Predicting Viewer Reactions to Discourse in Political Debates

Peter Enns
Computer Science
University of Maryland
peter@cs.umd.edu

Abstract

It is increasingly common for researchers in the humanities and social sciences make use of computational analysis of large datasets to answer research questions. One field where computational analysis is getting traction is political science. In this paper, we leverage a particularly rich dataset from the domain of political science that contains the transcript of one of the 2012 presidential debates and fine-grained reactions from viewers over the course of the debate. This dataset allows us to investigate the effectiveness of tools from natural language processing for answering interesting research questions about the debate. In particular, we attempt to predict viewer reactions to utterances from both candidates. We weigh the benefits and drawbacks of several competing approaches and explore some potential improvements.

1 Introduction

In recent years, social scientists have embraced automated computational analysis of datasets to answer interesting research questions in their fields (Lazer et al., 2009; O'Connor et al., 2011). This is especially true in political science, a domain where until recently, progress has been hampered by the fact that it is impossible to manually analyze the available corpora with traditional methods despite it being well known that understanding what politicians *write* and *say* is central to making advances (Grimmer and Stewart, 2013). The importance of computational analysis will become increasingly apparent as larger corpora continue to become available. This will inevitably occur as more politicians engage in social media, political science researchers continue publish their high quality datasets on the internet, and companies that collect data for political science research arise.

The React Labs: Educate project is a particularly interesting source of data for political scientists. It is a mobile application that allowed students to register their real-time reactions to the 2012 presidential and vice presidential debates (Boydstun et al., 2014b). Political science instructors across the country were invited to offer their students extra credit for using the app while watching the debates (Boydstun et al., 2014a). In return, Boydstun et al. provided those instructors with teaching materials related to the debate and reactions that are now available online (http://reactlabseducate.wordpress.com/). This made it possible to collect moment by moment reactions from a group of 3,340 students with demographics closer to the national average than what could have been collected from a single-campus study. This data is extremely valuable for political science research because it provides unprecedented temporal resolution for viewer reactions over the course of a debate.

In this paper, we explore several techniques for analyzing the React Labs: Educate data in conjunc-

[0]Part of this paper originated from a group project with University of Maryland computer science students Isaac Julien and Alex Memory. In particular, section 4.1 describes work done primarily by Alex, and section 4.2 describes work done primarily by Isaac. The datasets were generously provided by Amber Boydstun, Philip Resnik, and React Labs.

tion with the debate transcript itself. We begin by providing a more detailed description of the data in section 2. Next, we discuss a framework for predicting viewer reactions, and discuss the benefits and drawbacks of several approaches of selecting features for supervised classification in section 3. Importantly, we compare the efficacy of completely automated techniques to a technique that requires manual contributions by political science experts. In section 4, we discuss the results of running the experiments outlined in section 3. We continue by exploring how we might improve one of the automatic techniques with more sophisticated statistical models in section 5. Finally, we discuss our results and further research directions in section 6.

2 Data

2.1 Debate Corpus

The corpus used here is an annotated transcript from the October 3rd 2012 presidential debate between President Barack Obama and Mitt Romney. The corpus is split into "quasi-sentences," each one timestamped and painstakingly annotated with the current speaker, primary topic, and primary frame by a group of expert political scientists (Boydstun et al., 2014b). The **topics** are subjects or themes derived from from the Political Agendas Topics Codebook. [1] A **frame** is a particular conceptualization of a topic (e.g., moral or constitutional). The corpus is comprised of approximately 12,000 quasi-sentences. We preprocessed the corpus by chunking it into turns in each debate, and by performing other typical preprocessing steps (e.g., word tokenization and stopword removal). For our purposes, a **turn** is a continuous portion of the transcript spoken by a single person (usually multiple quasi-sentences long). Finally, since we perform 10-fold cross validation in experiments described in section 3, we created 10 unique splits of the corpus with 90% of the turns dedicated to training and 10% of the turns dedicated to testing.

2.2 Viewer Reactions

In addition to the debate corpus, we also used reactions from viewers of the debate collected by React Labs: Educate. This dataset contained 193,287 reactions (e.g., agreeing with one of the candidates) from debate viewers using the React Labs: Educate mobile app. Each datapoint corresponds to a single reaction (Romney:Disagree, Obama:Spin, etc...) with a timestamp and metadata about the user who submitted the reaction.

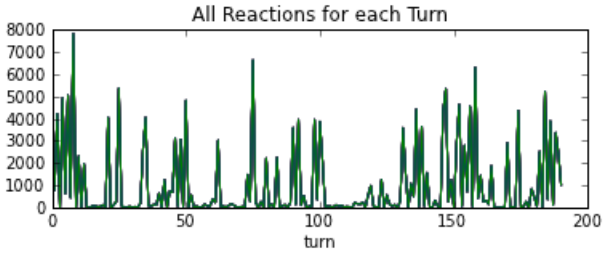

Figure 1: Reactions by all users for each turn.

The reactions started a half hour before the debate and continued for a half hour after the debate – we discard reactions from these time periods – and overall the reactions are tightly clustered in time; see Fig. 1

Since the annotated debate corpus and reactions dataset are both timestamped, we were able to associate reactions to each *turn* taken by a candidate during the debate. However, it is common for users to react to someone who is not speaking. In Fig. 2 we see that it is especially common for users to react to the moderator while one of the candidates is speaking, perhaps because the moderator speaks for shorter periods of time than the candidates. Also, an even larger number of reactions to the candidates are assigned to one another or the moderator, perhaps because the users are still reacting to a candidate's last turn.

For this reason, we limit the reactions we consider to those where the reaction is associated with the person *currently speaking*. This reduces the number of reactions records by approximately 21%. [2]

Overall, we see in Fig. 3 that reactions to Obama are overwhelmingly agreement, while a high percentage of reactions to Romney are dodge, spin or disagree reactions. This is not surprising, given the stated preferences of the users at the beginning of the debate, cf. Fig. 4; the users prefer Obama over Romney by over two to one. We will see that this

[1] See http://www.policyagendas.org/

[2] Boydstun et al. use any 5 second rolling window in which a candidate discussed a topic for their analysis to address this issue.

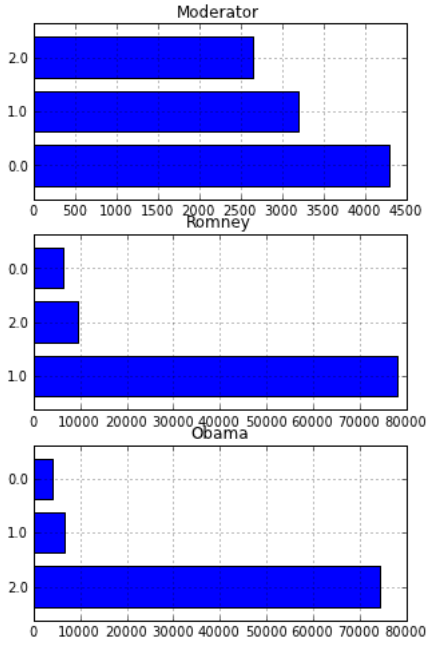

Figure 2: Reactions to the moderator (top), Romney (middle) or Obama (bottom) while speaker 0 (Moderator), 1 (Romney) or 2 (Obama) was talking.

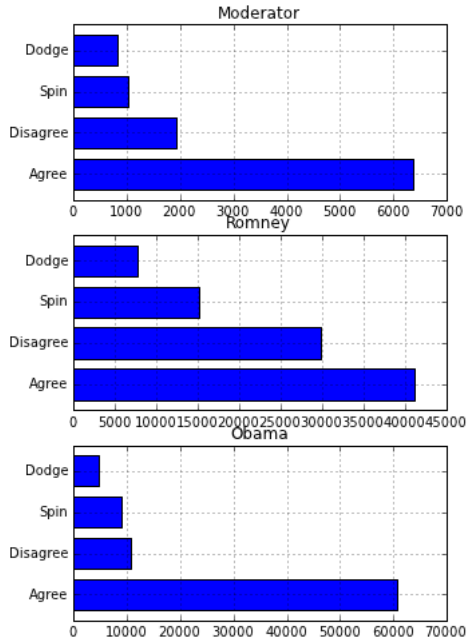

Figure 3: Frequencies of reactions of each type over the course of the debate for the moderator (top), Romney (middle) and Obama (bottom).

imbalance may create issues of bias in training examples once we begin to predict reactions of Obama or Romney supporters for what is being said in each turn.

3 Predicting Debate Viewer Reactions

With this data, we can attempt to answer a very interesting question: how well can viewers' reactions be predicted from portions of the debate transcript? With viewer reactions matched with turns in the debate, it is possible to frame this as a supervised classification problem where we predict the most likely reaction given a vector of features generated from properties of what was said during a turn. With this framework in mind, we will investigate the advantages and disadvantages to several approaches to generating features.

3.1 Features

3.1.1 Ngrams

The simplest possible model one could make to predict user reactions would use n-gram features, so we decided to use this as a baseline. To extract n-gram features from the transcripts of the turns, we

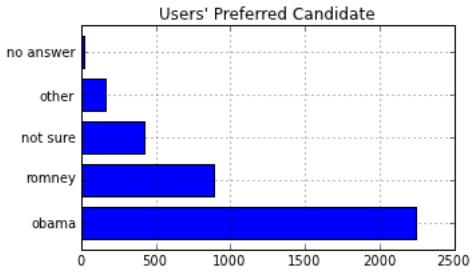

Figure 4: Pre-debate candidate preferences reported by the users reacting to the debate.

begin by splitting the text into tokens using the English tokenizer from NLTK (Bird, 2006). We then remove punctuation, numbers and stop-words; and then convert all n-grams to lower-case. Finally, we produce a single feature for each unique n-gram in each turn indicating its presence (not the count of tokens for that n-gram).

3.1.2 Manual Annotations

Each “quasi-sentence" in the debate corpus was annotated by a group of expert political scientists (Boydstun et al., 2014b). The annotations include the candidate speaking, the primary topic (e.g., “defense"), and the primary frame (e.g., “moral"). We thought the primary topics could potentially distill the information carried by the n-grams into a much smaller number of features, making for a smaller and more elegant model. For this reason, we decided to follow our baseline model with a model that used the coded topics as features. For a given turn, the value of that feature is the fraction of quasi-sentences within the turn labeled with that topic.

3.1.3 Automatic Annotations

Manually annotating debates is slow and laborious process, so it may not always be practical but, fortunately, there are alternatives. Latent Dirichlet Allocation (LDA) is a Bayesian generative model where each document in a corpus is modeled as a finite mixture over a set of latent categories (Blei et al., 2003). It is commonly used to model the topics (latent categories) discussed in a corpus of documents, viewed as “bags of words." LDA topics are simply multinomial distributions over words. Running a sampler for LDA on a corpus of documents yields a multinomial distribution over topics for each document that is easily converted into a feature vector for that document. Since it was unlikely that the topics generated with LDA would be of the same caliber as the coded topics, we thought the next plausible step would be to try more complicated (and possibly more appropriate) generative topic models.

We ran Mallet's (McCallum, 2002) implementation of LDA on the entire corpus with hyperparameter optimization on, treating each turn as a document. Figure 5 shows the top terms for each topic distribution generated by one run of LDA. We decided to model 19 topics in order to agree with the number of topics in Boydstun's annotations. The features for each document in our predictive models were the proportions of topics LDA allocated to those documents.

3.2 Tasks

There were many possible reactions that were not mutually exclusive (e.g., “obama:agree from a Democrat and “obama:spin" from a Republican in response to the same turn), so it was impractical and uninformative to simply predict one reaction from the text of a turn in the debate. For this reason, we carved out three more manageable tasks in order to compare the performance of the different methods of generating features.

3.2.1 Task 1: Predicting Overall Reactions

The purpose of **Task 1** is to evaluate how well-suited the features in each model are to predict the overall rate of user reactions for Obama voters and Romney voters separately. To do this, we calculated the number of reactions for each turn in the debate and divided by the length of the turn in seconds. Each turn in the debate was then labeled with either a 1 if the reactions per second for that turn was greater than the median for all turns or 0 otherwise. This was done once for reactions from Democrats and a second time for reactions from Republicans.

3.2.2 Task 2: Predicting Agree Reactions

The purpose of **Task 2** is to evaluate how well-suited the features in each model is to predict if Obama voters or Romney voters will agree or disagree with what the current speaker is saying. To do this, we calculated the ratio of reactions agreeing with the current speaker to the reactions disagreeing with the current speaker for each turn in the debate. Each turn in the debate was then labeled with either a 1 if the ratio of agrees to disagrees for that turn was greater than median for all turns or 0 otherwise. This was done once for reactions from Democrats and a second time for reactions from Republicans.

3.2.3 Task 3: Predicting Spin and Dodge Reactions

The purpose of **Task 3** is to evaluate how well-suited the features in each model is to predict if Obama voters or Romney voters will judge the current speaker to be spinning or dodging. To do this,

Top Terms	Description
president america country states united didn made today respect difference american don fact mistake make policy decisions question kids decision	Incoherent
campaign people country foreign american fact issues town lobbyists house congress tough white honor running campaigns interested pledge john character	Campaigning
health care plan insurance medicare costs cost government give companies **system** buy seniors choice program lower billion **private** provide **premiums**	Healthcare
drugs don act **law** congress rights bill **crime** line **drug police protect** patriot border **enforcement** legislation citizens fact enterprise **fighting**	Law Enforcement
ve people don make ll time back work things years put lot good country america important american president point thing	Incoherent
iraq afghanistan senator **troops strategy** obama **pakistan russia war georgia qaeda al** mccain **military** situation states understand russians **defeat** united	Foreign Conflict
jobs trade free agreement **job** country fair **industries workers overseas business** standards base defense agreements wage century **pay** minimum growing	Jobs
governor romney government **federal states** board approach **obamacare repeal conditions** fact reason **replace** difference **cost** military investments crisis opportunity **Massachusetts**	Health Insurance Reform / Pre-existing conditions
world war military troops national security forces army peace russia europe **nuclear draft** general question **defense cold** union superpower democracy	Defense
nuclear iran north korea president **weapons talks** threat senator united mccain **proliferation** sanctions countries israel involved china table states ambassador	Nuclear Weapons
education school schools child money **children kids** america system job **teachers** state **public** abortion program funding college continue choice aids	Education
mr president **question senator minute** perot bush governor kerry **minutes** clinton seconds **answer audience tonight debate questions** presidential candidates sir	Debate Phrases
senator obama voted spending senate states united record party opposed constitution fought reform bill marriage friends times issue justice completely	Legislative Branch
iraq war world **saddam hussein** plan troops **weapons** free opponent **bin osama laden terror** win wrong safe threat strong intelligence	Middle East / Terrorism
energy oil nuclear fuel percent **technology power** reduce stem united companies **coal drilling clean wind** states **solar science** mccain issue	Energy
congress people economic years mr government american governor economy clinton control growth spend social program change country interest programs jobs	Clinton Years Economy
tax taxes cut jobs billion spending people **middle** small **percent** plan pay **class** america **raise** budget **business** money **income** deficit	Taxes
mccain sen senator obama crisis **economy street financial** question **banks wall policy** regulation **economic** homes americans tonight system package fix	Economy
children people women family person **love** american **faith dream life values** great **woman** strong part **laughter god personal wife daughters**	Pathos

Figure 5: LDA topics (descriptions were determined manually by the author)

Features	DecTree	MaxEnt	Naive Bayes
Unigram	0.65	0.57	0.57
Bigram	0.40	0.60	**0.60**
Manual Labels	0.70	0.50	0.45
LDA Topics	**0.71**	**0.70**	0.20

Table 1: Accuracy on Task 1 (Obama voters)

Features	DecTree	MaxEnt	Naive Bayes
Unigram	0.58	0.68	0.67
Bigram	**0.73**	**0.73**	**0.73**
Manual Labels	0.72	0.39	0.39
LDA Topics	0.71	0.68	0.20

Table 2: Accuracy on Task 1 (Romney voters)

we calculated the number of spin and dodge reactions for the current speaker for each turn in the debate and divided by the length of the turn in seconds. Each turn in the debate was then labeled with either a 1 if spins and dodges per second for that turn was greater than median for all turns or 0 otherwise. This was done once for reactions from Democrats and a second time for reactions from Republicans.

4 Results

We predict the labels described in all three tasks tasks for each turn in the debate using Decision Tree, Maximum Entropy and Naive Bayes classifiers. We measure our final accuracy on all tasks (displayed in Tables 1 through 6) with 10-fold cross validation (with 90% of turns in the training set, 10% in the test set).

The results show that although unigram features performed very well throughout, no one set of features outperformed the others for every task. Encouragingly, classification accuracy with LDA topic features frequently approached manual and occasionally *exceeded* classification accuracy using manual labels as features.

Features	DecTree	MaxEnt	Naive Bayes
Unigram	0.74	**0.76**	**0.87**
Bigram	0.46	0.54	0.44
Manual Labels	0.52	0.58	0.59
LDA Topics	**0.76**	0.74	0.23

Table 3: Accuracy on Task 2 (Obama voters)

Features	DecTree	MaxEnt	Naive Bayes
Unigram	**0.83**	**0.80**	0.50
Bigram	0.63	0.63	**0.63**
Manual Labels	0.46	0.56	0.57
LDA Topics	0.70	0.71	0.21

Table 4: Accuracy on Task 2 (Romney voters)

Features	DecTree	MaxEnt	Naive Bayes
Unigram	**0.83**	**0.86**	0.81
Bigram	0.50	0.50	0.33
Manual Labels	0.80	0.82	**0.82**
LDA Topics	0.74	0.70	0.22

Table 5: Accuracy on Task 3 (Obama voters)

4.1 Ngram Features

To determine the number of n-gram features to use to avoid overfitting, we vary their number while evaluating mean accuracy during repeated random sub-sampling validation. To select which n-grams to include among the features, we select the most frequent n-grams first. We also remove very short turns (less than thirty words long) from the set of examples for training and testing.

First we consider results with unigram features. In Tables 1 and 2 we see on that **Task 1** all three classifiers performed similarly. This task is challenging for the classifiers, for example with the Naive Bayes model we see in Fig. 6 that test accuracy quickly reaches a limit as we consider any more than a few hundred unigram features and the model quickly overfits due to the relatively high numbers of features per training example.

In contrast, the models are substantially more successful on **Task 2**. Interestingly the Naive Bayes model performed very well when predicting reactions of Obama voters, cf. Fig. 7 – the most informative features were *Romney* and *Governor*, perhaps because the words are more often said by their cho-

Features	DecTree	MaxEnt	Naive Bayes
Unigram	0.80	**0.84**	0.49
Bigram	0.60	0.40	0.60
Manual Labels	**0.81**	0.81	**0.81**
LDA Topics	0.77	0.71	0.26

Table 6: Accuracy on Task 3 (Romney voters)

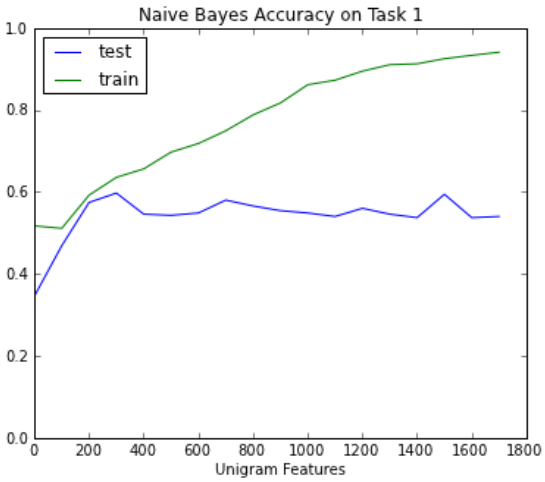

Figure 6: Unigrams hyperparameter tuning for Task 1 Naive Bayes for Obama supporter reactions.

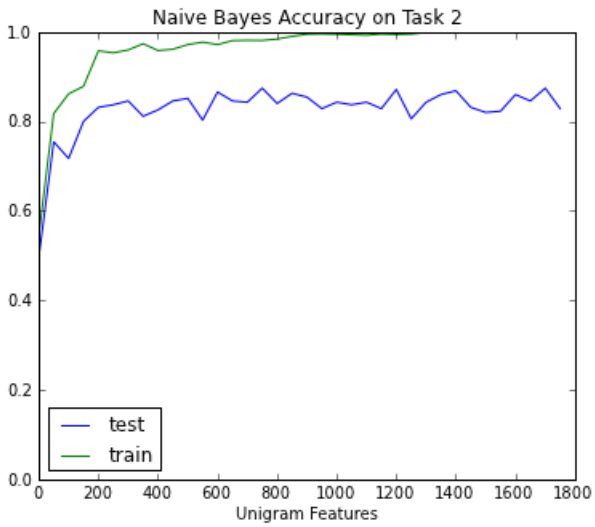

Figure 7: Unigrams hyperparameter tuning for Task 1 Naive Bayes for Obama supporter reactions.

sen candidate than the opponent. But, it did not perform well for Romney voters, cf. Table 4 – here the most informative feature was *idea*, which doesn't seem to have as much clear meaning as the name of an opponent.

Something that may help to explain this difference is an interesting difference between the overall reactions of the Obama supporters and Romney supporters over all turns. In Fig. 8, we see frequency distribution of ratios of reactions in agreeing with Obama to the reactions disagreeing. There are four clear modes in this distribution. From left-to-right on the x-axis, the first mode reflects turns in which Obama supporters strongly *disagreed* with what was being said, the next two modes reflect turns in which they slightly disagreed or slightly *agreed*, respectively, and the right-most mode represents turns when Obama supporters agreed very strongly.

Now contrast this with the corresponding distribution of Romney supporter reactions in Fig. 9. The patterns that emerges is that the modes in the distribution representing strongest disagreement is absent from the Republican supporter reactions and the mode representing strong support is higher (in relative but not absolute numbers, because the number of Obama supporters outnumber the Romney supporters). As a consequence, there is a greater bias in the training sets for the Naive Bayes classifier on Task 2, which – in combination with a small training

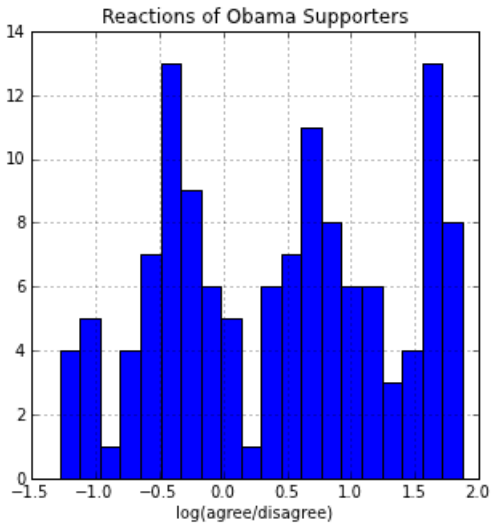

Figure 8: The ratio of agreeing reactions to disagreeing reactions among Obama supporters throughout the debate.

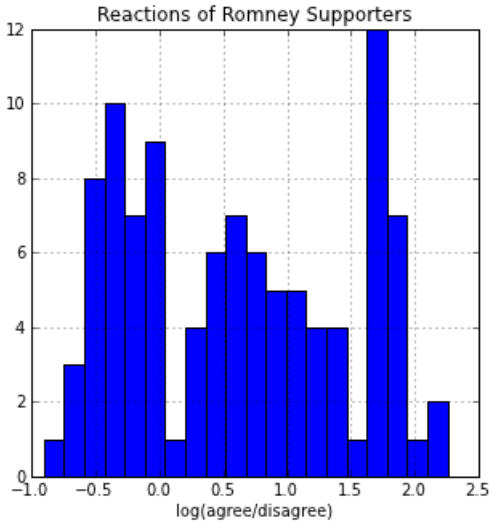

Figure 9: The ratio of agreeing reactions to disagreeing reactions among Romney supporters throughout the debate.

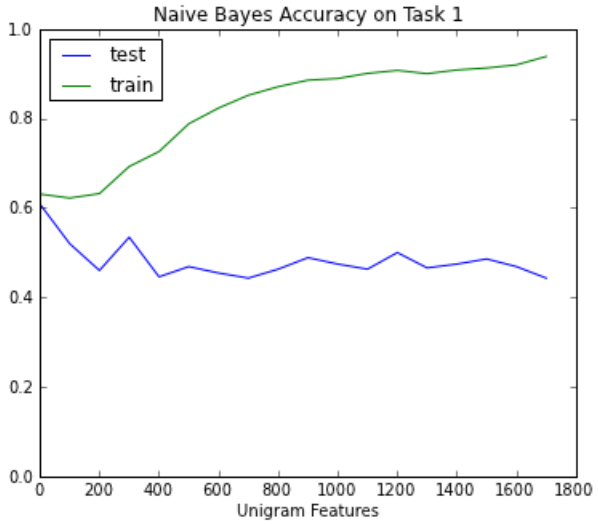

Figure 10: Unigrams hyperparameter tuning for Task 1 Naive Bayes for Romney supporter reactions.

size over-all – can lead to a tendency to predict the common class regardless of the features. Another reason for the difference in performance may be the difference in numbers of Obama supporters and Romney supporters in the data set, such that there is a clearer signal of consensus over topics (here, unigrams) that Obama supporters react to, and a tendency to over-fit. Indeed, we see in Fig. 10 that the Naive Bayes model over fits easily as we increase the number of unigram features.

Continuing, we see in Tables 5 and 6 that on **Task 3**, Maximum Entropy performed best overall while Naive Bayes continued to struggle at predicting reactions of Romney voters.

Bigram features consistently underperform the unigram features, which is not surprising since there were very few training examples. This caused overfitting if we increase the number of bigram features, and the models are not smoothed. Tables 1 and 2 reveal poor performance on **Task 1** for Obama supporters by all classifiers; interestingly, Romney supporters' reactions were more easily predicted in this setting. The most informative bigram feature to predict strong reactions from Romney supporters on **Task 1** was *cut taxes*, which is easily understood as a meaningful collocation in general and for this group in particular. The most informative bigram to predict low reactions is *Governor Romney*, perhaps because this is most often said by the moderator or the other candidate rather than Romney himself. Bigram features continued to performs poorly on **Task 2** and **Task 3**, cf. Table 3 through Table 6.

4.2 Manually Labeled Topic Features

The Decision Tree classifier performs by far the best on Task 1, with 70% accuracy on reactions by Obama supporters and 72% accuracy on reactions by Romney supporters. On Task 2, all classifiers perform surprisingly poorly, with the best accuracy being 59% using a Naive Bayes classifier. The best performance is on Task 3, on which all three classifiers score in the low 80% range. The MaxEnt and Naive Bayes classifiers score the highest with 82% and 81% accuracy on reactions by Obama and Romney supporters, respectively.

Inspecting the information gain for features provides unique insight into user reactions. Task 1 involves predicting the overall volume of reactions based on the mixture of topics. The features with the highest information gain on task 1 give us an idea of what topics viewers were most passionate about. For Obama supporters, the features with the highest information gain were labor, employment, immigration, education, and health. For Romney supporters, the features with the highest information gain are government operations, macroeconomics, and education.

Predicting “spin" and “dodge" reactions in Task 3 is akin to predicting when viewers think they are being deceived. As it turns out, the “candidate personal information" label exhibits high information gain for both candidates. It is reasonable to conclude that when a candidate tells a personal anecdote, viewers from both parties believe that the candidates are straying from the topic they should be addressing. For example, the following quote is Obama's response to a question about Social Security (bold text is labeled as "candidate personal information"):

"...I want to talk about the values behind Social Security and Medicare and then talk about Medicare because that's the big driver of our deficits right now. **You know, my grandmother, some of you know, helped to raise me. My grandparents did. My grandfather died awhile back. My grandmother died three days before I was elected president. And she was fiercely independent.** ..."

We did not perform the same post-hoc analysis of features for Task 2 since the performance was so poor compared to the other tasks.

4.3 LDA Topic Features

Much to our surprise, using LDA topic proportions occasionally outperformed the manual topic labels for task 1 and task 2 with the Decision Tree and Maximum Entropy classifiers as shown in Tables 1 and 4. Although it was more than adequate for task 1 and task 2, it fell short of the coded topic features for task three (Tables 5 and 6).

The discrepancy between the accuracy achieved with Naive Bayes (where LDA features performed abysmally) and the accuracy achieved other classifiers is revealing. Decision trees split the dataset on an attribute that gives them the maximum information gain (in this case, this was topic 0, which I claimed was incoherent), so it has a mechanism using features in order of importance. Additionally, maximum entropy classifiers try to be as agnostic as possible about any attributes that aren't particularly helpful for classification. Naive Bayes, however, attempts to use the probability of each feature given a class label to compute the posterior probability of that class. It is clear that the choice of classifier can greatly impact the success of LDA features as incoherent topics learned from LDA can detract substantially from the overall accuracy.

LDA features show some promise since they performed better than even unigram features for Task 1 for Democrat reactions in both Decision Trees and Maximum Entropy models. They likely outperform unigram features with decision tree classifies because the dimensionality of the feature space is much lower. In a decision tree classifier, each time the tree is split, there are fewer training examples for each branch. It seems plausible that LDA is merging the unigram features into fewer and more informative splits.

5 Improving upon LDA Topics

Given that LDA topics performed, perhaps unexpectedly, competitively, it is prudent to investigate how we might improve upon LDA topics to push classification accuracy even higher. There are many different possible avenues to explore. For example, Nguyen et al. (Nguyen et al., 2012) incorporate *speaker identity* into a topic model similar to LDA in order to segment topics over the course of a debate and determine when speakers are changing the subject. Since this model is significantly more sophisticated than what we have already tried, it makes sense to begin with more incremental steps. Given that LDA only discovers a fixed number of topics, a more natural next step is to investigate its nonparametric cousin, the Chinese Restaurant Process (CRP) and a version of the CRP that incorporates temporal locality, the Distance Dependent Chinese Restaurant Process (ddCRP). Why might we want to incorporate temporal locality into our model of the debate? Debates are highly organized. Unless one of the debaters chooses to dodge a question, the discussion is likely to stay on a topic set by the moderator for a while

5.1 The Chinese Restaurant Process

The CRP is a nonparametric Bayesian generative model used to model topics in a corpus of documents where the number of topics is not known beforehand. The CRP is typically explained with an analogy to Chinese restaurants where “customers" (words) sit at “tables" (topics). Any given customer may choose to sit at an existing table with probability proportional to the number of customers already

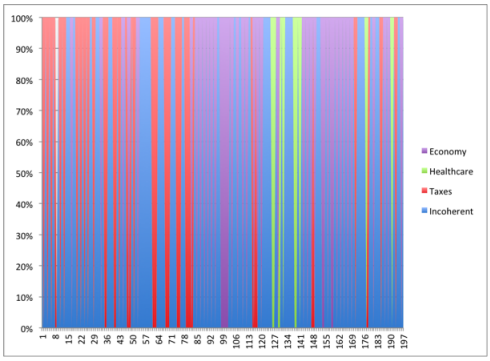

Figure 11: Topics over time from CRP.

sitting at that table, or at a new table with probability proportional to a dispersion parameter (stated formally in equation 1 where n_k is the number of customers already sitting at table k).

$$p(z_i = k | z_{1:(i-1)}, \alpha) \propto \begin{cases} n_k & \text{for } k \leq K \\ \alpha & \text{for } k = K + 1 \end{cases} \quad (1)$$

5.2 The Distance Dependent Chinese Restaurant Process

Unlike the CRP, which uses an analogy of customers sitting at tables to describe the probability of a seating arrangement, Blei and Frazier describe the ddCRP in terms of customers sitting with other customers (Blei and Frazier, 2011). Let c_i be the customer who the ith customer chooses to sit with, D be a matrix of distances between customers, and f be a decay function. The probability that customer i sits with customer j is calculated according to equation 2. Customer i sits with no one (itself) with probability proportional to α, and sits with another customer with probability proportional to the value of f (a decay function) at the distance between i and j.

$$p(c_i = j | D, \alpha) \propto \begin{cases} f(d_{ij}) & \text{if } i \neq j \\ \alpha & \text{if } i = j \end{cases} \quad (2)$$

5.3 Results

We ran a Gibbs sampler for 20 iterations for both the CRP and ddCRP with $\alpha = 1$ and $\lambda = 0.5$. To control for as much as possible, we used the same sampler for both the CRP and ddCRP with

Top Terms	Description
two president let right governor minutes go going first government segment yeah romney economy well jim ok federal said president	Debate
tax get people taxes cut want make jobs governor deficit said romney business trillion years well plan small going president	Taxes
people medicare going care insurance health plan let get government one said make way governor cost right state could number	Healthcare
regulation banks know going make got romney big governor on street loans wall sure say qualified economy said repeal party	Economy

Table 8: Topics from CRP (descriptions were determined manually by the author)

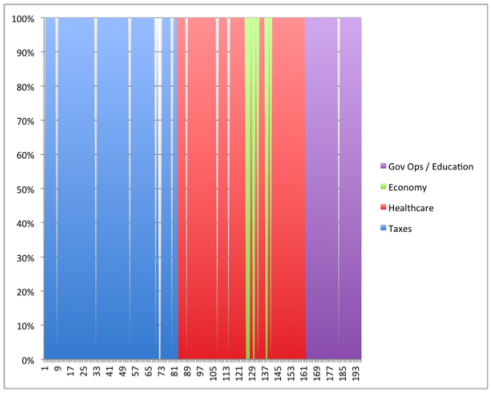

Figure 12: Topics over time from ddCRP.

Top Terms	Description
tax get taxes people cut president going said make well governor deficit want got plan jobs trillion right romney let	Taxes
medicare going people care let insurance health plan governor two said get right way one cost make government president romney	Healthcare
people government going governor president education federal america right romney know one make state go well cut look get schools	Marginal
regulation banks big qualified make streed know wall mortgage one economy got hurt repeal romney loans become excessive let	Economy

Table 9: Topics from ddCRP (descriptions were determined manually by the author)

different decay functions. This is possible because the ddCRP where $f(d_{ij}) = 1$ is equivalent to the CRP. We used a logistic decay function ($f(d_{ij}) = logistic(-d_{ij} + 2)$) for the ddCRP. A low number of iterations is acceptable for our sampler because table splits and merges allow customer partitions to change dramatically at each resampling step which tends to make the sampler converge very quickly.

The CRP inferred 5 topics while the ddCRP inferred 16. The four most common topics for both are shown in Tables 11 and 12. For the most part, the topics are quite similar, but there is one major difference. The CRP had one very common topic that was comprised of debate jargon. We can see why this is by looking at figure 11. The debate jargon topic (labeled incoherent) is spread throughout the debate rather than in one contiguous segment.

Figures 11 and 12 show the topics associated with each turn in the debate. As expected, CRP topics are scattered somewhat haphazardly while ddCRP topics are clearly segmented in time. Quantitatively, the ddCRP model reached a greater log likelihood (-52934.77) than the traditional CRP (-53377.62). Although the ddCRP does exhibit some nice qualities, it is unclear whether its topics are really better overall than the regular CRP or even LDA. Worse, there were many more incoherent topics (likely the result of the single “debate jargon" topic from the CRP split into smaller chunks), which is obviously undesirable.

6 Discussion

6.1 Results

Predicting user reactions yielded some very promising results. Although the unigram features baseline frequently outperformed the other features, they did not always achieve the greatest accuracy. Most importantly, the classification accuracy with LDA features was very competitive with the accuracy achieved with hand-coded features. Furthermore, there are ample ways to potentially improve automatic topic labels with more sophisticated models than LDA, such as the CRP and ddCRP. This is very interesting and worth further investigation given that manual coding is so expensive.

6.2 Future Work

An obvious next step is to retry the classification experiments with topic features from the CRP and ddCRP. From there, it might be interesting to try more complex topic models. It would be interesting to see if you get better topics and/or predictions if the supervised learning happens in conjunction with the topic modeling as in supervised LDA. It would also be interesting to see if hierarchical topics are more useful for classification with a model like the Hierarchical Dirichlet Process. Finally, it would be very interesting to see if topic shift indicators inferred by the SITS model would improve dodge reaction predictions.

Another direction worth pursuing would be to repeat the experiments with finer grained subsets of the users. Since the data from React Labs : Educate contains detailed demographic information about its users, it is possible to hone in on very specific groups of people.

References

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, page 69–72.

David M. Blei and Peter I. Frazier. 2011. Distance dependent chinese restaurant processes. *The Journal of Machine Learning Research*, 12:2461–2488.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.

Amber E Boydstun, Rebecca Glazier, Jessica T Feezell, Timothy Jurka, Matthew Pietryka, and Jack Reilly. 2014a. Colleague crowdsourcing: A method for incentivizing national student engagement and large-n data collection. In *APSA 2013 Teaching and Learning Conference Paper*.

Amber E Boydstun, Rebecca A Glazier, Matthew T Pietryka, and Philip Resnik. 2014b. Real-time reactions to a 2012 presidential debate: A method for understanding which messages matter. *Public Opinion Quarterly*, 78(S1):330–343.

Justin Grimmer and Brandon M Stewart. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*.

David Lazer, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. 2009. Life in the network: the coming age of computational social science. *Science (New York, NY)*, 323(5915):721.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit.

Viet-An Nguyen, Jordan Boyd-Graber, and Philip Resnik. 2012. SITS: a hierarchical nonparametric model using speaker identity for topic segmentation in multiparty conversations. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, page 78–87.

Brendan O'Connor, David Bamman, and Noah A Smith. 2011. Computational text analysis for social science: Model assumptions and complexity. *Public Health*, 41(42):43.