



## Tech Stack

### Languages

Python  
Node.js  
jQuery

### APIs

ClaimBuster  
Facebook  
Slack  
AWS

### Amazon Web Services

Lambda  
Redshift  
DynamoDB  
CloudWatch  
Step Functions

## Abstract.

Tech & Check Alerts is an automated service that helps fact-checkers find claims to check.

1 SCRAPE

In the first step of our pipeline, we parse news transcripts, organizing claims and metadata into useful data structures. We currently scrape CNN transcripts, NBC transcripts for Meet the Press, politicians' Facebook accounts, and the Congressional Record.

2 SCORE

In the next step, we "score" claims from the transcripts using ClaimBuster, a natural language processing API that returns the probability that a given quote is a factual claim (worth checking).

3 STORE

We store ratings and other metadata using Amazon Redshift. Persisting the data allows us to create archival resources for journalists and fact-checkers.



SHARE

The sole goal of our project is to make a service that journalists and fact-checkers find useful. Finding a way to transmit data effectively is therefore, paramount. We currently send daily emails to journalist and fact checkers that include the top 15 most fact-checkable claims of the day we've recorded, and we're in the process of creating an interactive Slack bot that will allow them to access our database at their leisure.

## Results

So far, *The Washington Post*, *PolitiFact* and *FactCheck.org* have published **7** articles based on quotes they first saw in our daily alerts. As we increase our sources and distribution list, we expect that number to increase.

## Future

In addition to **increasing the amount of sources** we scrape, we hope to begin gathering feedback from fact checkers on what kinds of claims they find useful to incorporate into ClaimBuster's recurrent model. We also intend **match new claims with related statements** from previously published fact-checks using the Reporters' Lab in its Share the Facts database. Furthermore, we hope to increase the quality of our own products by **building recurrent entity recognition models** that result in clearer speaker IDs.

## Acknowledgements

We would like to thank everyone in the *Duke Reporter's Lab*, *DeWitt Wallace Center for Media & Democracy*, and *Tech & Check Cooperative* who has worked with us.

In addition, we're immensely grateful to *Chengkai Li* of UT-Arlington, who created the ClaimBuster product our service is based around, and to our sponsors who allow us to continue helping journalists around the country.