Appendix for Demographic Dialectal Variation in Social Media: A Case Study of African-American English

Su Lin Blodgett Lisa Green Brendan O'Connor
University of Massachusetts Amherst

August 23, 2016

1 Appendix

1.1 Census demographics (§2.1)

These four "races" (non-Hispanic whites, Hispanics, non-Hispanic African-Americans, and Asians) are commonly used in sociological studies of the U.S. The Census tracks other categories as well, such as Native Americans. The exact options the Census uses are somewhat complicated (e.g. Hispanic is not a "race" but a separate variable); in a small minority of cases, these values do not sum to one, so we re-normalize them for analysis and discard the small fraction of cases where their sum is less than 0.5. For simplicity, we sometimes refer to these four variables as *races*; this is a simplification since the Census considers race and ethnicity to be separate variables, and the relationship between the actual concepts of race and ethnicity is fraught on many levels.

1.2 Unicode ranges for emoji removal (§4.1)

We observed that emoji symbols often caused *langid.py* to give strange results, so we preprocessed the data (improving *langid.py*'s predictions) to remove emoji and other symbolic-type characters by removing all characters falling into particular Unicode ranges. We were not able to find effective pre-existing solutions, so developed range lists by consulting documentation on Unicode and emoji standards, and observing message samples and how our proposed rules changed them. See *emoji.py* for our implementation. Note that Unicode consists of 17 planes of 65,536 codepoints each; each plane contains a number of variable-sized blocks. The first three planes are most often used today (Basic Multilingual, Supplemental Multilingual, Supplemental Ideographic). We remove all characters from the following ranges.

- 10000–1FFFF: The entire Supplemental Multilingual Plane, which contains emoji and other symbols, plus some rarely used scripts such as Egyptian hieroglyphics.
- 30000–10FFFF: The fourth and higher planes.
- 02500-02BFF: A collection of symbol blocks within the Basic Multilingual Plane, including: Box Drawing, Box Elements, Miscellaneous Symbols, Dingbats, Miscellaneous Mathematical Symbols-A, Supplemental Arrows-A, Braille Patterns, Supplemental Arrows-B, Miscellaneous Mathematical Symbols-B, Supplemental Mathematical Operators, Miscellaneous Symbols and Arrows.

- 0E000–0EFFF: A "private use" area in the BMP, for which we observed at least one nonstandard character in Twitter data (U+E056).
- 0200B–0200D: Zero-width spaces, joiner, and nonjoiner. The ZWJ is used to create compositional emoji.[1]
- 0FE0E, 0FE0F: Variation sequences, which are invisible post-modifiers allowed for certain emoji.[2]

1.3 Posterior inference via CVB0 for ensemble classifier (§4.2.1)

The posterior inference task is to calculate the posterior expectation of

$$P(\theta \mid w, \phi, \alpha) \propto P(\theta \mid \alpha) P(w \mid \theta, \phi)$$

where ϕ are the trained topic-word language models and $\theta \sim Dir(\alpha)$ is a prior over topic proportions, with a fixed symmetric prior $\alpha_k = 1/16$.

The ϕ topic-word distributions are calculated via training-time posterior inference by averaging Gibbs samples $\bar{N}_{wk} = (1/S) \sum_s$ (where s indexes the last 50 samples of the Gibbs sampler), as well as adding a pseudocount of 1 and normalizing:

$$\phi_{k,w} \propto (\bar{N}_{k,w} + 1)$$

(The detailed balance theory of MCMC implies no pseudocount should be added, but we found it seemed to help since it prevents rare words from having overly low posterior expected counts.)

The $\hat{\theta}$ prediction is inferred as the posterior mean given the words in the message by using the "CVB0" version of variational Bayes (**?**), which is closely related to both Gibbs sampling and EM. It iteratively updates the soft posterior for each token position $t = 1..T$,

$$q_t(k) \propto (N_{-t,k} + \alpha_k)\, \phi_{k,w_t}$$

where $N_{-t,k} = \sum_{t' \neq t} q_{t'}(k)$ is the soft topic count from other tokens in the message. The final posterior mean of θ is estimated as $\hat{\theta}_k = (1/T) \sum_t q_t(k)$. We find, similar to **?**, that CVB0 has the advantage of simplicity and rapid convergence; $\hat{\theta}$ converges to within absolute 0.001 of a fixed point within five iterations on test cases.

1.4 Syntactic dependency annotations (§5)

The SyntaxNet model outputs grammatical relations based on Stanford Dependencies version 3.3.0;[3] thus we sought to annotate messages with this formalism, as described in a 2013 revision to **?**.[4] For each message, we parsed it and displayed the output in the Brat annotation software[5] alongside an unannotated copy of the message, which we added dependency edges to. This allowed us to see the proposed analysis to improve annotation speed and conformance with the grammatical standard. For difficult cases, we parsed shortened, Standard English toy sentences

---

[1] `http://www.unicode.org/emoji/charts/emoji-zwj-sequences.html`

[2] `http://unicode.org/reports/tr51/`

[3] Personal communication with the authors.

[4] We only had access to the 2015 version, currently available online. We also considered follow-up works **?** and `http://universaldependencies.org/`, deferring to **?** when in conflict.

[5] `http://brat.nlplab.org/`

to confirm what relations were intended to be used to capture specific syntactic constructs. Sometimes this clearly contradicted the annotation standards (probably due to mismatch between the annotations it was trained on versus the version of the dependencies manual we viewed); we typically deferred to the parser's interpretation in such cases.

In order to save annotation effort for this evaluation, we took a partial annotation approach: for each message, we identified the root word of the first major sentence[6] in the message—typically the main verb—and annotated its immediate dependent edges. Thus for every tweet, the gold standard included one or more labeled edges, all rooted in a single token. As opposed to completely annotating all words in a message, this allowed us to cover a broader set of messages, increasing statistical power from the perspective of sampling from a message population. It also alleviated the need to make fewer difficult annotation decisions - linguistic phenomena such as mistokenized fragments of emoticons, symbolic discourse markers, and (possibly multiword) hashtags.

We use the *twokenize* Twitter-specific tokenizer for the messages, which separates emoticons, symbols and URLs from the text (**??**)[7] and use the space-separated tokenizations as input to SyntaxNet, allowing it to tokenize further. This substantially improves accuracy by correctly splitting contractions like "do n't" and "wan na" (following Penn and English Web Treebank conventions). However, as expected, it fails to split apostrophe-less forms like "dont" and more complicated multiword tokens like "ima" (*I am going to*, which **?** sought to give a joint Pronoun-Verb grammatical category), typically leading to misanalysis as nouns. It also erroneously splits apart emoticons and other multi-character symbolic expressions; fortunately, these are never the the head of an utterance, so they do not need to be annotated under our partial annotation design.

We find that the Stanford Dependencies system worked well as a descriptive formalism for tweets' syntax, including AAE constructions; for example, several cases of "gone-V," "done-V," and habitual *be* were analyzed as auxiliary verbs (e.g. aux(let,done) in "I done let alot of time go by"), and the SD treatment of copulas trivially extends to null copulas.

Multiword tokens like an untokenized "dont" (*do not*) or "af" (*as fuck*, syntactically a PP) pose a challenge as well. We adopt the following convention: for all incoming edges that would have gone to any of their constituent words (had the token been translated into a multitoken sequence), we annotate that edge as going to the token. If there are mulitple conflicting edges—which happens if the subgraph of the constituent words has multiple roots—the earliest token gets precedence. For example, "I 'm tired" has the analysis nsubj(I,tired), cop('m,tired); thus the multiword token "Im" in "Im tired" would be internally multirooted. "I" has priority over "'m", yielding the analysis nsubj(tired,Im).

Punctuation edges (*punct*) were not annotated. We found discourse edges (*discourse*) to be difficult annotation decisions, since in many cases the dependent was debatably in a different utterance. We tended to defer to the parser's predictions when in doubt. The partial labeling approach does not penalize the parser if the annotator gives too few edges, but these issues would have to be tackled to create a full treebank in future work.

---

[6] We take **?**'s view that a tweet consists of a sequence of one or more disconnected utterances. We sought to exclude minor utterances like "No" in "No. I do not see it" from annotation; in this case, we would prefer to annotate "see.". A short utterance of all interjections was considered "minor"; a noun phrase or verb-headed sentence was considered "major."

[7] Using Myle Ott's implementation: `https://github.com/myleott/ark-twokenize-py`

1.5 Annotation materials

We supply our annotations with the online materials[8] as well as working notes about difficult cases. Annotations are formatted in *Brat*'s plaintext format.

---

[8] http://slanglab.cs.umass.edu/TwitterAAE/