

# Text Extraction

FinTech  
Lesson 12.3



# Class Objectives

---

By the end of the class, you will be able to:



Understand spaCy capabilities and where to find documentation.



Be able to use POS tagged text to extract specific words.



Use dependency parsed text to extract descriptors.




Extract specific types of entities from text.



Correlate text features to real-world series like stock prices.



Create a dashboard from NLP sentiment features.

The background of the slide is a dark, almost black, field filled with a complex, repeating geometric pattern of triangles and squares in various shades of gray, creating a textured, low-poly effect. The word "spaCy" is centered in the middle of the slide in a clean, white, sans-serif typeface.

spaCy

## spaCy

- Core functions depend on language models learned from tagged text
- Fast and flexible
- Designed specifically for production use

## NLTK

- Core functions depend on language models learned from programmed rules
- Accurate
- Intended for educational and prototyping purposes

# spaCy

---

We will be using spaCy for:



Part of speech tagging



Named entity recognition



Dependency parsing



These tasks are more suitable for **model-based solutions** because they are complex and depend highly on context.

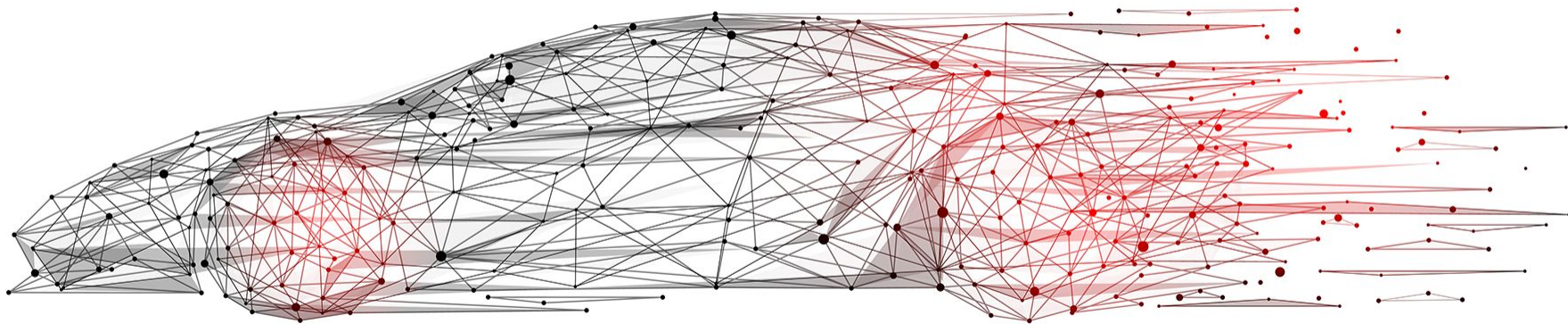


**spaCy** also provides tools for tasks like **tokenization** and **lemmatization**, which we've already learned with NLTK, and creating word vectors.

# spaCy

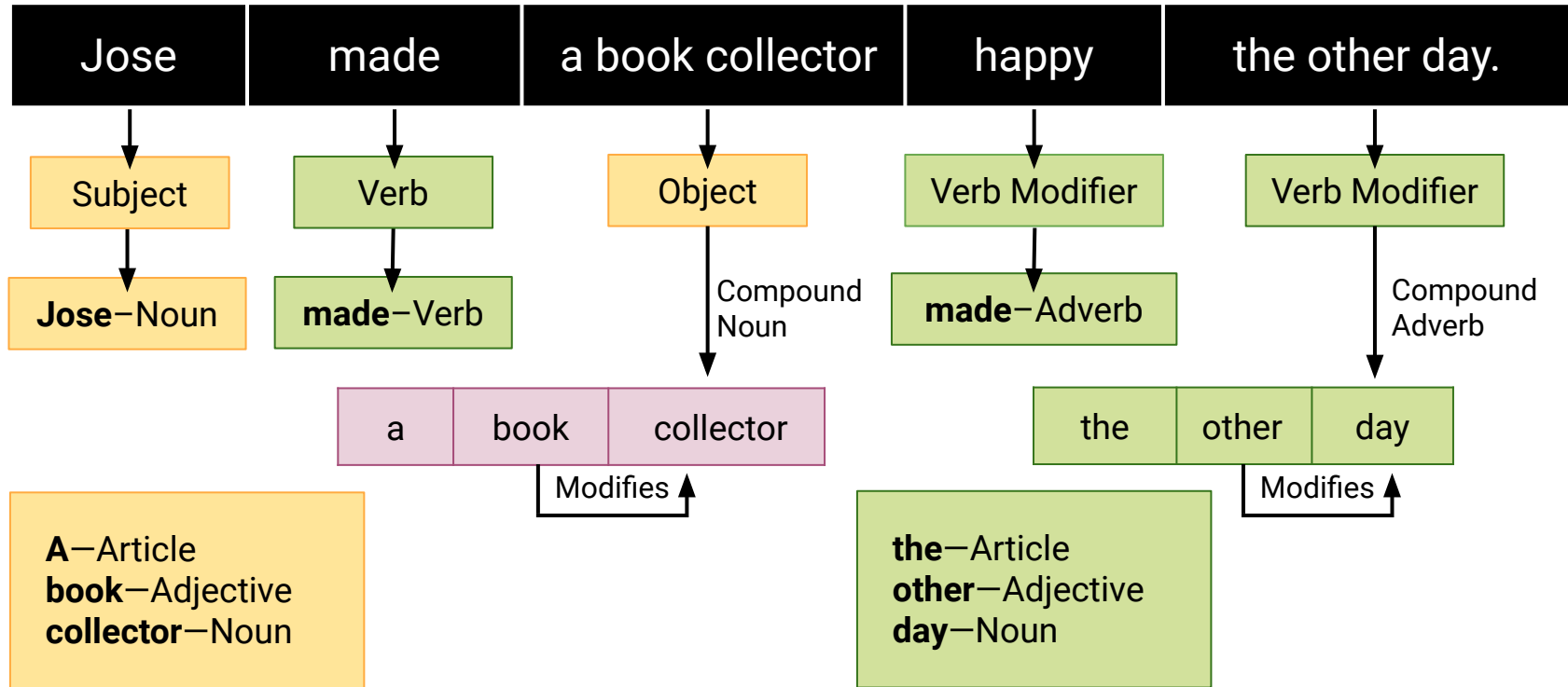
---

In comparison to NLTK, spaCy's language models trades off accuracy for speed, so if the corpus is large then you may prefer a simpler, rule-based solution.

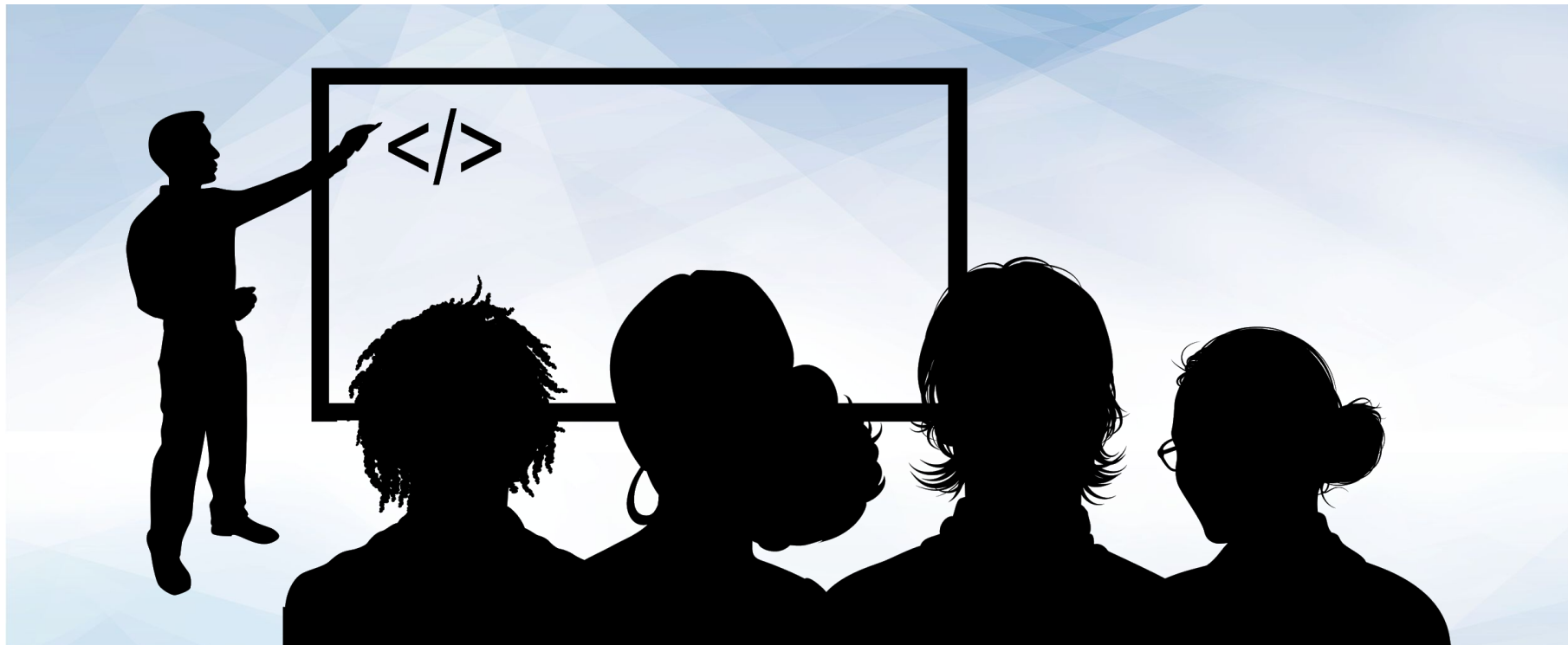


# Part of speech tagging

Categorizing each word in a sentence by its grammatical role in the sentence.







# Instructor Demonstration

## POS Tagging and Dependency Parsing



## **Activity:** Describing America

In this activity you will use the inaugural address corpus from NLTK and spacy's parsing and tagging modules to analyze the text that presidents have used to describe America.

**Suggested Time:**  
15 minutes





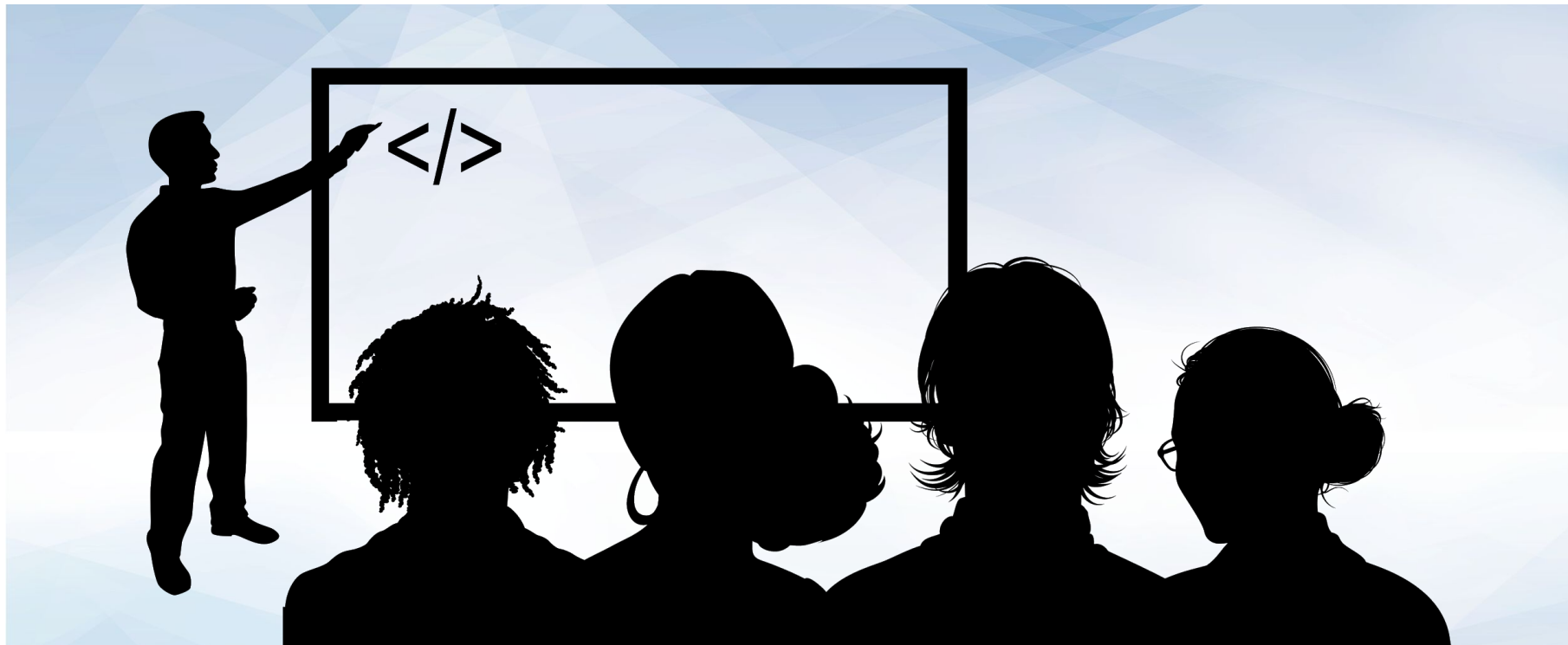
**Time's Up!** Let's Review.

# Named Entity Recognition

---

Extracting named entities, which include proper nouns and other specific types of nouns such as currencies, from a text.

**US GPE** unveils world's most powerful supercomputer, beats **China GPE**. The **US GPE** has unveiled the world's most powerful supercomputer called 'Summit', beating the previous record-holder **China GPE**'s **Sunway TaighuLight ORG**. With a peak performance of **200,000 CARDINAL** trillion calculations per **Second ORDINAL**, it is over twice as fast as **Sunway TaighuLight ORG**, which is capable of **93,000 CARDINAL** Trillion calculations per second. Summit has **4,608 CARDINAL** servers, which reportedly take up the size of **two CARDINAL** tennis courts.



# Instructor Demonstration

## Named Entity Recognition



## **Activity:** NER Clouds

In this activity you will extract named entities of their own choosing from the Reuters corpus and build a wordcloud from these entities.

**Suggested Time:**  
15 minutes





**Time's Up!** Let's Review.



A close-up photograph of a computer keyboard. The central focus is a large, white, rectangular key with rounded corners. On this key, there is a dark blue icon of a coffee cup with three wavy lines above it representing steam. Below the icon, the word "Break" is printed in a dark blue, serif font. The key is set against a light-colored keyboard frame. Surrounding the main key are other keys: to the left is a key with double quotation marks, above it is a key with a right square bracket, and to the right is a key with a left square bracket. The lighting is soft and even, highlighting the texture of the keys.

Break



# Text as Feature

# Tools and Techniques

---

Tools and techniques used to create numerical features (structured data) from text (unstructured data):

## Tools

- NLTK
- wordcloud
- spaCy

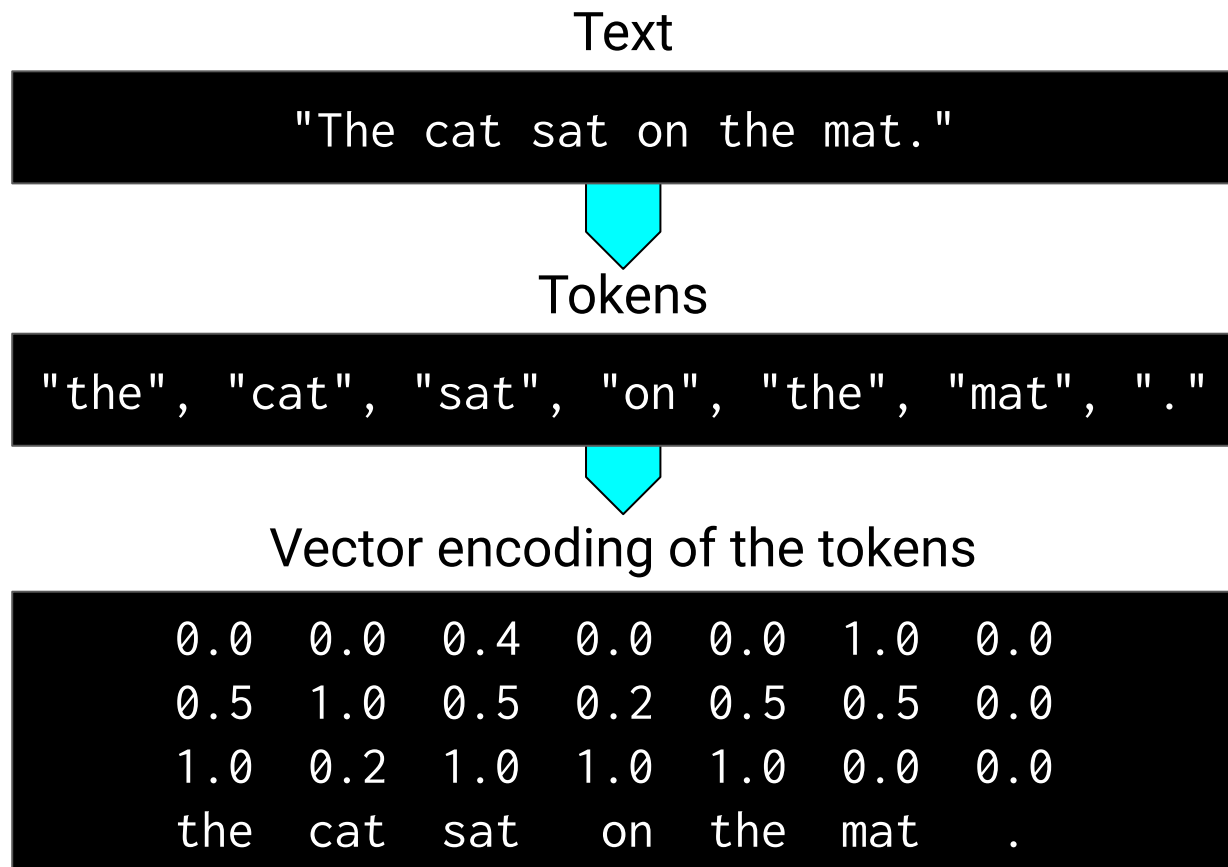


## Techniques

- preprocessing
- tokenizing text
- lemmatizing text
- aggregating word counts
- creating n-grams
- normalizing to tf-idf weights
- sentiment analysis
- parsing and pos-tagging text
- named entity recognition

# Text as Feature

In order to use this data for classification or prediction, we need to make them features—numerical representations of unstructured text.





**What are some examples of features that can be created from text documents?**

# Text as Feature

---

Examples:



count of a keyword



count of named entities



sentiment scores



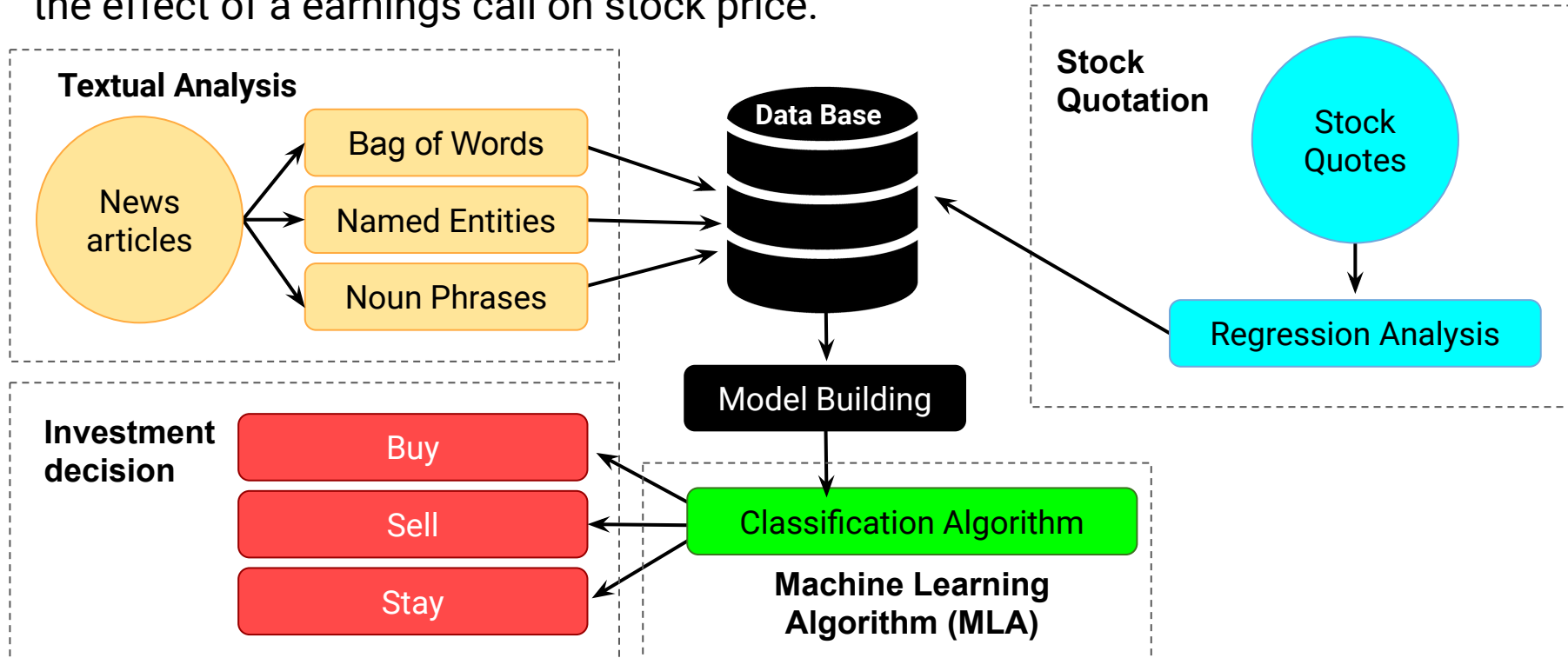
percent of words that are adjectives



total tf-idf score

# Text as Feature

These features can then be used to classify a document to a category or predict the effect of a earnings call on stock price.





## **Activity:** Correlating Returns

In this activity you will create a sentiment index from newsapi headlines and correlated it to S&P 500 daily returns, looking for text topic that generates the highest correlation.

**Suggested Time:**  
15 minutes





**Time's Up!** Let's Review.





## Challenge:

# Crisis Analysis Dashboard

In this challenge you will use your new sentiment analysis skills in combination with some of the skills you've already mastered, such as: Pandas, PyViz, Plotly Express and PyViz Panel to create a data visualization dashboard.

**Suggested Time:**  
40 minutes





**Time's Up!** Let's Review.



Questions?