CS6740/IS 6300, Lecture 21:

1. **What was the "state of the art" on our A3 challenge data?** The results of the "Towards Linguistically Generalizable NLP Systems" shared task.

2. **Challenge datasets and/vs. generalization to unseen/"new" data**:

Challenge datasets have been used to expose/analyze model weaknesses.
But even just "new" can pose at least a little challenge.

3. What changes when domain changes? Elsahar and Gallé's (2019) description of previous distinctions, letting "s" stand for "source" and "t" for "target"

1. *Covariate shift*, where the input distribution differs:
   $P_s(x) \neq P_t(x)$
   but the conditional distribution doesn't:
   $P_s(y|x) = P_t(y|x)$

2. *Concept shift*, where the input distribution is the same but the conditional distribution differs $P_s(x) = P_t(x)$ and $P_s(y|x) \neq P_t(y|x)$

3. Label shift [$P_s(y) \neq P_t(y)$ and $P_s(x|y) = P_t(x|y)$], mixtures of these, etc., which we'll ignore.

4. **Are there ways to measure these differences with only unlabeled target data? (No labeled target data?)**
   Do these measures in fact correlate with performance (drop) on the target domain?

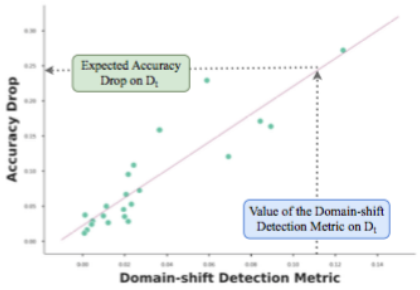

Figure 1: In this paper we introduce several domain-shift detection metrics (x-axis) and employ them to estimate the performance drop on a new target domain D_t by regressing on those metrics and their associated real performance drop (green dots).

5. **Measure: How different are $P_s(x)$ and $P_t(x)$?** (Kifer et al 2004, Ben-David et al 2010)
    a) Build a classifier G which predicts whether x is in dataset D_s or dataset D_t , and use G's error rate.
    b) What if there are domain variances that have nothing to do with the classification problem of interest?

6. **Measure: confidence change in labels**. (May need calibration step.) Train on D_s, use trained classifier to label D_t, look at average confidence of labels for items in D_s vs D_t

7. **Measure: reverse classification accuracy.** (Fan and Davison 2006, Zhong et al., 2010)[1]
    a) Assume we can estimate $\frac{P_t(x)}{P_s(x)}$ to deal with input-distribution difference. How can we get at $P_t(x|y)$ without labeled D_t data?
    b) Break the source data D_s into S test and $\overline{S}$ train.[2]
    c) Train on $\overline{S}$, use trained classifier f to label D_t , so you're using $P_f(y|x)$
    d) Build a new classifier g trained on D_t, check its accuracy back on S , so you're looking at $P_g(y|x)$.
        - In Elsahar and Gallé, they also try using heldout source data for D_t, which is like our experiment B in A3!

[1] You'd like to estimate $P_t(y|x)$. The Zhong et al. paper takes the stance that $P_g(y|x)$ can be treated as an interpolation between $P_f(y|x)$ and $P_t(x|y)$ plus some error, but this mostly seems to be true for when you are given some labeled data in the target domain. Otherwise, I don't see right away a theoretical guarantee that this should work.

[2] Notation mismatch is to deal with difference in notation between Elsahar and Gallé (2019) and Zhong et al. (2010).