Mastering Data with Spark and ML

Strata London 2019
About Me

IIT Delhi, 1998

Founder and CEO, Nube Technologies

Strata Data San Jose Program Committee

Speaker at Spark Summit, Strata, GIDS etc
Nube

India based startup

Deep technical problems with an enterprise solution

ML, Big Data, UX
This talk today

Problem Statement

Our Approach
Simple business asks

Customer LTV

Best supplier for a part

Supplier payment terms

Householding

Cross Sell Opportunities

M&A
### Actual Data

<table>
<thead>
<tr>
<th>Company</th>
<th>Name</th>
<th>Address 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>3C Incorporated</td>
<td>Markus Lau Ho</td>
<td>C2-002 Armour Plan</td>
</tr>
<tr>
<td>3C Inc</td>
<td>Markus Ho</td>
<td>C2/002 Armour Plan</td>
</tr>
<tr>
<td>3 C</td>
<td>Mr. Markus Ho</td>
<td>Armour Plan C2/002</td>
</tr>
</tbody>
</table>

- **Leading spaces**
- **Additional Middle Name**
- **Backslash separator**
- **Missing suffix**
- **Prefix/Salutation**
- **Words swapped**
WE HAVE INCOMPLETE DATA, SO I’LL NEED TO USE MY INTUITION AND EXPERIENCE TO MAKE THE DECISION.

BECAUSE MAGICAL THINKING FIXES IGNORANCE?

HUSH! I THINK I HAVE SOMETHING.

I THINK SO TOO.
Actual data

Silos

Data Quality

Volumes
Challenges

Variety of sources

Scale

Capturing rules for matching and merging

Working across different business entities
Wishlist

Any source and format

Any entity type

Any volume
Reifier

AI powered data management, matching and merging different data sources to build a holistic view.

- MDM
- Fraud and Analytics
- Sales and Marketing
- Customer AML/KYC/cross and Upsell
- Data Enrichment
- Reference data Management
- Data Quality
Our stack
Wishlist

Any source and format

Any entity type

Any volume
Any source and format

Based on RDDs

Custom source and sink formats written by us/borrowed from community
Any source/sink, Any format

Elastic:

```scala
import org.elasticsearch.spark.rdd.EsSpark

EsSpark.saveToEs(rdd, "spark/docs")
```

Cassandra:

```scala
val rdd = sc.cassandraTable("test", "kv")

rdd.saveToCassandra("test", "kv", SomeColumns("key", "value"))
```
Problems with RDDs

Record wise reading was good, but adding structure to the data was left to us.

reifier.Tuple - indexed data structure

Development and maintenance nightmare
Reifier 2.0

- Datasets
- Pipe abstraction

```java
public class Pipe implements Serializable{

    String name;

    Format format;

    Map<String, String> props;

    StructType schema;
}
```
Building Dataset through Pipe

```java
DataFrameReader reader = spark.read();

reader = reader.format(p.getFormat().type());

reader = reader.schema(p.getSchema());

for (String key: p.getProps().keySet()) {
    reader = reader.option(key, p.get(key));
}

Dataset<Row> input = reader.load();
```
Spark Integration

Tried Livy etc

Additional dependency

Finally two ways in which we integrate. One local SparkContext. Second through the SparkLauncher.
<table>
<thead>
<tr>
<th>r_id</th>
<th>firstname</th>
<th>lastname</th>
<th>streetnumber</th>
<th>street</th>
<th>area</th>
<th>city</th>
<th>areacode</th>
<th>statecode</th>
<th>dob</th>
<th>ssn</th>
</tr>
</thead>
<tbody>
<tr>
<td>rec-7880-org</td>
<td>liam</td>
<td>berry</td>
<td>12</td>
<td>learmouth drive</td>
<td>lynden park stud</td>
<td>broken hill</td>
<td>6014</td>
<td>vic</td>
<td>19271005</td>
<td>5315721</td>
</tr>
<tr>
<td>rec-54255-org</td>
<td>felicity</td>
<td>stephenson</td>
<td>11</td>
<td>the rookery</td>
<td>duncraig</td>
<td></td>
<td>3012</td>
<td>qld</td>
<td>19531111</td>
<td>3765432</td>
</tr>
<tr>
<td>rec-24540-org</td>
<td>bethany</td>
<td>tu</td>
<td>8</td>
<td>antill street</td>
<td>longstay caravan park</td>
<td>upper brookfield</td>
<td>4211</td>
<td>tas</td>
<td>19781230</td>
<td>1345472</td>
</tr>
<tr>
<td>rec-61144-org</td>
<td>anita</td>
<td>lutz</td>
<td>9</td>
<td>adinda street</td>
<td>brentwood village</td>
<td>bokarina</td>
<td>3806</td>
<td>qld</td>
<td>19021023</td>
<td>9192121</td>
</tr>
<tr>
<td>rec-76431-org</td>
<td>amber</td>
<td>white</td>
<td>41</td>
<td>oxley street</td>
<td>quarter deck</td>
<td>toowoomba</td>
<td>4169</td>
<td>vic</td>
<td>19351010</td>
<td>1468300</td>
</tr>
<tr>
<td>rec-96533-org</td>
<td>michael</td>
<td>webb</td>
<td>2</td>
<td>lyster place</td>
<td></td>
<td>glenella</td>
<td>2572</td>
<td>sa</td>
<td>19210224</td>
<td>8072876</td>
</tr>
<tr>
<td>rec-54907-org</td>
<td>mikaela</td>
<td>wiseman</td>
<td>71</td>
<td>littler place</td>
<td>killarney park</td>
<td>morphettville</td>
<td>2774</td>
<td>nsw</td>
<td>19351117</td>
<td>9810844</td>
</tr>
<tr>
<td>rec-95638-org</td>
<td>ryley</td>
<td>george</td>
<td>10</td>
<td>araluin street</td>
<td>hopefield</td>
<td>alice springs</td>
<td>5043</td>
<td>vic</td>
<td>19341223</td>
<td>1186509</td>
</tr>
<tr>
<td>rec-33506-org</td>
<td>isabelle</td>
<td>coffey</td>
<td>3</td>
<td>michie street</td>
<td>tathra</td>
<td>chisholm</td>
<td>4216</td>
<td>qld</td>
<td>7934987</td>
<td></td>
</tr>
<tr>
<td>rec-26532-org</td>
<td>danielle</td>
<td>pettingill</td>
<td>5</td>
<td>narrabundah lane</td>
<td>ash-mere</td>
<td>wy yung</td>
<td>2074</td>
<td></td>
<td>19201019</td>
<td>4108963</td>
</tr>
<tr>
<td>Date</td>
<td>Log</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:46:52 PM</td>
<td>Matching has completed successfully</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:46:51 PM</td>
<td>Finished writing matching records</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:46:49 PM</td>
<td>Start writing duplicates ...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:35:33 PM</td>
<td>Starting matching...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:35:32 PM</td>
<td>Saved similarity rules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:35:32 PM</td>
<td>Finished learning similarity rules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:25 PM</td>
<td>Indexing rules learnt</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:19 PM</td>
<td>Learning indexing rules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:13 PM</td>
<td>Prepared job</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:13 PM</td>
<td>Initializing learning similarity rules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:08 PM</td>
<td>Initializing job for matching</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:08 PM</td>
<td>Read job arguments</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug 24, 2015 12:34:08 PM</td>
<td>Preparing job</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Wishlist

Any source and format
Any entity type
Any volume
Any entity type

- Traditional rule based system fails
- AI to the rescue
- Also Cassandra
Do the following pairs match?

<table>
<thead>
<tr>
<th>Brooklyn</th>
<th>Blake</th>
<th>63</th>
<th>O'Connell Street</th>
<th>Eden Hills</th>
<th>810</th>
<th>Vic</th>
<th>19101201</th>
<th>2258884</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brooklyn</td>
<td>Jehne</td>
<td>2</td>
<td>Dumasq Street</td>
<td>Lamington</td>
<td>6061</td>
<td>WA</td>
<td>19420723</td>
<td>2935590</td>
</tr>
<tr>
<td>Caitlin</td>
<td>Noack</td>
<td>28</td>
<td>Hamilton Row</td>
<td>Sec 375</td>
<td>7011</td>
<td>Vic</td>
<td>8328282</td>
<td></td>
</tr>
<tr>
<td>Caitlin</td>
<td>Westbrook</td>
<td>631</td>
<td>Casson Street</td>
<td>Blacksdale Private Sanctuary Gidgegannup</td>
<td>3114</td>
<td>NSW</td>
<td>19360908</td>
<td>9439473</td>
</tr>
<tr>
<td>Match 10/35</td>
<td>Non Match 24/35</td>
<td>Unsure 1/35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>isabella</td>
<td>ryan</td>
<td>keira street</td>
<td>pyndari</td>
<td>toowoomba 3500 vic 19400405 6806487</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>isabella</td>
<td>hingston 57</td>
<td>loureiro street</td>
<td>thornlie 3500 nsw 19290125 4688367</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nicholas</td>
<td>wilkey 12</td>
<td>caladena street</td>
<td>rivonia</td>
<td>kingston 2480 wa 1399646</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nicholas</td>
<td>wickel 12</td>
<td>jardine street</td>
<td>wuikikakiwi</td>
<td>highton 3085 wa 19570620 2126121</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Any scale

Add Spark to the mix

Ouch, cartesian join - 1million records = Order of a trillion comparisons

Learn what to join
AutoML

Build multiple models based on the training data

Optimize for accuracy and performance

Use Spark to train and assess different models
Cassandra

Any Entity

Any Scale
Cassandra Training

Primary Key - Cluster Id, Record Id

Secondary Index - r_isMatch
Cassandra Entity

Primary Key - Record Id

Secondary Index - Cluster Id
Elastic

Free flowing search

Adhoc analytics

Realtime Plugin
Thank You!

www.nubetech.co

sonal@nubetech.co