Determining support or opposition from Congressional floor-debate transcripts

Cheuk Hang Lee

December 14, 2015

1 Introduction

This project investigates a system proposed by Thomas, Pang, and Lee [1] that determines if a speech in U.S. Congressional floor debate supports the proposed legislation. In addition to assessing each speech individually, the system also look for references to other speakers in the speeches in order to improve its accuracy. This project will reimplement the system, verify some claims made in the paper, and modify the system to account for direct references to the legislation. The code for this project is available on

https://github.mit.edu/chsamlee/6806-final

2 Reimplementing the System

2.1 Procedure

Let $s_1, \ldots, s_n$ be speeches in a debate. The binary classifiction system aims to find the assignment $c(s_1), \ldots, c(s_n) \in \{Y, N\}$ (Y indicates support, N indicates opposition) that minimizes

$$\sum_s ind(s, \bar{c}(s)) + \alpha \sum_{s,s':c(s)\neq c(s')} \sum_{l \text{ between } s,s'} str(l),$$

where $ind(s, \bar{c}(s))$ is a nonnegative function that indicates preference of an individual-speech classifier to assign speech s to the opposite label of $c(s)$, and $str(l)$ is the nonnegative strength of link l between two speeches that indicates preference that the two speeches is assigned to the same class. α is a hyperparameter. The system is constructed by the steps below. Stage 3 data is used unless otherwise specified.

1. The individual-speech SVM is trained using presence-of-word vector of unigrams, i.e. a component is 1 if the corresponding word is present in the speech and 0 otherwise. The vector is normalized before training.

2. The individual score is given by
$$ind(s, Y) = \frac{d(s)}{4\sigma_s} + 0.5,$$
where $d(s)$ is the distance from separating hyperplane and σ_s is the standard deviation of $d(s)$ over all speeches in the debate. It is limited to the range of $[0, 1]$, and $ind(s, N) = 1 - ind(s, Y)$.

3. The input to agreement SVM is a normalized presence-of-word vector of unigrams, but only part of the speech is used for vector construction (versus the whole speech in individual SVM). We use the 30 tokens before and 20 tokens after a reference to another speaker. Stage 1 data is used for training.

[1] http://www.cs.cornell.edu/home/llee/papers/tpl-convote.dec06.pdf

4. The strength of agreement is given by

$$str(r) = \frac{d(r) - \theta_r}{4\sigma_r},$$

where $d(r)$ is the distance from separating hyperplane, σ_r is the standard deviation of $d(r)$ over all references in the debate, and θ_r is a tunable debate-dependent parameter. In the paper and in the project, θ_r will either be 0 or μ, the mean of $d(r)$ over all references in the debate.

5. Keeping α fixed, find the optimal assignment by modelling the debates as minimum-cut problems (see next subsection). Find the best value of α using development set. Stage 2 data is used in this step.

6. Run the complete classifier on test set using stage 2 data for agreement links and stage 3 data for individual scores.

2.2 Assignment as a Minimum-cut Problem

We construct a single-source, single-drain flow network for each debate as follows:

1. Construct a node for each speaker in the debate.

2. For each speech s, construct an edge from source to the speaker with capacity $10000 \cdot ind(s, Y)$, and an edge from the speaker to the drain with capacity $10000 \cdot ind(s, N)$.

3. For each agreement link with positive strength, add two edges between the two speakers (one in each direction) with capacity $10000\alpha \cdot str(l)$.

Recall that a cut (S, T) is a partition of nodes such that the source is in S and the drain is in T, and the capacity of the cut is the sum of capacities of all edges from S to T. Suppose that we assign the label of Y to all nodes in S. Then if $s \in S$, the edge from s to drain crosses the cut, and the edge from source to s crosses the cut if $s \in T$. Thus the sum of capacities so far is

$$10000 \sum_s ind(s, \bar{c}(s)).$$

Furthermore, if s and s' have different labels, then for each reference there is exactly one edge that crosses the cut by our construction. Thus the capcity of the cut is

$$\sum_s ind(s, \bar{c}(s)) + \alpha \sum_{s,s':c(s)\neq c(s')} \sum_{l \text{ between } s,s'} str(l),$$

which is the function we are trying to minimize. Thus we can find the optimal assignment from a minimum cut of the network. Note that we use one node per speaker instead of one node per speech with same-speaker links to enforce the same-speaker constraint.

2.3 Results

The results are divided into 3 columns: the first column is the result reported by Thomas, Pang and Lee. The second column is the result of the reproduced system, using the same debates as the paper for each dataset. The third column's results are obtained from randomizing the debates to pick for each set.

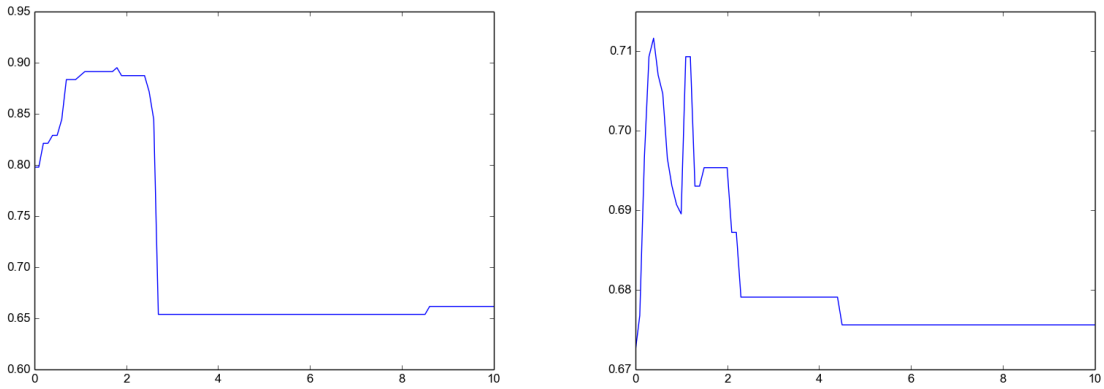
(Development set)	paper	reimplementation	randomized input
Majority baseline	54.09	54.09	53.35
SVM	70.04	70.04	68.61
SVM + same-speaker	79.77	79.77	79.38
SVM + same-speaker + agreement link ($\theta = 0$)	89.11	89.49	82.73
SVM + same-speaker + agreement link ($\theta = \mu$)	87.94	87.94	81.95

(Test set)	paper	reimplementation	randomized input
Majority baseline	58.37	58.37	54.67
SVM	66.05	66.05	65.61
SVM + same-speaker	67.21	67.21	66.00
SVM + same-speaker + agreement link ($\theta = 0$)	70.81	69.53	69.44
SVM + same-speaker + agreement link ($\theta = \mu$)	70.81	70.35	68.56

It is worth noting that randomizing the inputs have a huge impact on accuracy: the accuracy using SVM, same-speaker constraint and agreement links can range from 60% to 85% depending on separation. We are not sure of the reason for the huge variance, but one explanation is that more data is needed as the development set sometimes only contains 3 debates. The output for third column is obtained by assigning the following debate numbers to each set:

Training: [13, 16, 38, 48, 52, 79, 84, 88, 90, 102, 108, 132, 168, 182, 199, 204, 239, 321, 360, 367, 421, 426, 467, 470, 472, 493, 495, 507, 519, 534, 548, 553, 559, 585, 599, 616, 621, 631, 635]
Development: [31, 539]
Test: [6, 144, 282, 296, 371, 372, 414, 475, 506, 533, 547, 645]

The gap between development set results and those of test set is more consistent across trials. We notice that inputs that yield larger gaps tend to have larger values of optimal α. This suggests that α may be context-sensitive, and large value of α may indicate overfitting. The paper also mentions hard agreement constraints where speeches in agreement are forced to have the same label, but does not provide numerical data on its effect on accuracy (but it is stated that it degrades performance). Hard agreement constraint can be simulated by taking the limit $\alpha \to \infty$. The following graphs show the performance of system as a function of α, using unrandomized input:



Development set and test set performance as function of α

It is clear that performance drops substantially as α increases, but stabilizes as $\alpha \to \infty$. The effect is also less noticible on test set.

3 References to Legislation Name

Exploiting references to other speakers in a speech improves classification accuracy. Thus it is natural to ask whether references to the legislation itself contain clues to the speech's stance. A quick analysis on individual scores seems to point to "yes": SVM gives higher accuracy on speeches with mentions (86.21% vs. 70.04% on development set, 68.75% vs. 66.05% on test set) and with slightly higher confidence (measured by average difference of individual scores and 5000, the neutral score). To account for mentions, the total cost function

to minimize is now

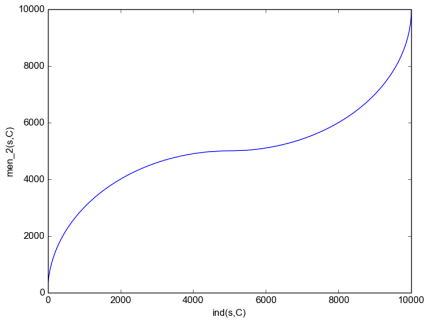
$$\sum_s ind(s, \bar{c}(s)) + \beta \left[\sum_{s:s\in M} men(s, \bar{c}(s)) \right] + \alpha \left[\sum_{s,s':c(s)\neq c(s')} \sum_{l \text{ between } s,s'} str(l) \right].$$

where M is the set of speeches that mentions the legislation. This extra cost can be accounted for in the flow network by adding an edge from source to s with capacity $men(s, Y)$ and an edge from s to drain with capacity $men(s, N)$. For simplicity, we will only use references by bill number (e.g. H.R. 27, S. 256). We will experiment with different men functions:

- $men_1(s, C) = ind(s, C)$
  This functions weigh the individual scores of speeches with mentions more heavily. A downside to using this simple function is that we weigh a speech with mention and individual score of 5001 more heavily than a speech without mention but with individual score of 10000.
- $men(s, C)$ is a nonlinear function of $ind(s, C)$
  A nonlinear function fixes the problem mentioned above because the quantity $\frac{men(s,C)}{ind(s,C)}$ is allowed to vary. We use the function

  $$men_2(s, C) = \begin{cases} 10000 - \sqrt{5000^2 - (ind(s, C) - 5000)^2} & \text{if } ind(s, C) \geq 5000 \\ \sqrt{5000^2 - (ind(s, C) - 5000)^2} & \text{if } ind(s, C) < 5000 \end{cases}.$$

  Observe that the edge weight is very small when $ind(s, C)$ is close to 5000.



- $men_3(s, C)$: SVM-based scoring
  We first find the direct references to the legislation with the form of TOKEN NUMBER, whwere token is one of h.r., s., or res. (the full identifier should be h. res.). For each reference we find, we train the SVM using the speech's stance as label and a normalized presence-of-word vector of unigrams as input, constructed using 10 words before and after the reference. Predictions are handled in the same way, except that we only use the first direct reference in the speech. This is because the subsequent mentions often are technical explanations on the congressman's stance, and are far noisier than the first mention.
  Another choice we have made is to not normalize the raw scores unlike in individual and agreement SVM. This is because there are few speeches that directly mentions the legislation, and normalization severely distorts the data. For example, after normalization the scores (4999,5001) is undistinguishable from (0,10000).

The results can be found in the table below. For accuracy and ease of reproducing the result, we used the same separation of sets as the paper.

		development set	test set
$\theta = 0$	baseline: SVM + same-speaker+agreement link	89.49	69.53
	baseline + $men_1(s, C)$	89.49	70.47
	baseline + $men_2(s, C)$	89.49	71.98
	baseline + $men_3(s, C)$	89.88	70.93
$\theta = \mu$	baseline: SVM + same-speaker+agreement link	87.94	70.35
	baseline + $men_1(s, C)$	87.94	70.23
	baseline + $men_2(s, C)$	87.94	70.35
	baseline + $men_3(s, C)$	88.33	76.40

We are not sure if the huge jump of accuracy using SVM-based mention score and high-precision agreement links is coincidental or not. Randomizing the input usually gives 1 to 2 percent increase of accuracy compared to baseline for both choices of θ. The result may possibly be further improved by using both men_2 and men_3.

Similar to α, forcing speeches to receive the same label as the SVM prediction degrades the performance of the system. However, the degradation is not as drastic: development set accuracy decreases by 6% and 7%, and test set accuracy decreases by 2% and 0.1% for the cases $\theta = 0$ and $\theta = \mu$ respectively. This result is surprising as the mentions classifier has lower accuracy and precision than the agreement classifier (81.58% vs 86.30% on development set).