CPSC 490: Senior Thesis Proposal
Tim Follo (tbf5) MC ‘16
February 15, 2016

1. Advisors

Official Advisor: Professor Ruzika Piskac, Computer Science
Additional guidance: Professor Robert Frank, Linguistics
Swapna Reddy, Yale Law School ’16
Debayan Gupta, Computer Science Doctorate Program ’16

2. Background

In early November 2015, part way through the fall semester, I heard about a volunteer project at Yale Law School that was providing refugee families with legal services. My friend Elizabeth Villarreal SY ’16, who was taking a course on Immigration Law at the time, notified me of the project’s pressing need for volunteer help, and brought me to help.

Volunteers were split up on the basis of Spanish language fluency. Because I am not fluent, I joined the data entry team. We were each assigned a block of ID numbers, and we looked through a database that was maintained by case workers at a family detention center, attempting to reconcile any inconsistencies in the database and ultimately ascertain a detainee’s contact information and status, so that the Spanish speaking team could contact the refugees and provide remote legal services.

Unfortunately, I was soon too caught up in academic work for the semester to be a reliable volunteer, and I ceased my involvement. But I could not stop thinking about how this project might benefit from an automated system that could complete data entry efficiently and reliably, freeing up human resources to provide much needed legal resources. I contacted the organizers of the project and met with Swapna Reddy LAW ’16. Swapna informed me that the organizers had, in fact, been planning to integrate more technology into the operation at various levels: making information more accessible with websites, improving internal communication infrastructure, and automating steps that were taxing human researchers.

Swapna will continue to advise my work over the course of the semester, but she took the time to give me a thorough understanding of the legal context, organizational difficulties, and proposed technical solutions, which I have paraphrased below:

**a. Family Detention Centers**

As a result of an influx of American families seeking refugee status during the American refugee crisis in 2014 and, perhaps, anti-immigrant sentiment in the U.S. which generated a negative political reaction to the crisis, the government began to incarcerate refugee families en masse. Multiple detention centers were opened
across the country, designed to detain families who had not yet attained approval for asylum. The largest, a for-profit facility in Dilley, Texas, is called the South Texas Family Residential Center and is designed to hold 2,400 women and children fleeing persecution and intense violence in Central America.

Refugee law globally and policy in the U.S. is a complicated and fraught topic, of which I will provide only a cursory and simplified view. It is important to note that refugee status is granted only to those who can demonstrate a “well-founded fear of persecution based on religion, race, nationality, political opinion or membership in a particular social group.” However, assessing this criterion can sometimes take years, and in the meantime, asylum seekers simply reside in the country they fled to. Usually, in the U.S., people would be allowed to go freely once declaring themselves as a refugee, but in 2014 the government resorted to wholesale incarceration as a containment method.

Detention of refugee families, which contradicts pre-2014 standard practice, is an extremely controversial new immigration enforcement tactic. The average age of children in detention is six years old, and detainees suffer severe psychological trauma under detention conditions: mothers have attempted suicide and children have developed serious disorders.

b. Existing Support Services

A movement of activists, legal scholars, and legislators challenged the new practice of family detention, calling for the immediate release of incarcerated women and children. This was not a fringe movement: 178 U.S. Representatives and 33 U.S. Senators backed the request. Finally, in August 2015, a federal judge ordered an end to the detention of minors and most parents.

A number of groups have emerged to help address persistent problems that refugee families face in detention and once they are released. The CARA Family Detention Pro Bono Project is a coalition made up of the Catholic Legal Immigration Network, the American Immigration Lawyers Association, the Refugee and Immigrant Center for Education and Legal Services, and American Immigration Council. Working with hundreds of volunteers, CARA represents detained families.

c. Service Gap and Yale Law Team

What about the order from a federal judge to stop detention? Unfortunately, in practice many families are still detained due to the complicated situation on the ground, refugees’ limited ability to self-advocate without legal support, and overburdened federal systems and non-profit support networks. CARA works to address this problem by constantly recruiting legal professionals to work pro-bono and engaging outside groups to provide remote legal services.

A group at Yale (which I will refer to as the Yale Law Team), organized by several students at Yale Law School (YLS) and consisting of over one hundred volunteers from Yale College and YLS, have stepped up (in partnership with CARA) to help provide remote legal services to families who have been released from detention
centers but have no legal council or access to legal resources or information about how to navigate the refugee law system.

Considering the narrow categories presented in the definition of a “legitimate” refugee, it can often be difficult for refugees, especially those who do not speak English or are not native English speakers, to learn about the requirements and appropriately present their cases during “full merits hearings” which determine final refugee status. Many families both in detention and once released are still required to have full merits hearings. Missing these, due to poor communication (often caused by unsteady living arrangements, etc.) or misunderstanding, causes families to be ordered deported in absentia, regardless of the danger posed by returning home. But even if a refugee attends the hearing and presents a compelling case involving clear life-threatening conditions in their home country, they may be deported if they failed to explain how the danger they face is really persecution.

Refugees are not promised any sort of representation or legal advocacy, and so they have only the support they can afford to pay for or get for free. Thus, providing continuing legal services to families even after they have left detention centers is critical, and the Yale Law Team fills this void.

\textit{d. Need for Automation}

Thanks to the unwavering dedication and skill of organizers and the work of volunteers, the Yale Law Team has been extremely successful in its efforts. They have helped ensure that every family with full merits hearings at the Dilley detention center receives representation, and they have won every case since May 2015. In addition, they have drafted over one hundred emergency filings to prevent scheduled family deportation.

Accomplishing these feats takes the labor and expertise of 150 qualified legal recruits. However, it also requires significant data handling. The Yale Law Team has collected case information on over 6,000 families, contacting 700 to notify them of important case developments like upcoming hearings. The important information collected is subject to sudden and frequent changes, and information provided by CARA volunteers in a shared database can be inconsistent or incomplete.

Keeping track of all this information has consumed hundreds of hours of volunteers’ time, including the valuable time of qualified legal members of the team. Furthermore, delegating work to a large number of volunteers requires careful orchestration, taking hours of lead organizers’ time. When I volunteered to help in November, I was one of about 100 members, slowly working to complete a reasonably repetitive task. When I dropped out, it took precious energy from multiple coordinators to reassign my caseload to other volunteers.

\textbf{3. Proposal}

I propose to help address the Yale Law Team’s limitations by designing an automated system to replace some human volunteers. This system will be targeted at a specific bottleneck point that is currently causing the most wasted potential in
the Yale Law Team’s ongoing project. The bottleneck emerges when volunteers need to call the Executive Office for Immigration Review (EOIR) Automated Case Information Hotline. This is the only way for volunteers to ascertain information about upcoming hearings, but the long duration of even the shortest phone call, the sheer number of cases to keep track of, and the tendency for case information to change after an initial query make handling more cases increasingly unrealistic as the process consumes volunteer energy.

The solution will consist of several parts. First, an autonomous “robocaller” program must programmatically place calls to the EOIR hotline, supply all relevant information, and interpret the programmed response. Next, a database application must manage the collected information. Finally, I will consider architectural decisions of deployment in the real world both for the Yale Law Team and for other pro-bono groups or future developers.

a. Robocaller

The first challenge of building a Robocaller for this project will be to get the program to interface with the phone network. Google Voice and similar software will provide ways to accomplish this, but writing robust and reliable code to deal with this will be new to me.

The next challenge comes from interpreting the hotline’s output. Because the output is automated, this is essentially a very restricted class of voice recognition problem. Bob Frank (Yale Linguistics), who leads the group Computational Linguistics at Yale, has offered to provide guidance on this critical area. Open source voice recognition packages exist, including one recently released by Baidu. Again, due to the automated response, it should be possible to tune a voice recognition system to get extremely good accuracy.

b. Database Application

Because the number of calls per minute made to the EOIR must be limited to avoid overtaxing the hotline, the database application required for this project does not need to meet any strenuous testing benchmarks, and it does not require a database configuration that is any more complex than a basic DBMS running on a single machine. Much more important characteristics will be ease of use/access and extendibility. These simply require good software design. Debayan Gupta, who is finishing his doctorate in Computer Science at Yale this spring, has offered to advise me on this subject if needed.

c. Real-world considerations

I will not just build a working prototype of a system that the Yale Law Team might pick up; I intend to deliver a fully functional implementation that is ready to help them immediately. This means I will address problems of how to initially seed the database with backlogged cases, I will test or negotiate the upper limit of robocall frequency, I will establish a long-term solution for using phone lines for calls (and I should design it so attached phone lines are interchangeable), and I will come up with a long-term
solution for how the team should host and run the system. Most importantly, I will consider security risks posed by any part of system implementation or deployment.

However, these promises raise some additional considerations. This project must be open-sourced, and yet clearly there are specific, private, parts of the working system described above that would not be a part of a redistributable product. Furthermore, I would like it to be as generalizable as possible, so that other remote legal services groups might be able to adapt it for their own needs.

Thus, I will document the system usage clearly, highlighting room for extensions. Also, I will write a report about all the practical decisions made about how to deploy the system. These will serve as a guideline for future use, as well as information for graders about what I have done. Then I will simply publish the code without seed data.

4. Tasks and Deliverables

- Research on voice recognition solutions
- Configure phone lines
- Develop Robocaller
- Develop Database Application
- Populate Database with existing Data
- Implement Live Server
- Report on Design considerations
- Thorough documentation
- Publish Open Source Code
  - Database Application (Empty)
  - Robocaller

5. Timeline

Week 0 - 2/7/16
- Project Proposal and Background Report
- Initial structures: set up repositories, blank DBMS, skeleton database application, skeleton automated caller application

Week 1 - 2/14/16
- Research voice recognition solutions
- Database ER diagrams
- Identify database application functions
- Identify government contacts, try to get database snapshot

Week 2 - 2/21/16
- Write all results into report
- Finalize voice recognition solution
- Configure phone lines within robocaller application
Week 3 - 2/28/16
- Implement database tables
- Robocaller: fully automate timed calls

Week 4 - 3/6/16
- Begin database application
- Populate private database with snapshot information
- Begin processing automated responses

Week 5 - 3/13/16 – (Spring Break)
- Catch up

Week 6 - 3/20/16 – (Spring Break)
- Catch up

Week 7 - 3/27/16
- Research private live server options
- Write up design considerations report

Week 8 - 4/3/16
- Finish Database application

Week 9 - 4/10/16
- Robocall response handling: Voice Recognition Testing

Week 10 - 4/17/16
- Robocall response handling: Completed application

Week 11 - 4/24/16
- Documentation: installation and usage instructions
- Set up and finalize live private server: full working system

Week 12 and beyond - 5/1/16 – (Reading Period and Finals)
- Catch up and polish