Identifying the horse race in elections

Miriam Boon
Technology and Social Behavior
Northwestern University
USA
MiriamBoon2012@u.northwestern.edu

Larry Birnbaum
Computer Science Division
Northwestern University
USA
l-birnbaum@northwestern.edu

ABSTRACT

Modern election coverage tends to focus on who is winning and what strategies campaigns are employing. With the ultimate goals of understanding the mix of story types from different venues, helping people to understand their news consumption, and recommending stories with more useful content, we explore methods for automatic classification of election news stories into a number of categories and sub-categories, including "horse race" and "policy."

CCS CONCEPTS

• **Computing methodologies** → Natural language processing • **Human-centered computing** → Natural language interfaces

KEYWORDS

Computer science, journalism, computational journalism, narrative frames, horse race

1 INTRODUCTION

An increasing proportion of election news coverage takes a "horse race" perspective [3, 4]. Horse race articles focus on who will win – performance – or on campaign strategy. These are the main components of horse race articles.

Horse race articles teach us to treat elections like a spectator sport, potentially promoting polarization and fatalism. Strategy articles in particular have been shown to promote cynicism [2], as they imply that strategic motivations are the only motivations politicians can hold.

We believe that there is value in providing readers with election news that minimizes horse race content, and maximizes informative content on policy and candidate qualifications. Doing so would serve our broader goal, which is to augment readers' media literacy. Towards that end, we trained and tested several classifiers, comparing their performance on a hand-coded data set.

2 METHOD

2.1 Data

For this work we elected to use the New York Times Annotated Corpus. NYTAC consists of 1.8 million articles written and published in the NYT between January 1, 1987 and June 19, 2007. Most articles have been manually tagged by a team of library scientists, and over 275,000 articles have been algorithmically tagged and then hand-verified.

In order to generate an election-focused sub-corpus, we identified a list of election-related tags. Because US coverage of foreign elections is not addressed to voters, we eliminated articles from the 'Foreign Desk'. We also ignored articles that were listed as having the type, "quote", "letter", "letters", and "letterletter". Finally, we identified a list of "stop tags" that identified types of election stories that did not address ongoing campaigns, such as "review", "obituary (obit)", or "referendum" (see Appendix). The resulting data set consists of 43,646 articles.

In order to prepare the data for classification and analysis, we used SciKitLearn's built-in TF-IDF Vectorizer, as well as a basic English stop words list.

2.2 Categories and coding

The first author and a research assistant coded a randomly selected sample of 829 articles, of which 101 overlapped. Based on that overlap, the intercoder reliability was 0.57.

Despite our efforts to eliminate undesirable data, some of the remaining articles in the data set covered unwanted topics, such as foreign elections, past elections, and election law and reform. Others were deemed too short to count as articles. These articles were categorized as not being "Relevant" and excluded from analysis; 435 relevant articles remained. These were categorized as "Policy", "Candidate", or "Horse Race" stories. They were further broken down into a total of 7 sub-categories. Under "Policy", "Economic Policy", "Social Policy", and "Policy – Other". Under "Candidate", "Biographical", and "Character". Under "Horse Race", "Performance" and "Strategy".

2.3 Classification and analysis

From a computer science perspective, we had the research question, "For categories and subcategories of this sort, is it better to classify into categories, and then within each category classify into subcategories, or is it better to classify into subcategories, and then infer categories based on subcategory?" We called these the hierarchical and inferential classifiers, respectively.

In each case, we have to ask these questions twice: Once to determine which approach yields the best results at the category level, and again to determine the best approach for subcategory performance.

We used cross-validated grid search to test a variety of potential parameters with four different algorithms available through the Python package SciKit-Learn [5]: Multinomial Naïve Bayes, Perceptron, Passive Aggressive Classifier, and the Stochastic Gradient Descent Classifier. For each classification problem we used the classifier that performed best.

To ensure that results were independent of the randomly selected training set, we used stratified k-fold cross-validation. Stratification distributed instances evenly across categories, and *k* ranged from two to three depending on the amount of data we were working with. This process was repeated 100 times, randomizing the partitioning of the folds each time.

3 RESULTS AND DISCUSSION

The data set consists of 214 "Horse Race" stories, 146 "Candidate" stories, and 75 "Policy" stories. A classifier that always predicted a horse race story would yield an average precision of 0.164, and an average recall of 0.333, which makes the performance shown in Table 1 a significant improvement over that baseline.

Table 1: Performance at Category Level

Method	Precision	Recall	F1-Score
H	0.683 +/- 0.049	0.630 +/- 0.048	0.645 +/- 0.047
I	0.667 +/- 0.047	0.595 +/- 0.042	0.613 +/- 0.044

The data set includes 130 "Strategy", 126 "Character", 84 "Performance", 39 "Social policy", 22 "Economic policy", 20 "Biographical", and 14 "Other policy" articles. Again, if we always predicted the most common subcategory, strategy, we'd expect a much lower precision and recall (0.0427 and 0.143 respectively). See Table 2 for the performance of our actual classifiers, which significantly outperform this baseline.

Table 2: Performance at Sub-Category Level

Method	Precision	Recall	F1-Score
H	0.404 +/- 0.102	*0.355 +/- 0.047*	0.355 +/- 0.056
I	0.423 +/- 0.084	*0.356 +/- 0.035*	0.367 +/- 0.038

All differences between hierarchical and inferential models are significant except recall at the sub-category level.

The first step for the hierarchical classifier is to classify at the category level. The first step for the inferential classifier is to classify at the sub-category level. In each case, these classifiers perform better than the other on their first step, and worse on their second step.

In the case of the inferential classifier, performance is entirely due to the efficacy of the sub-category classification, as the inferential process is perfect. Improving the inferential classifier depends entirely on improving its sub-category classification. In the case of the hierarchical classifier, however, poor performance can be ascribed only in part to cascading error. Improving the category-level classifier would improve both category classification and sub-category classification, and improving the sub-category classifiers would improve sub-category classification.

4 APPLICATION

In order to demonstrate the utility of this technology, we also gathered a data set of election-related stories from the last month of the 2016 presidential election. Publications included were the New York Times, USA Today, NY Daily News, ABC News, Washington Post, CBS News, and Reuters. We then applied the category-level algorithm to classify the stories into the categories of "Horse Race", "Candidate", or "Policy". The results can be seen in Figure 1.

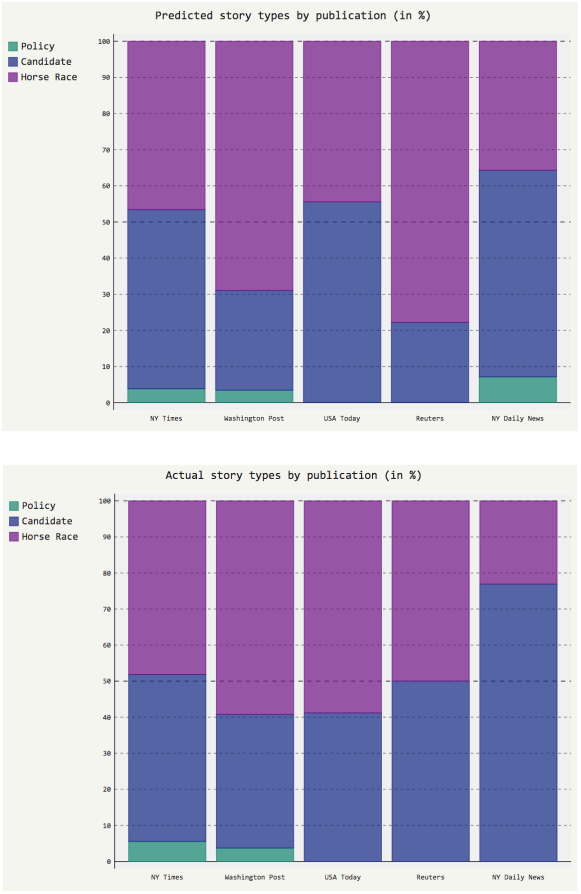

Figure 1: Top: Our algorithm's classification of stories from each source. Note that some sources published in excess of 200 stories during this time, while others published too few to classify. Bottom: Actual classification of 119 stories from each source, as coded by hand.

Because we hand-coded so few of these stories (119), any conclusions we draw should be taken with a grain of salt. You may note that the algorithm performed exceptionally well for the New York Times; this follows as the algorithm was trained on the New York Times. It is also true that the New York Times had the largest number of stories in the hand-coded data set (54). The Washington Post had half as many stories (27), and also performed quite well; this could be due to the similarity in style to the New York Times. USA Today, Reuters, and NY Daily News each had 17, 4, and 13 stories in the data set respectively, and

each ranged from poor to mediocre performance. It may be that the set simply wasn't large enough, and that the writing styles, at least in the case of USA Today and NY Daily News, were too different.

5 FUTURE WORK

5.1 Algorithm improvement

Performance was likely impacted by the small sample size. When split into folds, some of the sub-categories had as few as seven samples to learn from; this is far too few. Future work must necessarily include a data set of sufficient size to train more effectively.

We must also consider the possibility that there is a problem with how we coded the data. We treat all categories as mutually exclusive. But some articles have a topic, such as policy, while presenting the material in "horse race" terms. In other words, this category behaves more like a narrative frame than a topic. Examination of all categories suggested one other potential frame, which we call "Character". A wide variety of topics can be discussed in terms of how they reflect a candidate's character. Dimensions like these are pragmatic, rather than semantic or syntactic. That is, they are not really characterized by the meaning of the words used, or by the structure of the grammar.

Based on this analysis, we are in the process of coding a larger data set that codes the presence of frames regardless of the presence of topics. This will allow future models to take these insights into consideration.

5.2 Technological application

There are three possible applications of this technology. First, as demonstrated earlier, we can analyze election coverage from a variety of venues to see how well they resist the tendency to publish "horse race" stories. Second, we could create an election news aggregator that filters or minimizes "horse race" content. Third, we could use the News Context Project [1] to create a plugin that detects "horse race" style coverage, and warns the user. This last option is what we plan to do next, as it aligns with our broader goal of augmenting media literacy.

APPENDIX

The following tags were tracked:
'elections,primaries', 'debating,elections', elections, presidential elections (us), election issues, presidential election of 2000, presidential election of 2004, presidential election of 1996, presidential election of 1992, presidential election of 1988, presidential election of 2008, presidentialelection of 1996, electionissues, presidential elections, presidential_election_of_1996, presidential_election_of_2000, election_issues, presidential election of1996, presidential electionof 1996, presidential election (us), presidentialelection of 1992, presedential election of 1988, presidential election of 1996, presidential elections of 2004, presidential election of 1996, pesidential election of 1988, presidential electionof 1988, presidential election of 1996, presidentialelection of 1988, election, presidentialelections (us), presidential electionof 1992

The following tags were excluded:
referendum, referendums, caption, correction, paid death notice, review, obituary (obit), schedule, list, paid memorial notice, obituary, editors note, editors note, chronology, obituary(obits), glossary, reviewreview, corrections, presidential election of 1980, ad campaigns, election results, armament, defense and military forces, 'elections, public financing of', 'election results,editorials', politics and government, united states politics and government, election results, decisions and verdicts, presidential election of 1984, presidential election of 1948, presidential election of 1968, presidential election of 1960, presidential election of 1972, presidential election of 1976, presidential election of 1964, presidential election of 1876, presidential election of 1952, presidential election of 1940, presidential election of 1924, presidential election of 1800, presidential election of 1936, presidential election of 1836, presidential election of 1916, presidential election of 1932, presidential election of 1908, presidential election of 1828, presidential election of 2012, recall (elections), campaign buttons and posters, voting rights act of 1965, voter registration and requirements, registration of voters

ACKNOWLEDGMENTS

TSB, Google, NSF under grant number 0917261

REFERENCES

[1] Birnbaum, L., Boon, M., Bradley, S. and Wilson, J. 2015. The News Context Project. *Proceedings of the 20th International Conference on Intelligent User Interfaces Companion* (New York, NY, USA, 2015), 5–8.

[2] Cappella, J.N. and Jamieson, K.H. 1997. *Spiral of Cynicism: The Press and the Public Good*. Oxford University Press.

[3] Jamieson, K.H. and Waldman, P. 2004. *The Press Effect: Politicians, Journalists, and the Stories that Shape the Political World*. Oxford University Press.

[4] Lichter, S.R. 2001. A Plague on Both Parties: Substance and Fairness in TV Election News. *Harvard International Journal of Press/Politics*. 6, 3 (Jun. 2001), 8–30. DOI:https://doi.org/10.1177/108118001129172206.

[5] Pedregosa, F. et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 12, (2011), 2825–2830.