Babler - Data Collection from the Web to Support Speech Recognition and Keyword Search

Gideon Mendels, Erica Cooper, Julia Hirschberg

Columbia University

Introduction - Babler

- A high throughput, Multi-language Web Data collection System.
- Collecting large amounts of **clean** web data in Low Resource Languages.
- Data collected is used to build Language Models for ASR and Keyword Search.
- Evaluation is done over the Babel Language Packs and KWS Queries.

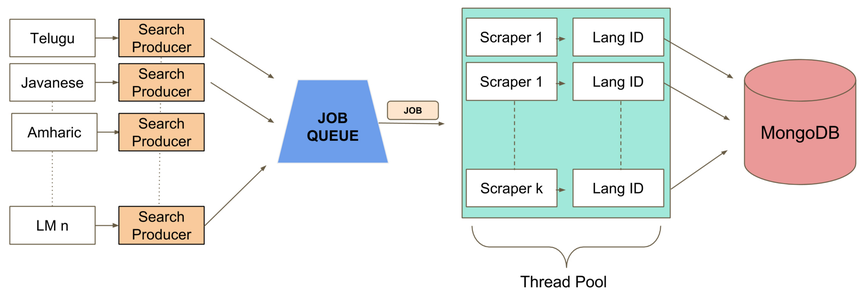
Web Data Sources

- **Blogs:** Collection is done from Wordpress and Blogspot Blogs using RSS feeds and boilerplate removal techniques.
- **Forums:** phpBB forums using manually designed CSS Queries.
- **Twitter:** Seeding API Queries and user level crawling.
- **News:** Manually designed scrapers for major news outlets in our target languages.
- **Wikipedia:** Automatically downloading and parsing Wikipedia dumps.

System

- **Seeding Language Models:** Computes a bigram LM to be used for seeding data collection.
- **Search Producer:** Polls a query from the LM and performs a search using the following: Bing search API, DuckDuckGo API, Google Search, Twitter API and Topsy API.
- **Job Queue:** Collects scraping jobs from the previous steps and distribute them concurrently over the various scrapers.
- **Scraper:** Polls and extracts the data from the source. Each data source requires a different logic for extraction.
- **Language Identification** Verifies that the data collected is in one of the target languages. We use a majority vote decision using the following classifiers:
  - LingPipe - Character ngram classifier trained on the Leipzig Corpora data.
  - TextCat - Custom implementation using pre-computed counts from the Crubadan Project.
  - Google CLD - Naive Bayes Classifier that supports 83 Languages.

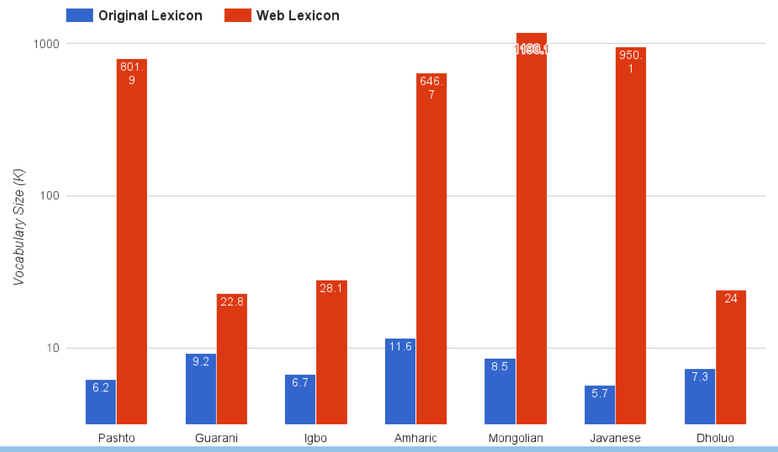
System Diagram



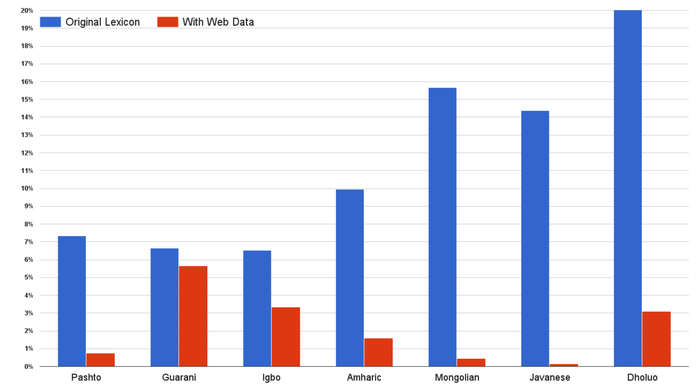
The Babel Program - Experiments

- Babel targets rapid development of speech processing technology in low resource languages, focusing on keyword search in large speech corpora from ASR transcripts.
- Our goal in collecting web data is to supplement language models for ASR and KWS by increasing the lexicon available from the ASR training corpus in order to reduce the number of OOV words.
- We calculate OOV reduction by comparing the web data augmented lexicon with each of the Babel LLP lexicons for the six Babel OP3 languages.

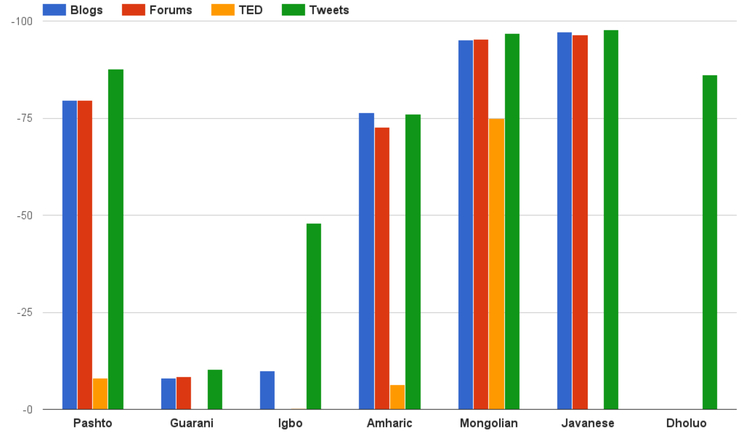
Original Vocabulary vs. Collected Vocabulary



OOV Hit Rate (Lower is better)



OOV Hit Rate Relative Change by Data Source



Conclusions and Future Work

- The data collected by the system has reduced Out-Of-Vocabulary rates for KWS in LRLs, resulting in significantly higher KWS scores.
- By including language identification and text normalization as part of our pipeline, we can be more confident that the data we collect is likely to be in the target language.
- **Future Work:** Exploring additional sources for conversational web data and open sourcing **Babler.**