Intro Slide:

- Hello, and thank you for allowing the Labs20 iteration of the C4ADS Arms Exporters project to present to you today
- On behalf of Andrea Christelle, David Hang, Curtis McKendrick, and Andre Savkin, I, Ian Forrest, will be walking you through what we've been doing for the past couple of weeks

Slide 2 - Overview:

- Here's a quick breakdown of what we'll be discussing
- A review of Labs18's project (and why we chose to go in a different direction)
- Quick overview of the work we did in Labs 20 (which in a nutshell is a text analysis of the dataset's 'DESCRIPTION_GOOD' column)
- A Corporate Registry Data breakdown (which we didn't end up incorporating)
- We're going to talk about our Data Cleaning methods
- How we processed the Russian text and vectorized it for our model
- KMeans Modeling & Findings
- We're gunna talk about the Predictive Function we created & it's results
- And finally we'll discuss some Additional Testing and results from that testing

Slide 3 - Review of Labs 18 Project:

- Lab18's project has Good analysis and thought
- They created an artificial dataset using a known Russian arms exporters list as a reference, carried over from Labs 16:
  - If a column's value corresponded with that of a known arms exporter from the list, a value of '1' was generated
  - If a column's value didn't correspond with that of a known arms exporter from the list, a value of 0 was generated, as you can see in the image below
  - Long story short, they converted the dataset into a series of boolean indicators
- Then they performed machine learning analysis on the dataset of 1s and 0s
  - Labs18 obtained valid results in their list of new INNs
  - But they performed their analysis on a dataset of 1s and 0s, rather than on the underlying data
- There were also Issues with list of INNs:
  - We took a deep dive and found some inconsistencies
  - There were many INNs of large Russian arms exporters NOT on the list, for example 'JSC Kalashnikov Concern', one of the largest gun exporters, was missing from the list
  - Other INNs on the list had nothing to do with the arms industry
- After some discussion, Labs 20 decided to go in a different direction and

analyze the text within the ‘DESCRIPTION_GOOD’ column for our model

Slide 4 - Corporate Registry Data:

- Unfortunately we were not able to incorporate the Corporate Registry Data into our analysis
- As Labs 18 group discussed the data was messy, only 12% of the dataset had valid INN numbers
- Later on in our process we figured out a way to clean some of these INN numbers, but by that time Labs 20 had concluded

Slide 5 - Data Cleaning:

- Our data cleaning was divided into two phases:
  - PySpark/EMR Cleaning
  - Regular Python cleaning
- A PySpark SageMaker Notebook linked to an EMR cluster was the first phase
  - PySpark/EMR together were very useful in cutting down the size of the initial dataframe, which was too large for a regular Python notebook
    - Limited columns to Consignor Name, Consignor INN, and Description of Goods
    - Limited dataset to export trades only, as our project focuses on Russian exporters
  - We also had to remove certain punctuation marks, like semi-colons, from the ‘DESCRIPTION_GOOD’ column because they caused import errors in subsequent Python notebooks
- Cleaning the data frame in a Python3 notebook was the second phase
  - We created a Python dictionary of every valid Name-ID combination in the dataset and applied it to the entire dataset
  - This dictionary cleaning process saved over 177,000 rows of data that would have otherwise been dropped for having an invalid INN

Slide 6 - Russian Text Preprocessing:

- Our next step was pre-processing the text in the ‘DESCRIPTION_GOOD’ column to make it more useable for our model
- Done through processes of tokenization, which in simple terms means isolate each term in the ‘DESCRIPTION_GOOD’ column
- Preprocessing also removes punctuation marks and ‘stop words,’ words like ‘the’ and ‘it’ that shouldn’t be influencing the model
- Through the process of trial and error, we found our model was attributing value to trade-specific words. To reduce that influence, we added these words to the list of stop words for our final model

Slide 7 - Text Vectorization:

- Once preprocessed, we vectorized the text in the ‘DESCRIPTION_GOOD’

column
- Vectorization converts the text into numbers, where each number value corresponds to how our model should use a given word to define the description as a whole
- Zeros and low decimals indicate a word shouldn't be used to define a given trade description
- High decimals indicate a word can be used to define a given trade description
- For example in the image below, the Russian terms for 'PCS' & 'spike' have high decimal values in rows 1, 3 and 5, so those terms can be used by the model to define those particular trades
- There are 301 columns, defined by our vectorizer, that are used to map the entire training dataset of over 8 million rows

Slide 8 - KMeans Modeling & Findings

- We used a KMeans clustering model to group similar bodies of vectorized trade descriptions together
- Grouped them into a total of 10 clusters, and the groupings worked well
- The model did a good job of clustering trades together by industry. Below are some examples:sc
  - Cluster 0's trade descriptions are grouped together because of words like material, steel, metal, length, roll, and diameter. This likely indicates INNs whose trades fall into Cluster 0 are arms or large industrial companies
  - On the other hand, Cluster 4's trade descriptions are grouped together because of words like knitted, knitting machine, cotton, yarn, clothes, and luxury. This likely indicates INNs whose trades fall into Cluster 4 are fashion or textile companies
- We updated Lab 16's list of Known Russian Arms exporters, and we got some hits in our dataset
  - Around 76,000 trades in the training dataset were associated with INNs on our list
  - The large majority those trades fall into cluster 0, around 73% of them
  - descriptions associated with Clusters 2, 3, 4, 5, 7, 8, & 9 make up only 2.46% of known arms exporter trades in the dataset
  - THIS CLUSTER BREAKDOWN SERVED AS OUR 'PROFILE' FOR RUSSIAN ARMS EXPORTERS

Slide 9 - Predictive Function:

- Our product is a function that executes the entire process illustrated throughout this powerpoint
- It calculates the trade descriptions per cluster for every INN in the input dataset and converts them to percentages of the total trades, as you can see in the top image
- It then compares those percentages to the percentages for known Russian

arms exporters from the previous slide
- We created a score that measures the similarity between those percentages
  - If company's score is above a certain threshold, it's INN is included in the function's output dataset
  - If a company's score is below a certain threshold, it's INN is not included in the function's output dataset

Slide 10 - Predictive Function:
- Our score is measured using inverse Euclidian distance, which is a measure of similarity
- Because percentages for the smaller clusters were so low, our product's initial scores were artificially high for INNs that had little to no similarity to known arms exporters
  - For example, if a particular INN's trade descriptions were not grouped to the 'fashion' cluster, it was rewarded incorrectly with a high score
- Because of this fact, our product conducts its final Euclidian similarity score using only the large clusters, as seen in the bottom image

Slide 11 - Results:
- We ran our function on a test dataset with over 2 million rows with great success!
- Our product narrowed the 32,195 unique INNs in the dataset down to just under 3,000 INNs

Slide 12 - Additional Testing:
- We wanted to test our product's strength on an unrelated dataset with different columns, and found the Ukrainian trade dataset in one of the S3 buckets
- The results were very positive! Our product narrowed a 7.3 million row trade dataset down to 182 EDRPOUs of interest (which is the Ukrainian Tax ID)

Slide 13 - Conclusions:
- Some tinkering can be done to the product
  - Too many trade descriptions in the training data were grouped in Cluster 0, giving it far too much weight for our final model. Future groups would find value in breaking Cluster 0 apart, as it would create a better profile for Known Russian Arms exporters
  - Allowing the vectorizer to be more robust would increase the predictive abilities of the model as well
    - If our model trained on 1200 columns instead of 301, it would provide more accurate results
  - We also recommend expanding the list of known arms exporters to improve the strength of the model

    - Expand list of known non-exporters to improve the product's performance
        - Less computing power required if fewer INNs need to be processed
- THANK YOU!