔ 0 if there are no conj. words

Other Data Sets Used

Main_city.csv

- 3579 location names
- Mostly consist of names city names (Most commonly known cities around the world)
- All the countries
- US States Names
- Location names in a few other languages

창원시
천안시
청주시
춘천시
충주시
태백시
통영시
파주시
평택시
포천시

Korean

إستونيا
أثيوبيا
فيجي
فنلندا
فرنسا
الجابون
غامبيا
جورجيا

Arabic

Counties and US States

Virginia
Washington
West Virginia
Wisconsin
Wyoming
United States
Afghanistan
Albania
Algeria
Andorra
Angola
Antigua and Barbuda
Argentina
Armenia
Australia
Austria
Azerbaijan
Bahamas
Bahrain
Bangladesh
Barbados

2080 Cities

city_ascii
Tokyo
Jakarta
Delhi
Manila
Sao Paulo
Seoul
Mumbai
Shanghai
Mexico City
Guangzhou
Cairo
Beijing
New York
Kolkata
Moscow
Bangkok
Dhaka
Buenos Aires

Feature Engineering- Contains Location Name Feature

Entity Name	Entity Type	has_city_list2
Eni Gabon S.A.	Company	1
Jamaica Vacations	Company	1
National Taiwan University of Science and Tech...	Company	1
ALBA Alimentos de El Salvador	Company	1
Scotiabank Uruguay S.A.	Company	1
Airports Company South Africa	Company	1
Polyplas Dominicana - Grupo Diesco	Company	1
Kingstronic (Hong Kong)	Company	1
Africa Improved Foods Rwanda	Company	1
Industriji Elettronići Iran	Company	1
Institute of History of Academy of Sciences of...	Company	1
Bangladesh Municipal Development Fund	Company	1
Jordan Insurance Company P.L.C	Company	1
Trichem de Colombia S.A.	Company	1
Mario Salvador Pérez Fleites	Person	1
Telia Carrier Latvia SIA	Company	1

Why is this feature important?

- If an entity has a location name it is most likely a company
- Many companies around the world include the name of a city, country, state, etc.

Preview of the matches

- 721 matches
- Most entities classified are companies.

Feature Engineering- Contains Common Person name

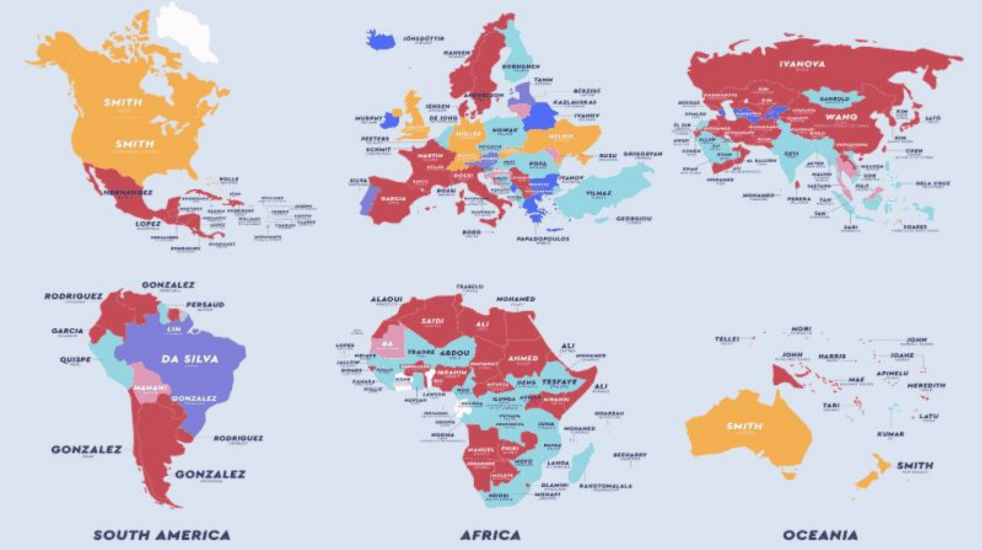
names.csv

- 1249 common names
- Used to classify an entity as a Person Name
- Names around the world

李
王
张
刘
陈
杨
赵
黄
周
吴
徐

David
Abdul
Ana
Ying
Michael
Juan
Anna
Mary
Jean
Robert
Daniel
Luis
Carlos
James
Antonio
Joseph
Elena
Francisco
Marie

Feature Engineering- Contains Common Person name



- This feature checks if an entity contains a common person name.
- It turns the names data frame into a list and goes through the list the to look for matches.
- Classified 3148 entities as having a common person name in it.
- It contains many different languages and names from around the world. So we can avoid bias.

Model Selection and Evaluation

Key Definitions

- **Machine Learning Model** : program that is trained to recognize patterns in data
- **Ensemble Methods:** Machine learning method that utilizes the predictions of many models to classify a new datapoint
- **Accuracy Metrics**
  - **AUC** : percentage of random points in your distribution your model properly classifies
  - **Precision:** Out of all your positive predictions, how many were actually correct?
  - **Recall:** Out of all the positives in the dataset, how many does our model capture?
  - **F1 Score:** Mathematical balance of precision and recall
- **Overfitting:** When a model fails to generalize well on new data because it pays too much attention to the particulars of the training dataset
- **Interpretable/transparent:** The level to which we understand how and why a model makes it's predictions

Algorithm Research and Selection

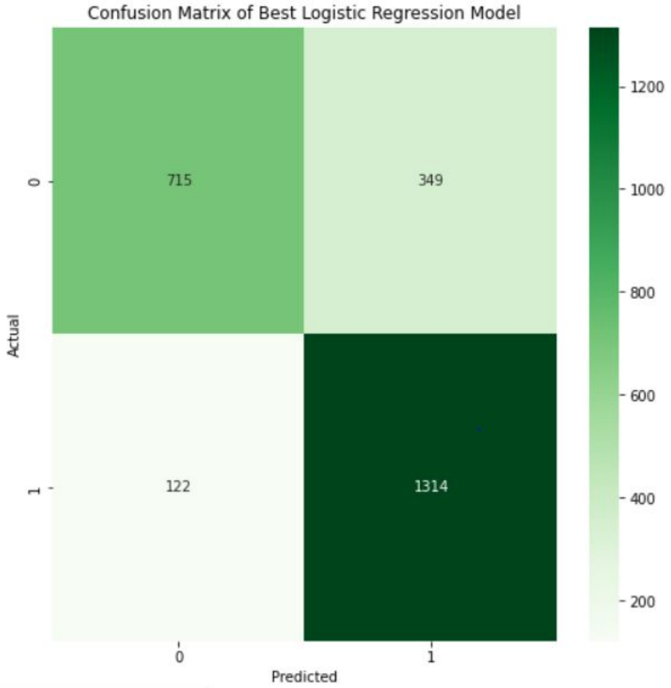
- Binary Classification Algorithms: identifies, out of two possible categories, which category an object belongs to
    - Logistic Regression
    - Gradient Boosted Descent
    - K-Nearest Neighbors (KNN)
    - Decision Tree
    - Random Forest
- Using GridSearch CV and Hyperopt to find the best parameters to improve these models
- We want precision, recall, and F1 score to be above 90%

Model Comparison

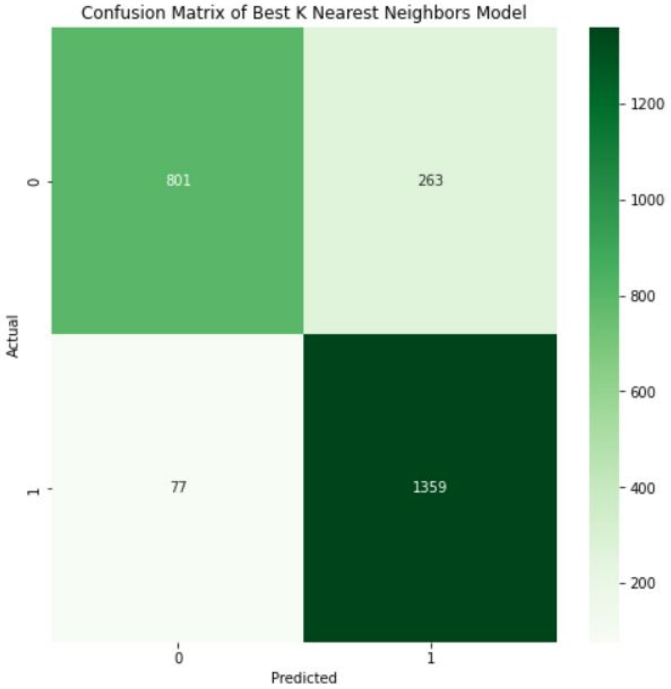
Model Name	Description	Results	Pros	Cons
K-Nearest Neighbors	Uses proximity to make classifications or predictions about the grouping of an individual data point	AUC = 92.8% Precision = 83.8% Recall = 94.6% F1 Score = 88.9%	➔ Simple ➔ It constantly evolves	➔ Can be slow with large datasets ➔ Dimensionality
Logistic Regression	Logistics regression uses a sigmoid function to return the probability of a label	AUC = 89% Precision = 79% Recall = 91.6% F1 Score = 84.8%	➔ Easy to implement ➔ Easy to update	➔ Sensitive to Outliers ➔ Overfitting
Gradient Boosted Decent	Trains simple models on the errors of previous models thereby having each new model focusing on the weaknesses of the previous iteration	AUC = 97% Precision = 91% Recall = 85% F1 Score = 88%	➔ No data preprocessing ➔ Flexible	➔ Less interpretable ➔ Overfitting ➔ Requires a lot space and time
Random Forest	Generates a group of decision trees and takes the majority vote to classify information	AUC = 96% Precision = 89% Recall = 83% F1 Score = 86%	➔ Does not tend to overfit ➔ Will adapt well to more features being added ➔ No scaling needed	➔ Less interpretable ➔ Slow with large datasets

Graphics For Top 2 Non-Ensemble Models

true-negitive: 715
false-positive: 349
false-negative: 122
true-positive: 1314

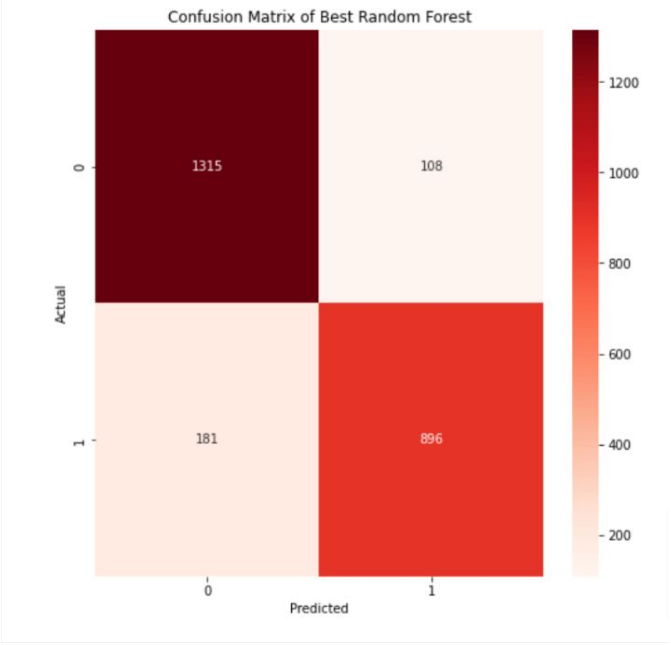


true-negitive: 801
false-positive: 263
false-negative: 77
true-positive: 1359

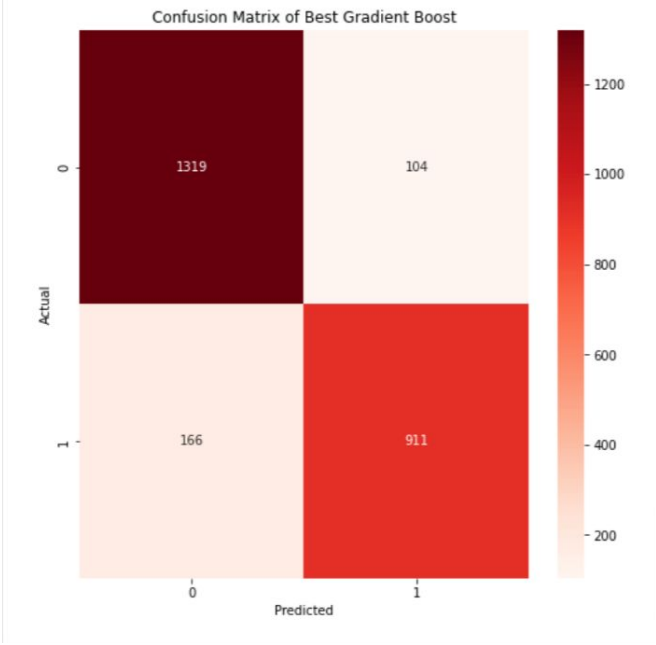


Graphics For Top 2 Ensemble Models

```
true-negitive: 1315
false-positive: 108
false-negative: 181
true-positive: 896
```




```
true-negitive: 1319
false-positive: 104
false-negative: 166
true-positive: 911
```



Feature Importance

- The features we developed were essential for our model's accuracy results
- Originally, precision, recall and f1 scores were unbalanced
- Adding two features, Common Locations and Common Person Name feature, balanced the scores by decreasing precision and increasing recall

```
1 :  has_co :  0.2804136353974996
2 :  comma :  0.1848945800923264
3 :  word_count :  0.1802361832772355
4 :  has_common_person_name :  0.14484488080740693
5 :  other_punc :  0.04171280646882276
6 :  CJK :  0.03498327178866768
7 :  conj_pres :  0.03402344934562214‌6
8 :  LATIN :  0.023757466586668708
9 :  HANGUL :  0.020493157686498134
10 :  has_city_list2 :  0.017873764344333073
11 :  CYRILLIC :  0.011554533771114183
12 :  ARABIC :  0.00455218680357851
13 :  has_digit_num :  0.004398011487520961
14 :  KATAKANA :  0.003832596133182033‌3
15 :  DEVANAGARI :  0.003103892972904334
16 :  GEORGIAN :  0.0018592493590370595
17 :  GREEK :  0.0016871448888725986
18 :  ARMENIAN :  0.0016023895693279035
19 :  SINHALA :  0.0009784497831248013
20 :  HEBREW :  0.0009315323696047536
21 :  MYANMAR :  0.0007158639955576728
22 :  THAI :  0.000568366471036491
23 :  HIRAGANA :  0.0005340316623733374
24 :  LAO :  0.0004485549376843315
25 :  MASCULINE :  0.0
26 :  KATAKANA-HIRAGANA :  0.0
```

Final Model Selection

- We decided to choose the K Nearest Neighbors Model as our selection due to it having a higher F1 and recall score
  - The Gradient Boosted Descent is our runner up due to it's high level of precision and the fact that it's F1 score is within .9 of the KNN model
  - It is also possible that given more time on hyperparameter optimization it could achieve better results as there are many untested hyperparameters

Final Thoughts

What We Learned

- We learned how to navigate through a multitude of different language groups with contrasting grammar, punctuation, and linguistics
- New hyperparameter optimization tools
- Some new material we learned:
  - Delved into feature engineering
  - Web-scraping
  - Natural language processing
  - API's
  - Language Detection

Potential Next Steps

- Analyze model errors to discern how to improve accuracy
- Concentrate on further developing + improving our features
- Further fine-tune and optimize model hyperparameters
- Aim for even higher accuracy, precision, f1 and recall scores