Detecting Textual Saliency in Privacy Policy

Phase-2 Progress Updates

Kaushik Nishtala ■ Shubhanshu Gupta

Goal

Provide key information to end-users using Machine Learning and Natural Language Processing

INFORM
Module

- Visualization
- Annotating policies

QUERY
Module

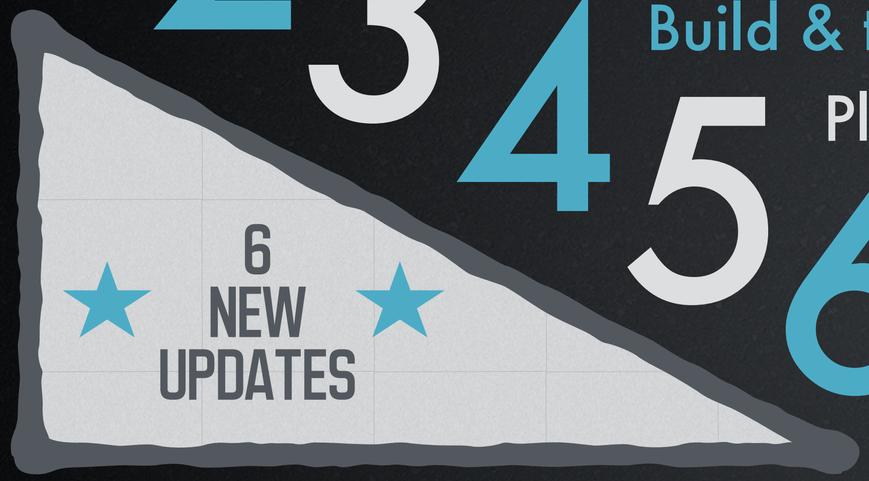
- Enable users to ask policy-related

Interactive

Web Framework

Progress Updates

1. Lay down methods to scale up data and model complexities
2. Train Domain-specific word embeddings (in-progress)
3. Semantic segmentation of raw privacy policy texts
4. Build & train language neural architectures
5. Plan the pipeline for QUERY module
6. Interactive web interface (v1.0)



★ Major focus on the implementation of web interface

Roadblocks

Data and model scalability issues

Semantic segmentation

Building custom BERT architectures for classification and QA

Build, deploy and integrate the model stack with the web interface

Methodology
(Phase-2)

- ★ Reliable predictions from baselines
- ★ Hard to predict classes with lower counts of positive examples
- ★ NEXT: Scale up data and model complexities
- ★ Understand strengths and weaknesses of each model in predicting a category
- ★ Leverage the expertise of each learner to perform predictions on unseen data

Scaling Up Data Complexity

(Domain-Specific Non-Contextual Embeddings)

- ★ Limited labeled data => Feed unlabeled data first, then increase model complexity
- ★ Pre-trained generic embeddings showed very little performance improvement
- ★ Previous literature recommends training domain-specific embeddings, if possible

Dataset

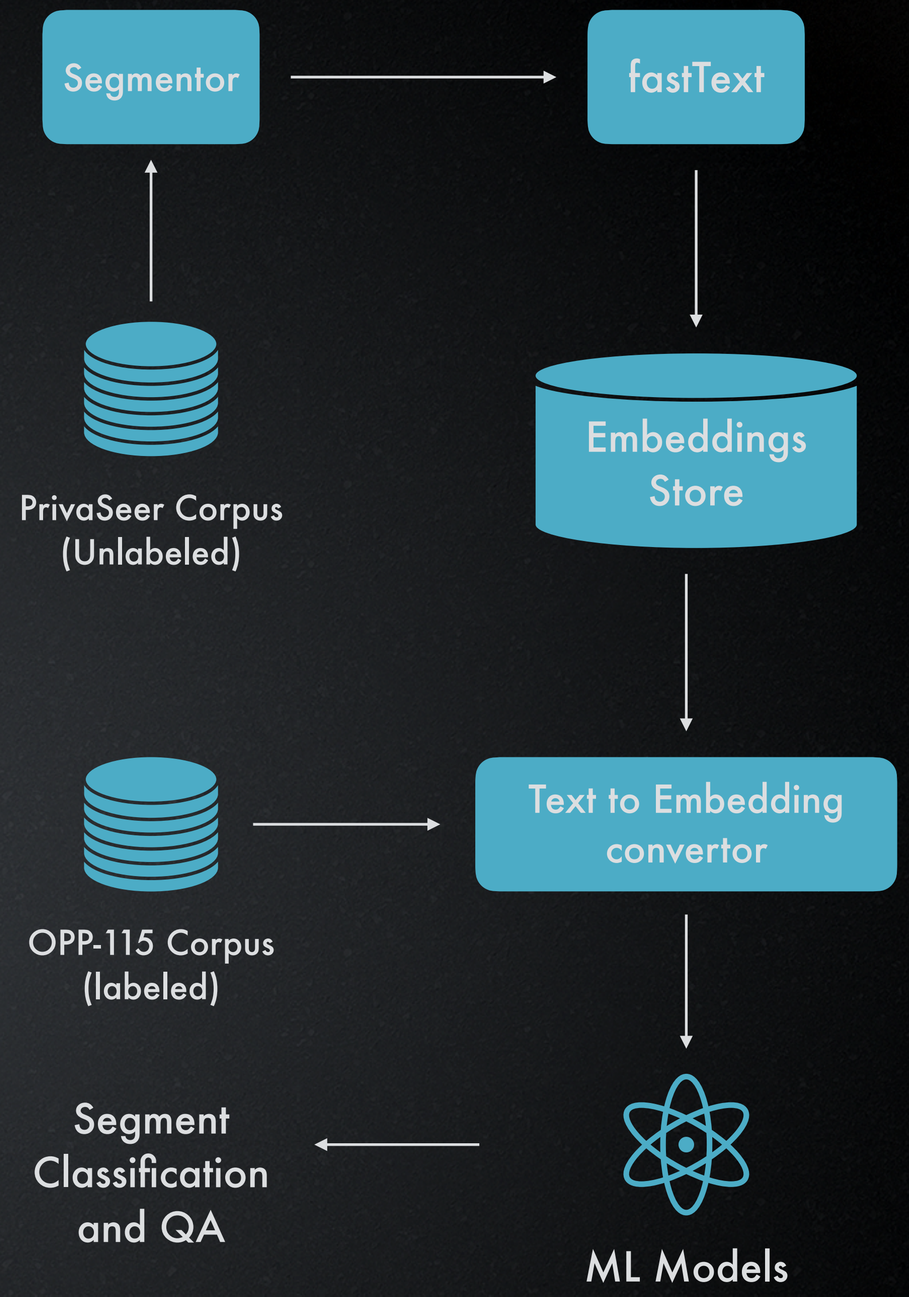
- ★ Obtained access to 1M raw privacy policies dump (PrivaSeer Corpus)
- ★ May not need all 1M policies (performance may saturate sooner)

Domain-Specific Non-Contextual Embeddings
(Process)

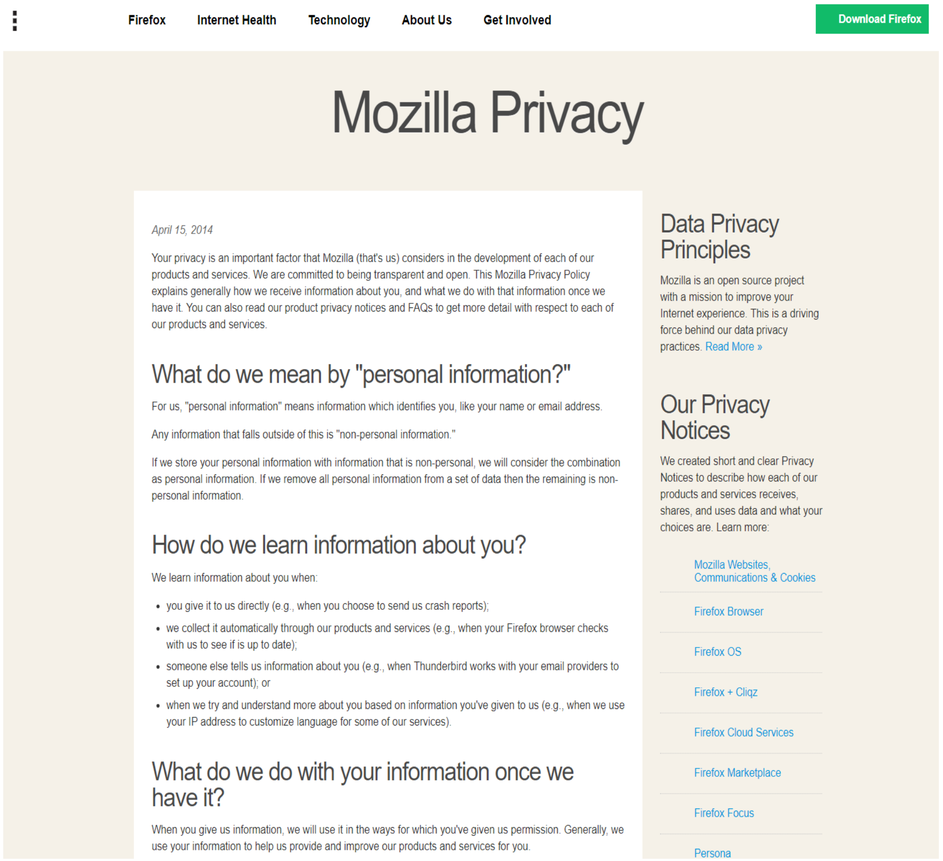
- Significant time spent on segmentation, fastText and handling the scale of the corpus (still in-progress)
- Automatic Segment Detection using Unsupervised and Supervised Learning (ASDUS)

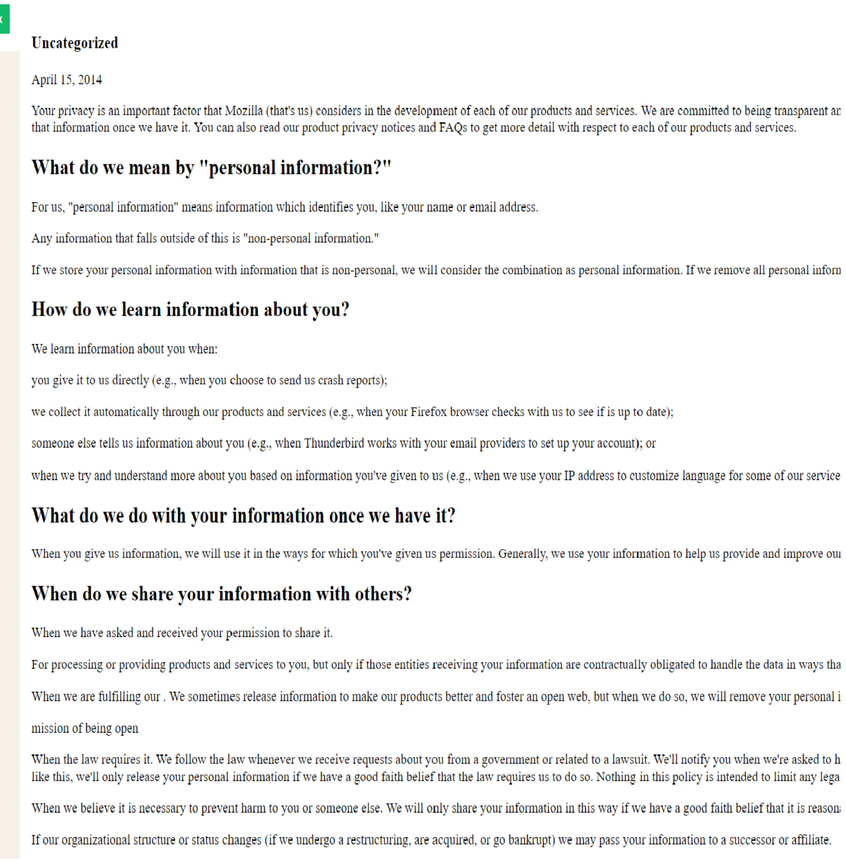
fastText

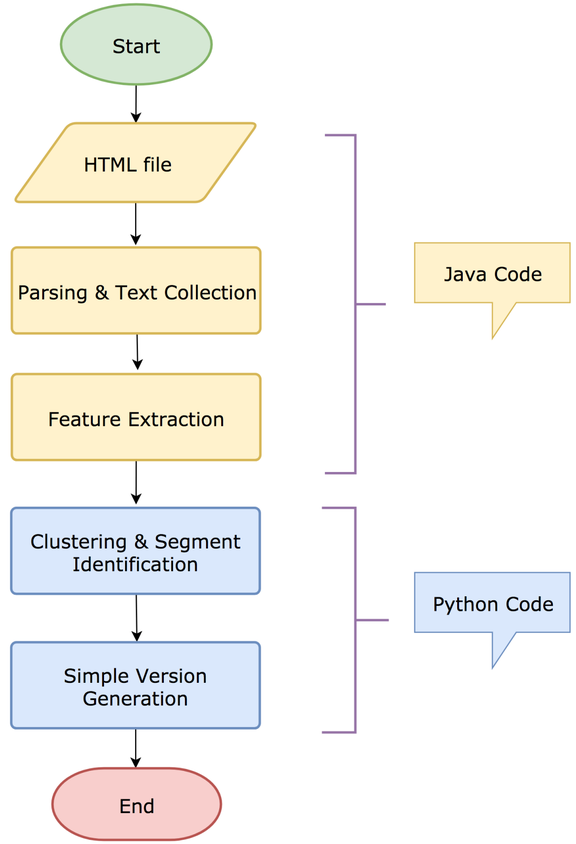
- Open-source library that allows users to learn supervised and unsupervised text representations and text classifiers.
- Supports both continuous bag of words and the Skip-gram model
- Performs better than Word2Vec and GloVe - uses character level information to handle OOV or rare words.
- Our objective: train semantic embeddings for downstream tasks



Automatic Segment Detection using Unsupervised and Supervised Learning (ASDUS)



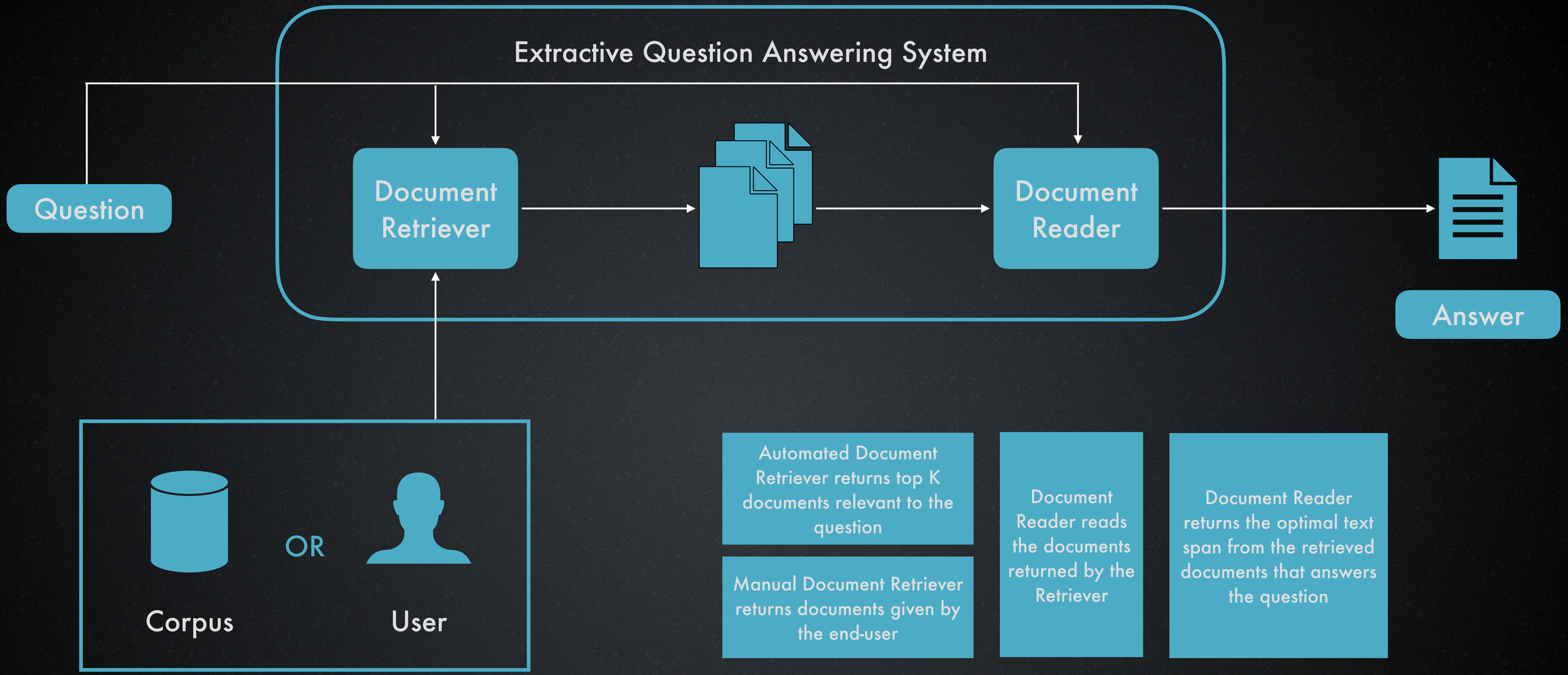




Source

Question-Answering System

(QUERY Module)



Question-Answering System
(Continued)

- ★ Extractive Question Answering
- ★ Use a recently released PolicyQA dataset curated from the OPP-115 corpus
  - ◆ Enables us to model the task as predicting the answer in the given policy segment using existing neural approaches from literature (utilizing Hugging Face transformer API)
  - ◆ A similar dataset SQuAD (Stanford Question Answering Dataset) curated from Wikipedia articles could help provide more insight into the process.
  - ◆ Use pretained BERT on SQUAD 2.0 dataset and fine-tune it on the PolicyQA dataset (with domain-specific word embeddings)

Powered by

	Example 1	Example 2
Policy Text	Information You Give Us: We receive and store any information you enter on our Web site or give us in any other way. Click here to see ...	Promotional Offers: Sometimes we send offers to selected groups of Amazon.com customers on behalf of other businesses. When we do this, we do not give that business your name and address. If you do not want to receive such offers, ...
Question	How do you collect my information?	Is my information shared with others?
Answer	information you enter on our Web site	we do not give that business your name and address

Project Timeline

Full-semester project

- ☆ Literature Survey
- ☆ Preliminary Results

01 Phase 1 Pitch

- ☆ Data Preparation
- ☆ Visualizations
- ☆ Front-end (some progress)

02 Phase 1 Progress Updates

- ☆ Implement Front-end + Back-end (viz)
- ☆ Traditional models
- ☆ Language Neural Models (except BERT)

03 Phase 1 Presentation

- ☆ Transformer models
- ☆ Literature Survey (QA systems)

04 Phase 2 Pitch

- ☆ QA using Transformers
- ☆ Front-end + Back-end

05 Phase 2 Progress Updates

06 Phase 2 Presentation

INFORM module — QUERY module

Final Deliverables

(INFORM + QUERY modules)

- Complete training advanced models for both the classification and QA tasks
- Build and Integrate QA with the web interface
- Build a model playground using Streamlit for technical audience
- Document, attribute, package, host and deploy the complete project stack on Heroku


THANK
YOU