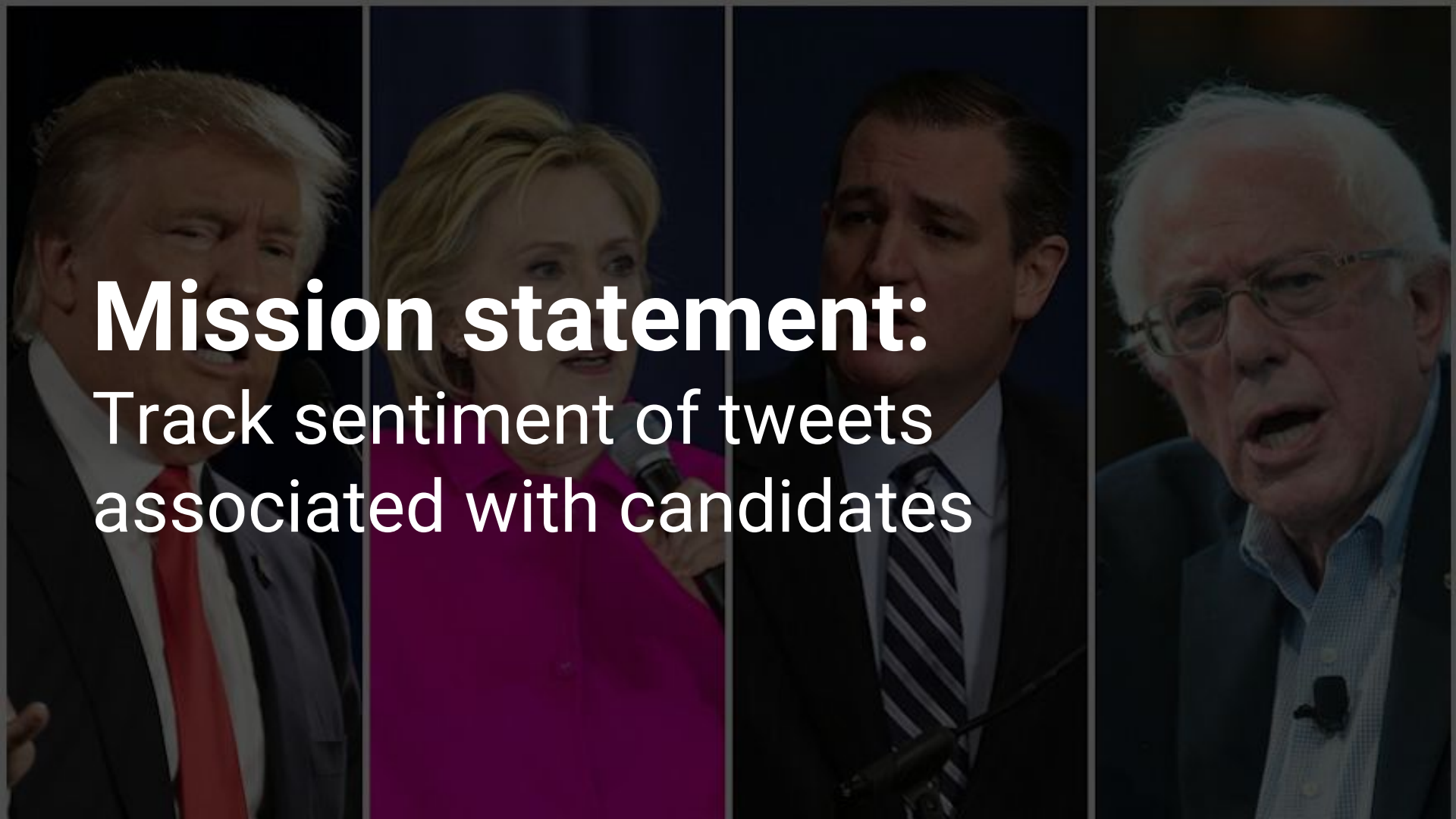


QUITTER/ TWITTICAL

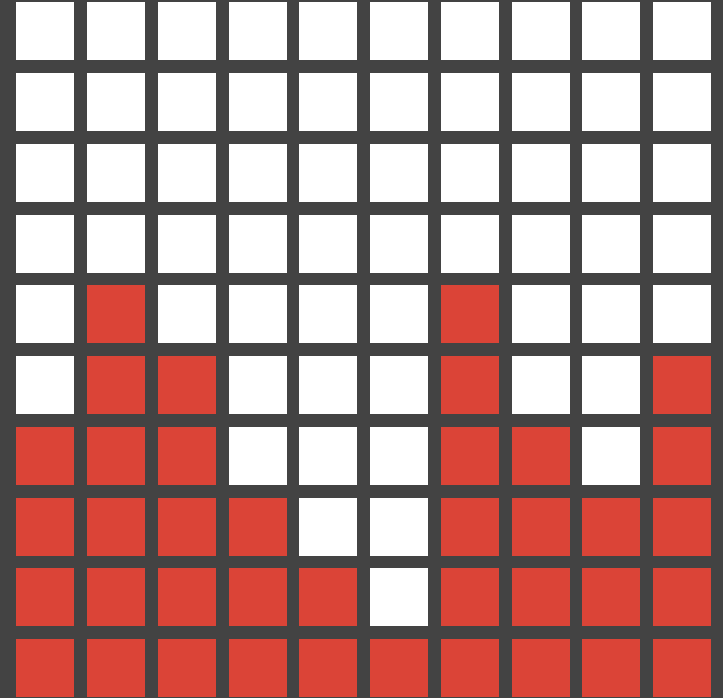
Rishab Kanwal, Bryce Strickland, Loic Guegan, Ahmed Almutawa, Nika Shafranov

A collage of four political figures: Donald Trump, Hillary Clinton, Marco Rubio, and Bernie Sanders. The image is divided into four vertical panels, each featuring a different candidate. Overlaid on the center of the collage is the text 'Mission statement: Track sentiment of tweets associated with candidates'.

Mission statement:

Track sentiment of tweets
associated with candidates

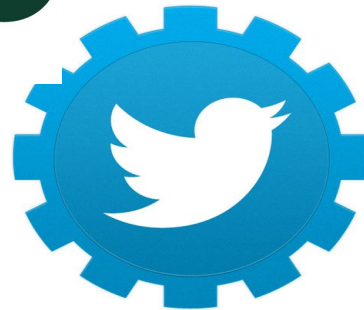
A look at the varying sentiments over the past few months pertaining to presidential candidates



Technology Stack



django



Architectural Pattern - MVC (AngularJS)

- The View is defined in HTML, while the model and controller are implemented in JavaScript
- HTML templates with a ng-controller directive specifying a controller allow it to bind data from the model
- Factory methods allowed for asynchronous calls to grab JSON to use for our model
- Decouples view from model, controller binds to HTML

AngularJS Example

Binding in HTML:

```
<h3>Latest Tweets For {{chartTitle}}</h3>
```

Declaration in controller:

```
$scope.chartTitle = Barack Obama
```

Displays → Latest Tweets For Barack Obama



Django + Cassandra + REST API

The Django logo, featuring the word "django" in a dark green, lowercase, sans-serif font.

+



Abstract Factory - Django Models

- Way to encapsulate a group of individual factories with a common theme without specifying their concrete classes
- A factory is the location of a concrete class in the code of a concrete class in the code at which objects are constructed

Abstract Factory - Django Models

```
class Candidate(Model):
```

```
    __abstract__ = True
```

```
class Bernie(Candidate):
```

```
class Cruz(Candidate):
```

```
class Hillary(Candidate):
```

CNN Model Trainer

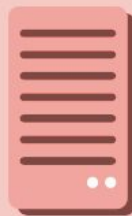
- Employs CNN feeding LSTM into an RNN language model. Brain.
- Trained on 200,000 tagged tweets. Accuracy: 84%. Humans can only agree on sentiment 80% of the time
- Very modular design: High cohesion, Low coupling. Adapter design pattern
- Allowed us to reuse a lot of code and identify bottlenecks quickly
- Created clusters of identical machines that could be threaded efficiently

Takeaways

Creating an object oriented REST API for our project worked well due to our roles

- querying database and formatting JSON data
- developing front end

BACK-END



FRONT-END



Modular design allowed for seamless integration with no overlap