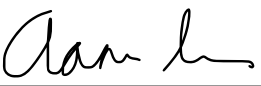


Master's Paper of the Department of Statistics, The University of Chicago
(Internal departmental document only, not for circulation. Anyone wishing to publish or cite any portion therein must have the express, written permission of the author.)

Evaluating the Effect of Round Type on College Policy Debate Outcomes

Austin Koort

Advisor(s): Aaron Schein

Approved 

Date 05-02-2024

May 03, 2024

Abstract

In intercollegiate policy debate, two teams argue in front of a judge who decides the winner. The affirmative team argues in support of a provided resolution, and the negative team in opposition to it. Within the debate community, teams and judges are thought of as belonging to one of three categories based on their argumentative preferences. Each combination of judge and team categories creates a category for debate rounds. Although there has been some community discussion of side bias, or the skew between affirmative and negative win rates, there has been no statistical analysis of the effect of team and judge categories on round outcomes. Reverse engineering the Mutual Preference Judging algorithm, I create and fit a variational inference model to infer latent team and judge categories from pairings of teams and judges. Then, I use Microsoft’s TrueSkill Through Time rating algorithm to evaluate the effect of these categories on debate outcomes. Ultimately, I find one particular style of debating results in consistently better competitive results.

Contents

1	Introduction	3
2	Background	4
2.1	Policy Debate	4
2.2	The Debate Tournament	5
2.3	Mutual Preference Judging	6
2.4	The Structure of Team Preferences	10
3	Data Collection	14
4	Learning Team and Judge Ideology	17
4.1	Potential Approaches	17
4.2	Variational Inference	18
4.3	Model Design	20
4.4	Implementing the Model	22
4.5	Experimentation and Results	25

5	Estimating Effect on Round Outcome	29
5.1	Skill Estimation	29
5.2	Experimentation and Results	34
6	Conclusion	38

1 Introduction

Intercollegiate Policy Debate traces its history back to the 1800s. After the success of “oratorical contests” across the country, in 1892 Harvard and Yale students decided to stake the pride of their universities on an explicitly debate-centric event [30]. This event was such a success that by the mid 1890s, “debating societies” at universities across the country were replicating it. They would jointly organize their own public debates over chosen, politically relevant topics [38]. With the continued success of these imitation events, the idea of intercollegiate policy debate began catching on nationally. By the end of the 1920s, competitive policy debate started to become an officially run co-curricular. Rather than one-off or strictly annual events organized between clubs at two universities, colleges began hosting multi-day debate tournaments which drew competitors from other nearby universities [10]. Over time, the rules and structure of these tournaments would slowly standardize, eventually evolving into a form recognizable to modern participants.

Although far from its heights in the 1970s and 1980s, when the largest tournament of the year could host upwards of 1000 debaters [25], Intercollegiate Policy Debate remains a major activity nationally. Over the past decade, roughly 100,000 debates have taken place between over 8,000 teams¹. However, despite being a highly competitive and technical activity, very little quantitative analysis of debate strategy has ever been performed. Most analyses suffer from either a lack of scale, focus strictly on side bias², or are restricted to high-school-level competition [15][16]. Although potentially useful for the committees that determine the topic for a year, information about side bias does little to inform *competitors* of what they should do. My goal is to generate actionable information for debate teams. To do so, I attempt to classify debate teams and judges based on their argumentative preferences. Then, I attempt to determine the effect of these preferences on competitive outcomes. The goal is to inform both both choice of argument and choice of judge for debate teams. For more specifics, it is necessary to first discuss the setting.

¹Teams consist of two debaters (most always) from the same university.

²Side bias refers to the effect of side selection on round outcomes.

2 Background

2.1 Policy Debate

In Intercollegiate Policy Debate, a single “resolution” is picked by committee the summer before each academic year and serves as the topic for that entire year. There are three semi-distinct governing bodies for Intercollegiate Policy debate, the National Debate Tournament (NDT), the Cross Examination Debate Association (CEDA), and the American Debate Association (ADA). Since 1996, they have agreed to help standardize Intercollegiate Policy Debate by sharing the same resolution [25]. Recently, these resolutions typically outline a list of policies the United States government could enact. For example, this year’s resolution reads:

Resolved: The United States should restrict its nuclear forces in one or more of the following ways:

- adopting a nuclear no-first-use policy;
- eliminating one or more of the legs of its nuclear triad;
- disarming its nuclear forces.

Hypothetically, the year’s resolution acts as the focal point for every debate between two teams, termed a “debate round,” in a given academic year. Typically, in any given round, the “affirmative” team defends the resolution’s desirability against the “negative” team, who argues against them. A judge, or panel of judges, takes careful notes and decides a winner.

However, what it means to affirm and negate the resolution has changed substantially over the past 120 years. For much of the 20th century, debates involves the whole of the resolution. Under this model, the affirmative team would win if they could prove that, on balance, adoption of the resolution would have a positive effect. For about the past 50 years, however, plan-focus models of debate have largely dominated [23].

Under the plan-focus model, each debate round is a referendum on the passage of “topical” policy provided by the affirmative team. This policy is referred to as the “plan.” For example, a plan under the current resolution could read: “the United States should eliminate its ICBMs.” Then, the affirmative team’s only burden is to explain why their plan is desirable. The plan-focused approach to debate typically referred to now by the community as simply “policy debate,” framing it as the standard mode of participating in the activity of the same name. Under this model, the rest of the resolution becomes functionally irrelevant once the affirmative introduces their plan. This has evolved into a very technical procedure, full of jargon, complex interactions between arguments, and places a very heavy research burden on both teams should they wish to be competitive.

In the 1990s, a new model of debate was introduced with the “kritik” (pronounced “critique”). Rather than directly contesting the implementation of a plan, kritik negatives con-

test the affirmative from a philosophical perspective [7]. Rather than treating the affirmative team’s plan as a policy solution to immediately act, they contest the entire first affirmative speech as an object of research. For example, an IR kritik could assert that the affirmative’s international relations scholarship should be rejected because it flattens understandings of non-Western states , or a gendered language kritik might argue that the affirmative should be rejected for using sexist metaphors that devalue women. Rather than an uncritical affirmation or negation of the resolution, the “k debate” approach is that individuals should find ways to be ethical actors in relation to it.

By the 2000s, there were very competitively successful teams reading kritiks of the resolution on the affirmative side. For example, in 2001-2002 season, the resolution read:

Resolved: The United States federal government should substantially increase federal control throughout Indian Country in one or more of the following areas: child welfare, criminal justice, employment, environmental protection, gaming, resource management, taxation.

Rather than defend the resolution on the affirmative, the University of Louisville debaters, often credited as the biggest innovators in alternative styles of debate in the 2000s [34], instead avocated that the United States should pay reparations to Native Americans. I want to iterate that the examples of kritikal debate are depicted in an oversimplified manner in order to keep things moving along. This is also a highly developed, complex approach to Intercollegiate Policy Debate. Kritik research involves parsing and utilizing huge quantities of incredibly dense academic philosophical literature that undergraduates are otherwise rarely exposed to.

Because they are typically considered disparate styles without much research overlap, teams generally specialize into either the “policy debate” or “k debate” style. The conventional wisdom is that there is a heavy incentive to specialize, as splitting your time means slower improvement and thus less competitive success. Usually, debaters also greatly prefer one style to the other, either because they believe it is more strategic or simply more fun. Rarely, “flex teams” will frequently switch between the two styles based on their opponent or judge.

2.2 The Debate Tournament

Intercollegiate Policy Debates occur at tournaments hosted by a university. Teams participating in tournaments first compete in preliminary rounds, which determine eligibility and seeding for elimination rounds. For each round, teams are “paired” together to debate and the debates are assigned judges.

Teams are divided into three categories based on their experience. Debaters with no high school or college experience in Policy Debate are eligible to enter the “Novice” division and teams with up to two years of college experience are eligible for “Junior Varsity.” The most competitive division is known interchangeably as “Varsity” or “Open.”

Typically, there are six to eight preliminary Swiss-system rounds, where teams are randomly matched against one another within their divisions based on their current number of wins. In the preliminary rounds, teams are assigned to the affirmative and negative side such that, by the end, they have affirmed and negated the resolution in equal proportion. With the exception of the National Debate Tournament, where every debate is decided by panel decision, preliminary rounds are decided by a singular judge. After the debate is over, they submit a ballot that determines the winner.

In the elimination rounds, pairings between teams are no longer random. Instead, preliminary round performs is used to seed a fixed, single-elimination bracket. Instead of being assigned sides by the tournament, teams flip a coin to determine who gets to choose whether to affirm or negate the resolution. Elimination rounds always have a panel of several judges. As in preliminary rounds, they individually submit ballots. Whichever team earns the majority of ballots wins the debate and advances up the bracket.

Unlike in other oratory activities, judges in Intercollegiate Policy Debate are almost never members of the general public. Instead, these judges are exclusively former debaters and university coaching staff. Because they are former debaters, and for similar reasons of interest and practicality, judges also tend to have more expertise in one style of debate than another. This has impacts on the desirability of difference judges for different teams, which eventually spurred the creation of the Mutual Preference Judging system.

2.3 Mutual Preference Judging

In the 1960s, judges were typically assigned to rounds completely at random. This eventually became quite unpopular, as it meant that important, high-quality debates were often decided by inexperienced judges or judges who had little investment in the subject matter of those debates. The next idea was then that the “best judges,” senior coaches with decades of experience, should judge the “best rounds.” By the early 80s, this was fairly common. As Gary Larson, one of the main architects of Mutual Preference Judging, describes of the 1990s,

Truly “random” judge assignment was rarely used at large tournaments... we typically had “tabroom”³ preference... In many cases, tabrooms were explicit (though not often publicly vocal) about their goal of having the “best” judges in the “best” rounds. In other cases the outcome was to assign judges “randomly” but to “fix” the egregious cases. In the worst case, judge assignments could directly favor teams associated with tabroom participants or elites whose interests would be “protected.” [12]

Because debate is a competitive activity, and so competitive equity is highly valued, even this system was untenable. The perception that the tabroom was possibly using its influence

³The term “tabroom” comes from the name of the room at a tournament where tournament organizers tabulate ballots. The phrase is colloquially used to refer to the tournament organizers.

to shift the scales towards one debate team over another was unacceptable to many, and also this semi-random approach still left many teams unhappy with their judges.

To address both these problems, a new judge assignment system called Mutual Preference Judging (MPJ) was introduced. Rather than completely-at-random assignment, each team would submit a secret ranking of judges called a “preference sheet,” with individual rankings colloquially shortened to “prefs.” Early on, this meant classifying each judge into three groups: “A,” “B,” and “C,” where “A” denotes highest preference and “C” lowest. Then, the tabroom would find the assignment of judges that maximized “mutual A” pairings, judge assignments such that both teams receive a judge they placed in their “A” group.

Although first experimented with at the National Debate Tournament in the mid 1980s, it was an incredibly difficult system to implement. It often involved crunch time for tabroom staff all through the night before the tournament, which prevented the possibility of widespread adoption. Early experiments with MPJ also involved lots of discretion from tournament organizers, who still manually decided the assignments among eligible choices.

During the 1990s, when the ideological dominance of plan-focus debate was beginning to fall into question, MPJ started becoming much more appealing to debaters due to the increase in nontraditional styles. As the 1990s drew to a close, debaters with nontraditional styles were becoming judges. With semi-random judging, this created big frustrations for debaters. K debaters were frustrated at being judged by policy judges who were unfamiliar with or dogmatically opposed to their arguments. Policy debaters were frustrated being judged by k judges, who viewed debate wildly differently than them.

When k debaters debated policy debaters, both teams often felt the victory was handed to whoever aligned closer ideologically with the judge. The obvious solution would be to adapt to the judge. However, this was not popular among debaters who both felt their particular form of debate was important and dreaded spending countless hours on research uninteresting to them.

These pressures, alongside technological developments which could automate much of the judge assignment process, meant that MPJ quickly became much more attractive. By the end of the 2000s, this system had been adopted by almost every tournament. However, this adoption was not without controversy. Despite its popularity among competitors, detractors worried that it would further serve to further fracture the Intercollegiate Policy Debate community [6] [36].

One worry was that MPJ would serve to reward debate programs with more resources and punish those with less. Universities with more resources devoted to debate had accumulated more institutional knowledge, had larger coaching staffs who were better socially connected, and simply traveled more. As such, they had a very good picture of the national pool of judges. Smaller and emerging programs were not as likely to draw well-connected, experienced coaches and often only had one full-time coach on payroll. More importantly, they simply didn’t have the resources to travel frequently. They would not know the hundreds

of judges at the National Debate Tournament well enough to rank them strategically. This also would potentially harm the judges and coaches at small and emerging programs. As unknown quantities to the community at large, they would potentially be ranked very low by the vast majority of teams.

The second worry was that MPJ would allow, and would strategically encourage, teams to operate within ideological echo chambers. Teams believe that their chance of winning, and their chance of getting valuable feedback, will be highest if they get judges ideologically aligned with them. As such, debaters would potentially be robbed of the educational benefits of having to argue in favor of things they might not believe in. Perhaps more importantly, they would not encounter conflicting viewpoints from a position of authority. A college student is far less likely to question their beliefs when questioned by another college student in a competitive setting than when confronted by an educator helping them understand why they failed to win.

Some detractors predicted these echo chambers would form along racial lines. As policy debaters, who made up the vast majority of teams, were disproportionately white and kritikal debaters were disproportionately people of color, it was not controversial to predict that MPJ would result in white teams very rarely being assigned non-white judges. Stereotyping non-white judges as biased towards the kritik would result in them never gaining the opportunity to prove their talent for judging policy rounds. As they would be ranked low by default, policy teams would never learn whether to change their mind about the placement. 20 years later, this is still believed to be a consequence of the adoption of MPJ [33].

However, this narrative does Mutual Preference Judging a disservice. Although not perfect, MPJ is what allowed teams with diverse argumentative styles and approaches to debate to succeed in the first place. When MPJ was introduced, judges amenable to non-standard argumentation were in the vast minority. Random assignment resulted in the crushing of these approaches, as adapting to the judge often meant complete conformity to the plan-focus status quo. With Mutual Preference Judging, adaption would remain important. However, teams would not have judges dogmatically opposed to their prepared strategies. This has broadened the possibilities of what Intercollegiate Policy Debate can be. After twenty years, the plurality of the national judge pool has become ideologically flexible enough to judge a wide variety of debates [34].

Since the inception of MPJ, there has also been a substantial reduction in racial bias on preference sheets. Whether this is because the community is diversifying, consciously addressing this structural issue, or some other reason entirely, it is empirically true that the gap in the sample means of preferredness between identity groups is closing. As of 2018, the largest gap between preferredness in identity groups is “significantly less than a standard deviation” [12]. However, this does not necessarily mean that polarization has decreased. The same investigation found a much higher sample standard deviation for preferredness of judges in minority identity groups than of majority identity groups.

As tournament organization has been digitized, the MPJ process has allowed to grow in

complexity. Since the early 2000s, rather than classifying judges to three categories, teams have been required to submit an ordinal list of every judge available at tournaments they attend. After it is decided who will debate whom, judges are assigned in order to maximize two criteria. First is “preferedness”, or how highly the teams rank a judge. Second is mutuality, or how similarly the teams rank a judge. The goal is, in essence, to maximize some score function $f(\text{mut}, \text{pref})$ with mutuality and preferedness as inputs. This is not a trivial problem. At last year’s NDT, for example, there were over 10^{220} potential permutations of judge assignments for just the first preliminary round. The exact form of this function is unknown. However, it is possible to gain a vague understanding of the algorithm that utilizes it. Gary Larson says in an interview for a debate audience [12]:

The computer makes a first pass, starting from the most important round and gets most highly advantageous pairings. That comes up with a pretty good answer, but not nearly a good enough one. And so the computer tests a variety of strategies where it backtracks and says “what happens if we swap these two judges, this judge into that round, this one to another.” So the computer tests, in an average round at the NDT, 16,000,000 permutations.⁴

In essence, concurrent debates are sorted by importance. In that order, each debate is assigned the best available judge. Judge assignments are permuted until some termination criteria is reached. Then, the best permutation is selected from those sampled.

Furthermore, permutations are performed within a restricted space. For teams who have not yet been eliminated from the tournament, judges must rank above some percentile cutoff for both teams in order to be considered. This ensures some minimum threshold of preferedness for important rounds. However, this cutoff is different for different debate tournaments and occasionally even different rounds.

Unfortunately, what it means for a round to be important not entirely clear. From the same interview, it seems that this generally means rounds with a greater effect on elimination round seeding are prioritized. However, this is not particularly informative. When multiple rounds have equal effect on elimination round seeding, it is unclear how they are weighted. It is also unclear how the effect on elimination rounds is determined in the first place.

Since the early 2000s, the MPJ system has remained relatively static in concept. However, the mechanics of the system allow for tournament organizers to tune the score function as they see fit. As an “optimal” assignment is a subjective mix of mutuality and preference, tournament organizers can increase or decrease the effect of either. In order to make tournaments run smoother and prevent exploitation of the system, several adjustments have been made to the Mutual Preference Judging process since its inception.

The first adjustment affects the preferences directly. Preferences are re-scaled based on the amount of available judging instead of number of available judges. Judges often are not signed up to judge every round of a tournament. For example, if there are 200 judges at

⁴Minor edits made for clarity

MPJ placement weights

Penalize non-mutual judges	40
Penalize non-mutual panels (elims only)	40
Penalize less preferred judges	20
Avoid burning commitments early	5
Prefer hard-to-place judges	10
Promote use of diverse judging	5
Prefer hired judges over obligated	10

Figure 1: Tournament organizers are offered these settings for MPJ when hosting their tournament through tabroom.com. It is unclear the effect these weights have on judge placement scores.

a tournament, and they are committed to judge a different number of rounds, the quantile preference of your first ranked judge is not $\frac{1}{200}$. Presume a team’s first ranked judge is only available for 1 round and all other judges are available for 5. Then, the scaled preference of that first ranked judge would be $\frac{1}{996}$, the second judge would have scaled preference $\frac{5}{996}$, and so on. This stops teams from gaming the system by inflating their top ranks with low-availability judges.

The other additions are more straightforward, but their actual effect on judge assignments are less well-defined. First, adding a bonus to “hard-to-place” judges helps ensure that tournaments will use their available judging appropriately. Similarly, including a bonus for hired judges over obligated⁵ judges eases the burden on coaches traveling with their teams. Last, adding bonus for “diversity-enhancing” judges is an attempt to counteract potential *de-facto* racial discrimination in judge preferences.

2.4 The Structure of Team Preferences

Due to MPJ, teams are afforded the opportunity to optimize their preference sheets. To figure out the most beneficial judges for them, teams will consider the ideological leanings of each available judge. This is usually discussed in similar terms as debate argumentation.

⁵(“Obligated judges” are brought by a university competing at the tournament. In order to ensure there is enough judging, schools are required to provide some amount of judging based on how many teams they bring to a tournament. “Hired judges” are hired by the tournament directly in order to ensure there is enough judging for each round.

Judges who are perceived as biased towards the policy argumentation are referred to as “policy judges,” while judges perceived as biased towards the kritik are referred to as “k judges.” The middle ground, judges who are not perceived as having strong ideological biases are called “clash” judges.⁶

Because teams want to win above all, it is desirable to them to have judges they think are biased in their favor. Thus, nearly every competitively-minded team ranks judges according to their perceived ideological similarity with that judge. Exceptions are generally limited to a handful of revered clash judges who occasionally find themselves highly preferred by teams far less ideologically flexible than themselves. This results in a general block structure to preferences for teams. Policy teams will place all policy judges in the top ranks, clash judges in the middle, and kritikal judges at the bottom, while kritikal teams will do the opposite.

However, bias is not the only reason that ideological alignment is seen as desirable. Interest, expertise, and enjoyment all motivate choosing ideologically aligned judges. For example, one of the greatest educational moments in debate comes shortly after the actual debating has concluded. After each debate, the judge gives an explanation of their decision to both teams and allows them to ask questions related to the round. If a team likes to make a particular argument, this heavily encourages them to highly rank judges with expertise in that argument in order to use their feedback to perfect it.

Furthermore, it is often taken as self-evident that it is less work to convince judges with subject-expertise related to an argument of its validity. A judge with lots of experience in policy debate might be familiar with all the relevant acronyms to mobile ballistic missiles and the treaties signed in the Cold War pertaining to their use, while a judge whose experience lies in kritikal debate might not. On the other hand, a judge with plenty of kritikal debate experience will be well-familiarized with the differences between Frank Wilderson’s scholarship and that of Tiffany King, while a judge focused on policy debate probably won’t. If there is some amount of prior knowledge necessary for an argument to make sense, teams will probably have more success and more fun debating in front of judges who already have that knowledge. They won’t have to spend as much speech time explaining background information and instead can jump right into making arguments.

This “ideological block structure” still allows for a lot of expression and optimization. Take, for example, a policy team who strongly dislikes a particular policy judge. If they rank that judge below every other policy judge, but before each clash and kritikal judge, they will be able to avoid that disliked judge while maximizing the ideological closeness of their judge assignments writ large. This can be understood by reasoning out the possibilities:

1. If they are paired against an ideologically opposed kritikal team, the disliked policy

⁶This term comes from the backlash to kritikal debate in the 1990s. Some policy traditionalists used the racist metaphor of “clash of civilizations” to refer to what was happening in Intercollegiate Policy Debate at the time. Although that term has rightfully fallen out of use to describe debates between (mostly white) policy and (disproportionately non-white) kritikal teams, they are still referred to in community as “clash debates” by fans of kritikal debate and policy debate alike. Although I’m sure at one point this was done either ironically or euphemistically, it has since become the default term.

judge is near the top of the policy team's preference sheet and is near the bottom of the kritikal team's preference sheet. Then, the judge will be moderately preferred but not at all mutually preferred. As such, presuming both teams ranked by ideological alignment, there will exist an equally preferred clash judge with far higher mutuality.

2. If they are paired against an ideologically similar policy team, the disliked judge is at the bottom of the policy block in the first team's preference sheet. As such, presuming both teams ranked by ideological alignment up to type of judge, there must exist a policy judge with higher preferredness and mutuality.
3. Flex teams are so rare as to not figure into a team's decision making process. Even so, if they are paired against a flex team, they will probably end up with a clash judge. As clash judges tend to make up the largest share of the judge pool, and flex teams will rank clash judges highest, it is highly probable that there is a more preferred and more mutual clash judge to assign.

The uniform acceptance of the ideological block approach to preferences allows teams to game their preference sheet. Say there is a judge commonly accepted as a clash judge with more policy sympathies. If a kritikal team thinks this is a misperception, they can attempt to match their ranking of this judge to that of the policy teams. If they are right, and keep their preference sheet otherwise sorted ideologically, this will likely be the most preferred and most mutual judge when they are paired against policy teams.

Perceptions of judges are fairly consistent between teams and universities. If a debater doesn't know about a judge, they will ask their coaches or teammates. If they don't know, they will ask friends from another school or read the judge's "paradigm," a publicly posted essay about their views on debate. Possibly, check which teams the unknown judge has judged in the past. Based on the type of rounds they have judged previously, teams will infer where that judge lies ideologically.

Perceptions of judges tend to ossify. First, judges tend to be ideologically consistent throughout their careers. Judges tend to come from one of two demographics: graduate students who debated in college and university coaching staff. Graduate students don't tend to stick around long enough for their views to change or for changes to be noticed. Coaching staff are unlikely to substantially change their approaches to debate over their careers. They became coaching staff by finding what works for them and what they like.

Second, the actual structure of preference sheets causes ossification. New teams will inherit perceptions from older teams and coaching staff. There are also far more judges than any one debater will ever be judged by, especially because of MPJ. For example, someone labeled a kritikal judge by all the policy teams will almost certainly never judge a policy team. They will never have the opportunity to be proven wrong.

Finally, tabroom.com, the website used to organize debate tournaments since 2012, includes a handy button for debaters to import their rankings of judges from previous tournaments.

This means that teams don't ever have to re-rank judges from scratch. Instead, they often simply slot unfamiliar judges into their previous rankings.

The consequence of all of this is that given

1. tournaments use MPJ for judge allocation
2. records of pairings for each round exist (i.e., affirmative team, negative team, judge, and available judges)
3. team preferences are sorted by ideological proximity

it should be possible to learn the ideological position of teams and judges.

3 Data Collection

Unfortunately, there is no easily accessible dataset containing the tournament information necessary to answer my research question. Before 2012, to the best of my knowledge, there is no accessible information on tournament pairings other than the records of National Debate Tournament, which only include elimination rounds and rarely list judges. Other resources, such as debateresults.com, are long defunct.

Since 2012, however, there has been a central repository containing almost all records of intercollegiate policy debate: tabroom.com. In the early 2010s, Chris Palmer, who worked in IT and as a debate coach, and Jon Bruschke, professor of human communication studies at Cal State Fullerton, received a grant from the Open Society Foundations to build a web-based platform to streamline the arduous processes of organizing and participating in debate tournaments [11]. This grant was not unconditional: Palmer and Bruschke were required to open-source the project and make data accessible [29]. Consistent with that ethos, the website launched with an API to allow convenient access to information.

However, after the Open Society Foundations stopped funding the project in 2014, and began to be funded by the National Speech and Debate Association instead, the requirements of the project changed. Although the original code for Tabroom is still publicly available, further updates were not required to be open-sourced and the API features were deprecated slowly over time until their complete elimination. Data now must be scraped from tabroom.com instead of accessed directly through API.

This creates a lot of problems for anyone trying to use all of the debate data that only exists on tabroom.com. The tabroom.com back-end has near-perfect records of debate data, with complete histories and unique IDs for each debater. However, users of the website have very little access to that format. This data can seemingly only be matched to actual round results through imperfect other imperfect simulacra of team identities. As far as I know, nobody has found a particularly effective method for extracting this data. This is magnified by the frequent minor redesigns that change hardly anything for the end-user, but break any existing scrapers.

Luckily, it is still possible to access these unique IDs for teams and judges. First, the homepage of [tabroom](http://tabroom.com) allows filtering by competitive season and circuit. This allows the generation of a list of all URLs corresponding to Intercollegiate Policy Debate tournament listings. These URLs contain unique tournament IDs which allow for precise distinction between tournaments of the same name, which would otherwise be a big problem for yearly tournaments.

Second, each tournament page contains a “judges” tab which displays the pool of eligible judges for each tournament. Importantly, on this page and this page only, there is a hyperlink directly to the permanent version of each judge’s [tabroom](http://tabroom.com) page. This is the only way to access the records of judges who have been inactive since January 2024, when inactive judges were unlisted from Tabroom’s search. This hyperlink also is the only place where a judge’s

unique ID is accessible. This ensures that there are no collisions in the data between judges with the same name and prevents avoid data errors otherwise caused when a judge changes their name.

Third, information from every round a judge has ever adjudicated is listed on their judge page. The information includes tournament name and hyperlink, date, division, round number, aff team, neg team, and the judge's decision. From this table, a hyperlink is accessible that contains the unique IDs for both debaters comprising each team. This is crux of how I construct the dataset. Without these unique IDs, there is almost no consistency in how teams are listed on Tabroom. Each tournament organizer uses their own standard on the tournament pages they run. They are also filled with strange errors, such as team names replaced with numbers or the name of an individual debater.

The data scraping process is as follows:

1. Using Selenium, interact with the JavaScript elements on the tabroom.com home page to find and store the URLs for each intercollegiate policy debate tournament
2. For each tournament, using BeautifulSoup, scrape each tournament judge list page to find and store permanent judge page URLs
3. For each stored judge, for each intercollegiate debate round in their record, record tournament name and ID, date, affirmative team name and IDs, negative team name and IDs, judge name and ID, date, and division.

This provides an exact record of each Intercollegiate Policy Debate round with relatively few data errors. The unaltered, complete dataset consists of records of 113302 ballots. Although there are unique identifiers for each team, there is still messiness that needs to be filtered manually.

First, it is necessary to ensure that all tournaments in the dataset actually included Intercollegiate Policy Debate rounds. Due to human error when constructing tournament pages, three tournaments are improperly listed as being part of the Intercollegiate Policy Debate circuit. Two are practice tournaments, and one is simply mislabeled. Eliminating them leaves 113302 total ballots.

Occasionally, Intercollegiate Policy Debate tournaments also run events that aren't policy debate. Because each round has a listed event, it is easy to filter ballots from improper events from the data. This leaves 111400 ballots.

On rare occasions, some teams manage to have no discernible ID. Outside of some sort of bug with the Tabroom back-end, this has only happened in two situations. The first is when both members of a team request the deletion their tabroom.com accounts and data. This purges both their team name and debater IDs from round records. The second is when a tournament has its results listed as private. As far as I can tell, this has only been the case for one small regional tournament in 2017. This is probably an accident, as the results

tab for each round is still public. After filtering for missing identifiers, there are 111292 acceptable ballots from debates between 8690 teams over nearly 11 years.

Finally, it is necessary to include an estimate of the list of available judges for each round. This is information that is unfortunately inaccessible. Each tournament page does list each judge and the number of rounds they are eligible to judge for, but these lists rarely correspond to reality. Commitment levels frequently change. Judges drop from the tournament due to illness, or are willing to judge more rounds to earn extra cash. Empirically, $\approx 2\%$ of rounds are not judged by individuals not even listed on the judge page before the tournament.

Due to these difficulties, I simply consider the pool of judges for a given round to be all judges who judge a round or any future round. I consider a judge's eligibility to be the number of future rounds they judge at a given tournament. For example, if judge A is placed in rounds two and three in an imaginary three round tournament, they would be considered in the pool for all rounds. As they judged two rounds, I would consider them to have two rounds of total eligibility. This allows us to reconstruct an estimate of the scaling factor used in scoring judge assignments. Note that this does not work for elimination rounds, where eligibility is decided differently.

4 Learning Team and Judge Ideology

4.1 Potential Approaches

The simplest approaches to estimate team and judge ideologies would involve clustering or topic modeling techniques such as probabilistic latent semantic analysis or latent Dirichlet allocation [18] [2]. A slightly more complex, but seemingly valid, approach might be to build a graph of team and judge interactions and generate representations of the nodes which can then be clustered. For example, one might cluster the Graph Laplacian or a random-walk representation such as Node2Vec or DeepWalk [35] [14] [31]. However, the unique structure of the data proves any of these approaches infeasible.

First, the data is structured into regional pockets, especially pre-COVID. This regional structure results in clusters and topics that represent geographic boundaries as opposed to ideological ones. For the vast majority of the years for which I have data, a large number of teams and judges would rarely travel outside of their regions. Teams in the West and Northeast, in particular, are often isolated due to cost of travel.

Second, these approaches would necessarily ignore Mutual Preference Judging. Even if it were properly able to distinguish ideological structure from regional structure, it would likely result in faulty classifications. For example, a policy team on the West Coast, where kritikal debate is much more common, would rarely be judged by a policy judge. They would both often be competing against kritikal teams, whose preferences would eliminate the possibility of their debates having a policy judge, and would have fewer available policy judges for their debate rounds.

The implication of this is that any successful model must incorporate both preference and judge pool, or else it would fail to pick up on the relevant structure and, even if it did, would be often wrong. As a result, I need a parametric model (excluding irrelevant structure) that can simulate the MPJ process from the moment when the affirmative and negative teams are chosen.

The goal of the model is to estimate continuous ideal points representing the argumentative proclivities of teams and judges. First, I believe that this conceptualization of the problem is natural given the MPJ system. Because ranks are scaled, competitors are required to order judges along a spectrum from 0 to 1. This makes the $(0, 1)$ interval a familiar setting for teams and debaters. If these results are to be used in practice, it is necessary to present findings in a familiar way, justifying the choice to estimate continuous ideal points in the $(0, 1)$ interval.

4.2 Variational Inference

To achieve this goal, I opt for a black box variational inference approach [32]. In this setting, the approximation of a conditional distribution is recast as an optimization problem. Where x are the observed variables and z are the unobserved latent variables, I define a generative process, the “model” $p_\theta(x|z)$ with parameters θ . From this structure, the aim is to approximate the potentially intractable “true posterior” $p_\theta(z|x)$ by forming the tractable “variational distribution” $q_\lambda(z|x)$ parameterized by λ .

Intuitively, fitting a model involves maximizing the log-likelihood of the observed data $\mathcal{L}(p_\theta(x))$. However, this becomes difficult when the relevant information is z , only x is observed, and $p_\theta(z|x)$ is difficult to compute. However, by carving up the log-likelihood it is possible to derive the Evidence Lower Bound (ELBO):

$$\begin{aligned}\mathcal{L}(p_\theta(x)) &= \log \int p_\theta(x|z)p_\theta(z)dz \\ &= \log E_{z \sim q_\lambda(z|x)} \left[\frac{p_\theta(x|z)p_\theta(z)}{q_\lambda(z|x)} \right] \\ &\geq E_{z \sim q_\lambda(z|x)} [\log p_\theta(x|z) + \log p_\theta(z) - \log q_\lambda(z|x)] \\ \text{ELBO}(\theta, \lambda) &= E_{z \sim q_\lambda(z|x)} [\log p_\theta(x|z)] - D_{KL}(q_\lambda(z|x) || p_\theta(z))\end{aligned}$$

The upshot is that now everything is computable: model $p(x|z)$, variational distribution $q(z|x)$, and prior $p(z)$ are all chosen for their tractability. And, because this is a lower bound on the log-likelihood of the data, maximizing it should lead to a better model fit. Intuitively, the interpretation is fairly straightforward. The first component is the conditional likelihood and the second component is the regularization term, pulling the variational distribution towards the prior.

The goal then to minimize the negative ELBO with respect to the model and variational parameters. With complex models, it is beneficial to use some type of gradient-based optimization scheme in order to speed things up. However, the gradient of the ELBO with respect to the parameters is problematic in its naive form:

$$\begin{aligned}\nabla_\theta \text{ELBO} &= E_{z \sim q_\lambda(z|x)} [\nabla_\theta \log p_\theta(x, z)] \\ \nabla_\lambda \text{ELBO} &= E_{z \sim q_\lambda(z|x)} [(\log p_\theta(x, z) - q_\lambda(z|x)) \times \nabla_\lambda \log(q_\lambda(z|x))]\end{aligned}$$

The first problem is that, although the gradient with respect to θ is relatively well-behaved, the gradient with respect to λ is scaled by some potentially large difference term. This can create very noisy updates which cause huge difficulty in finding local minima. Although “correct,” it is often too noisy to be useful, particularly in more complicated models. Second, it can also be computationally costly as it requires sampling through the entire distribution.

However, with most common parametric distributions, it is possible to re-parameterize such that the effect of parameters are entirely deterministic [21]:

$$z \sim q_\lambda(z|x) \iff z = g_\lambda(\epsilon, x)$$

for some vector function $g(\cdot)$, some independent marginal $p(\epsilon)$. It is easy to reformulate in this way if the variational distribution has a tractable CDF or comes from a location-scale family. Then, $q_\lambda(z|x)dz = p(\epsilon)d\epsilon$. This equivalence can be used to assemble an unbiased estimator for the ELBO differentiable with respect to the parameters.

$$\begin{aligned} E_{z \sim q_\lambda(z|x)}[\log p_\theta(x, z) - \log q_\lambda(z|x)] &= E_{\epsilon \sim p(\epsilon)}[\log p_\theta(x, g_\lambda(\epsilon, x)) - \log q_\lambda(g_\lambda(\epsilon, x)|x)] \\ &\approx \frac{1}{L} \sum_{i=1}^L \log p_\theta(x, g_\lambda(\epsilon_i, x)) - \log q_\lambda(g_\lambda(\epsilon_i, x)|x) \end{aligned}$$

for some arbitrary batch size L .

When the KL-Divergence between the prior and variational distribution is analytically tractable, there exists an even better estimator with lower variance

$$ELBO \approx -D_{KL}(q_\lambda(z|x)||p_\theta(z)) + \frac{1}{L} \sum_{i=1}^L \log p_\theta(x|g_\lambda(\epsilon_i, x))$$

This makes gradient optimization methods much more feasible. The gradient over λ can pass through an expectation over ϵ , making more of this particular estimate deterministic and less of it sample-dependent. Unfortunately, the remaining random elements mean updates must still be approximated through Monte Carlo sampling. To ensure the model converges quickly, it's beneficial to pursue other variance reduction methods as well.

Rao-Blackwellization is the reduction of variance of estimates by replacing expectations with conditional expectations [19]. Utilizing the conditional structure of this model, it is possible to reduce the variance of each parameter gradient estimate by only considering operations in the factor graph in the Markov blanket of that parameter. Explicitly, it is not necessary to sample random variables that do not depend on z_i in order to get an unbiased estimate z_i 's contribution to the loss. Rather than introducing more variance by always sampling top-down and including many variables, variance can be reduced by conditioning each variable on only "local" signals.

Lastly, variance can be managed with a "control variates" approach. Subtracting a constant c from the ELBO will not affect the its gradient but will reduce the variance of its gradient. Shown with the naive ELBO:

$$\begin{aligned} \nabla_\lambda ELBO &= E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x) - c) \times \nabla_\lambda \log(q_\lambda(z|x))] \\ &= E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x)) \times \nabla_\lambda \log(q_\lambda(z|x))] - c E_{z \sim q_\lambda(z|x)}[\nabla_\lambda \log(q_\lambda(z|x))] \\ &= E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x)) \times \nabla_\lambda \log(q_\lambda(z|x))] - c E_{z \sim q_\lambda(z|x)}[\nabla_\lambda \frac{(q_\lambda(z|x))}{(q_\lambda(z|x))}] \\ &= E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x)) \times \nabla_\lambda \log(q_\lambda(z|x))] \end{aligned}$$

But, trivially,

$$E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x) - c)^2] \leq E_{z \sim q_\lambda(z|x)}[(\log p_\theta(x, z) - q_\lambda(z|x))^2]$$

As such, by learning parameter $c \approx \text{ELBO}$, it is possible to minimize the variance of the estimator. In fact, it's not very useful to just have a control variate for the full ELBO. In practice, control variates can be designed for each of the Rao-Blackwellized "local" ELBO estimates.

4.3 Model Design

The goal is to estimate continuous ideal points representing the argumentative proclivities of teams and judges. Let $\phi \in \mathbb{R}^T$ denote the vector of T team ideal points and $\theta \in \mathbb{R}^J$ denote the vector of J judge ideal points, the variational distribution $q(\phi, \theta|x)$ and the model $p(x|\phi, \theta)$. In this section, I will first describe the choice of model, then describe the choice of variational distribution and prior.

From earlier discussion of MPJ, the model can be defined in more detail. With known judge ideal points θ , team ideal points ϕ , and affirmative team a_i , negative team n_i , set of available judges \mathcal{P}_i for $i \in \{0, 1, \dots, N-1, N\}$ the basic generative process is defined:

1. calculate preference penalty matrix $\Omega_{t,j} = g(\phi_t, \theta_j)$ for teams $t \in \{0, 1, \dots, T-1, T\}$ and judges $j \in \{0, 1, \dots, J-1, J\}$
2. for a given round i :
 - (a) observe affirmative team a_i , negative team n_i , set of available judges \mathcal{P}_i
 - (b) calculate quantile rankings:

$$\omega_{a_i} = \text{rank}(\Omega_{a_i, \mathcal{P}_i}) / |\mathcal{P}_i| \in \mathbb{R}^J$$

$$\omega_{n_i} = \text{rank}(\Omega_{n_i, \mathcal{P}_i}) / |\mathcal{P}_i| \in \mathbb{R}^J$$

- (c) draw judge $j_i \sim \text{Categorical}(f(\omega_{a_i}, \omega_{n_i}))$

In order to accurately match the MPJ process, this outline for the model is relatively set in stone. However, there remain several design choices still left open. Many of the minor components of the actual MPJ process are not features I have access to. I generally do not have a way to learn the identities, hire status, or difficulty to place of judges. These are also criteria that are not even incorporated into the MPJ process at many tournaments. As such, it makes more sense to exclude them from the model. I also do not include the scaled quantile rankings in my model. They are computationally costly to compute and did not seem to improve the quality of recovered latent information.

The first component of the model is the matrix of preference penalties, defined $\Omega_{t,j} = g(\phi_t, \theta_j)$. For each team, this assigns a penalty to each judge given their relative ideological scores. That penalty determines where each team will rank judges on their preference sheets for tournaments.

The choice of the function g is surprisingly thorny. The most intuitive approach would be to use the Euclidean distances between ideal points as the preference penalties,

$$\Omega_{t,j} = g(\phi_t, \theta_j) = \text{abs}(\phi_t - \theta_j)$$

This conveniently keeps everything in the same scale as before. In practice, however, this is ineffective. For some reason, “smoother” distance metrics tend to perform substantially better. Perhaps this is because these distances push small distances nearer together, increasing robustness to minor perturbations. Rounding things out by squaring the distance, as below, is effective when fitting the model

$$\Omega_{t,j} = g(\phi_t, \theta_j) = \text{abs}(\phi_t - \theta_j)^2$$

The structure of the next component is not much of a choice at all. The quantile rankings of judges are the workhorse of MPJ. Judge assignments are almost entirely decided by those rankings, as discussed in section 2.3. Given that these rankings are so important in the real world, they should be included in our model’s generative process. With the data including the set available judges \mathcal{P}_i , the affirmative and negative teams a_i, n_i , and the preference penalty matrix Ω , it is possible to evaluate the relevant rankings for each debate.

The last design choice worth discussing is how the judges are drawn. In order to make the model computationally tractable, it is necessary to simplify the MPJ process. In reality, because one person cannot judge two debates at once, simultaneous judge assignments are conditionally dependent. However, the structure of variable batch sizes with conditional dependency (as tournaments range from three concurrent debates to over one-hundred) is incredibly inconvenient. GPUs do not appreciate inconsistency in size and sampling with replacement begs the question of in which order to sample. Trying to map the MPJ process one-to-one, where judges assignments are greedily according to a penalty then permuted until the tournament organizers decide enough time has passed, is simply not feasible when performing inference.

I opt for a relatively simple approach. Defining “mutuality penalty” m_i and “preference penalty” p_i for judge assignments

$$\begin{aligned} m_i &= \text{abs}(\omega_{a_i} - \omega_{n_i}) \in \mathbb{R}^J \\ p_i &= \omega_{a_i} + \omega_{n_i} \in \mathbb{R}^J \end{aligned}$$

Then, it is possible to proceed with Bayesian multinomial regression using m_i and p_i as the features, fitting parameters c_1, c_2 .

$$j_i \sim \text{Categorical}(\text{softmax}(c_1 m_i + c_2 p_i))$$

This setup should place high probability on judge assignments with low MPJ penalty and low probability on judge assignments with high MPJ penalty while avoiding costly model complexity. Increasing the size of the parameter space by exponentiation of the penalties is a consideration, but practically does not result in a substantially better fit.

In designing the prior and variational distribution, there is still a bit of wiggle room. Ultimately, I choose to bound ideal points in the interval $(0, 1)$ in order to keep everything consistent in scale. Then, it is natural to draw ideal points from the beta distribution with hyperparameters λ :

$$\begin{aligned}\phi_t &\sim \text{Beta}(\lambda_1^{(t)}, \lambda_2^{(t)}) \\ \theta_j &\sim \text{Beta}(\lambda_1^{(j)}, \lambda_2^{(j)})\end{aligned}$$

for team $t \in [0, 1, \dots, T - 1, T]$, judge $j \in [0, 1, \dots, J - 1, J]$.

I believe that the distribution of these ideal points should be trimodal for teams and judges, as discussed in section 2. However, I have no solid understanding of where the modes should be or how much density to place under them. As such, I choose a uniform prior over $(0, 1)$ for ideal points.

4.4 Implementing the Model

To implement the model, I use the Pyro probabilistic programming language in Python [1]. Built on top of PyTorch, it allows for relatively quick and easy variational inference. Pyro, as opposed to base PyTorch, makes the previously mentioned variational inference techniques very convenient. For re-parameterizable distributions, it will automatically re-parameterize when sampling. Similarly, it will also automatically Rao-Blackwellize, computing the ELBO locally when possible. It also makes it very easy to implement control variates of common form. I opt for a simple approach, using decaying averages of local ELBO estimates as baselines.

However, there are still problems with the model as described. Most glaringly, the rank operation is not differentiable. In principle this makes it impossible to learn the beta concentration parameters λ in the variational distribution. Luckily, it is possible to create an arbitrarily accurate differentiable approximation of sorting and ranking.

By recasting the argsort and ranking operators into optimization problems, it is possible to also cast the sorting operator $s(\cdot)$ as a linear program over the permutahedron induced by a vector v , and the ranking operator $r(\cdot)$ as a linear program over the permutahedron induced by the reversing permutation $\rho = (n, n - 1, \dots, 2, 1)$ [3].

$$\begin{aligned}s(v) &= \arg \max_{y \in \mathcal{P}(v)} \langle y, \rho \rangle \\ r(v) &= \arg \max_{y \in \mathcal{P}(\rho)} \langle y, -v \rangle\end{aligned}$$

where $\mathcal{P}(v)$ is the permutahedron induced by a vector v such, or equivalently the convex hull of permutations of vector v . However, this formulation does not result in differentiability. The sorting operator is piece-wise linear, and thus differentiable almost everywhere. The

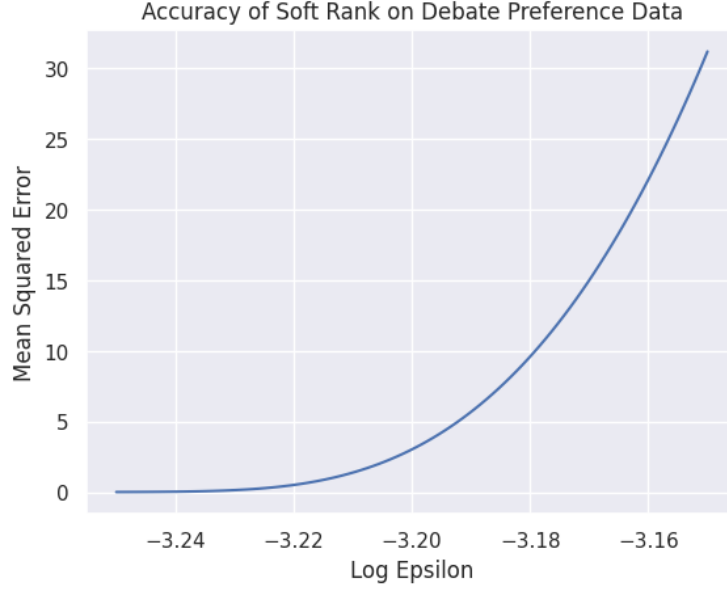


Figure 2: Mean squared error of soft rank with different values of regularization parameter ϵ . Performed on simulated preferences of the same dimension and scale as in the real data.

ranking operator, on the other hand, is discontinuous and piece-wise constant, making its derivatives either 0 or undefined.

To ensure differentiability, all that needs to be added is a regularization term. There are many options, but quadratic regularization serves well enough. By introducing a quadratic regularization term and regularization strength parameter ϵ , the “soft rank” operator is defined:

$$r_\epsilon(v) = \arg \max_{y \in \mathcal{P}(\rho)} \langle -v/\epsilon, y \rangle - \frac{1}{2} \|y\|^2 = \arg \min_{y \in \mathcal{P}(\rho)} \|v/\epsilon + y\|^2$$

Unlike the “hard” rank operator, the soft rank operator $r_\epsilon(v)$ is piece-wise linear - it’s differentiable. It is also guaranteed to converge to the real ranking operator as $\epsilon \rightarrow 0$. In fact, it can be proven there exists a threshold $b > 0$ for each ranking task such that if $\epsilon < b$, $r_\epsilon(v) = r(v)$. However, this causes the gradient to vanish. The best choice of epsilon is small enough that the soft rank approximation is fairly accurate but large enough to ensure a strong signal through the gradient. With the data dimensions I am working with, $\epsilon = 10^{-3.19}$ is my (relatively arbitrary) value of choice.

Fortunately, this optimization problem has an analytic solution. It can be further recast into an isotonic optimization problem and solved with the pool adjacent violators algorithm [24]. This results in an analytic solution with a block structure, making it possible to solve the soft ranking and compute its Jacobian in $O(n \log n)$ time.

After utilizing soft ranking in the model in Pyro, there are still a couple problems that need sorting out. In practice, it is difficult for multinomial regression coefficients c_1 and c_2 to avoid getting stuck at the local optima found when $c_1 = c_2 = 0$. Although a fairly poor local

optima, representing completely at random assignment of judges, it is a smooth optima that also results in very little gradient information for the parameters. This means that, once there, parameters cannot learn their way out.

To remedy this, I constrain $c_1, c_2 \in (-\infty, -0.5)$ and fix c_1 and c_2 to a reasonable initialization until a sufficiently large number of SVI steps have passed. The latter reduces the pull of the faulty optima, as unfit concentration parameters make judge assignment completely at random seem a good option. The former provides a safeguard on the off-chance that the model moves towards completely-at-random assignment anyways. Lastly, I set the learning rate for these regression coefficient parameters an order of magnitude lower than the rest. This helps prevent them from sprinting to zero before the signal from other parameters can pull them away.

Due to the use of the differentiable ranking method, it is unfortunately not feasible to have a large number of Monte Carlo samples when estimating the gradient updates for the variational parameters. The differentiable rank is difficult to perform on tensors with dimension greater than two while benefiting from the structure of GPU architecture. Massaging tensors into shape results in substantial slowdown, so unfortunately the variance of the gradient is likely rather high. This means it is necessary to use a stochastic optimization technique that will push it through the noise effectively.

In practice, stochastic gradient descent simply can't cut through the noise and slowly drifts towards the completely at random minima. ADAM [20] and RMSProp occasionally find solid minima, but are very sensitive to initialization and even order of sub-sampling. ADAM works best, but they all often get stuck early seemingly for any learning rates and momentum parameters. However, using ADAM with Nesterov's accelerated gradient for momentum works well.

The base ADAM algorithm is the combination of RMSProp and and SGD with weighted average momentum. With g_t the vector of gradients with respect to parameters, learning rate λ , model parameters θ , and mean decay parameters β , the ADAM update is defined:

$$\begin{aligned} m_t &= (\beta_1 m_{t-1} + (1 - \beta_1) g_t) \\ \hat{m}_t &= m_t / (1 - \beta_1^t) \\ n_t &= (\beta_2 n_{t-1} + (1 - \beta_2) g_t^2) \\ \theta_t &= \theta_{t-1} - \lambda \frac{m_t}{\sqrt{n_t / (1 - \beta_2^t)} + \epsilon} \end{aligned}$$

The RMSProp component n_t scales the gradient update by the inverse of the decaying mean of the norm of the gradients. This slows down learning for features that recently changed quickly and accelerates learning for features that are stagnant. The momentum component in ADAM m_t substitutes the gradient in the update equation with the decaying mean of the gradients. This accelerates learning in the directions where signal remains consistent and slows learning in directions where signal is highly variable.

However, there is no reason that this formulation of momentum must be the best approach. Note that m_{t-1} does not depend on the current gradient g_t . This means that it is possible to “look ahead” in momentum. This is the core idea behind Nesterov’s accelerated gradient [8]. Furthermore, the NAG involves the scheduling of β_1 , decaying it over time. Then, the new momentum update term \bar{m}_t , used in place of \hat{m}_t , is defined

$$\bar{m}_t = \beta_1^{(t)} \frac{m_t}{1 - \prod_{i=1}^{t+1} \beta_1^{(i)}} + (1 - \beta_1^{(t)}) \frac{g_t}{1 - \prod_{i=1}^t \beta_1^{(i)}}$$

In the deterministic, smooth convex setting, it has been established that Nesterov momentum will substantially outperform other methods early in training [37]. Although the relevant optimization problem is not necessarily convex, I find in practice that this approach to momentum results in a learning process more robust to initialization and more easily able to avoid the pitfalls of poor local optima.

4.5 Experimentation and Results

I started by constructing a synthetic dataset with the same dimensions as the real data. First, I draw team and judge ideologies. I draw from a mixture of three beta distributions, with modes at 0.85, 0.5, and 0.15 respectively. Then, I simulate tournament attendance, randomly assigning teams and judges to tournaments of the same sizes as observed in the data. After judges have their number of eligible rounds determined at random, I calculate preference rankings for each team for each tournament. This is done by hard ranking $|(\phi - \theta + \epsilon)|$, with some random noise ϵ . I make the choice to include a stochastic component to better reflect the minor adjustments made to preferences over time and the potential for teams to misrank judges.

Next, I optimize judge assignments round by round according to the MPJ penalty and subject to judge availability. The objective function is calculated $\sum_{i=1}^{\text{nround}} 40m_i + 20p_i$, where p_i is preference penalty and m_i is mutuality penalty. In a random order, I assign judges greedily to rounds to minimize this penalty then use simulated annealing to look for a better minimum.

I feed a subset of 20000 rounds to the model. I use a batch size of 6000 for each stochastic variational inference update and warm up the model by fixing $c_1 = -4, c_2 = -3.7$ for the first 6000 epochs. For the first 10000 epochs, I use a learning rate of 0.02 and then begin tapering it as needed when learning slows, until converging to an optima.

After the model is fit, I then convert the estimated distributions over judge and team ideal points into three categories. To do this, I simply measure where these distributions place the bulk of their density over the (0,1) interval. I compare these values to the true distribution each team and judge were drawn from. In practice, this works quite well. On synthetic data, I achieve a classification accuracy of 0.90 on judges and 0.95 on teams. Furthermore, the expected value of the learned latent ideal point distributions serve as reasonable estimates

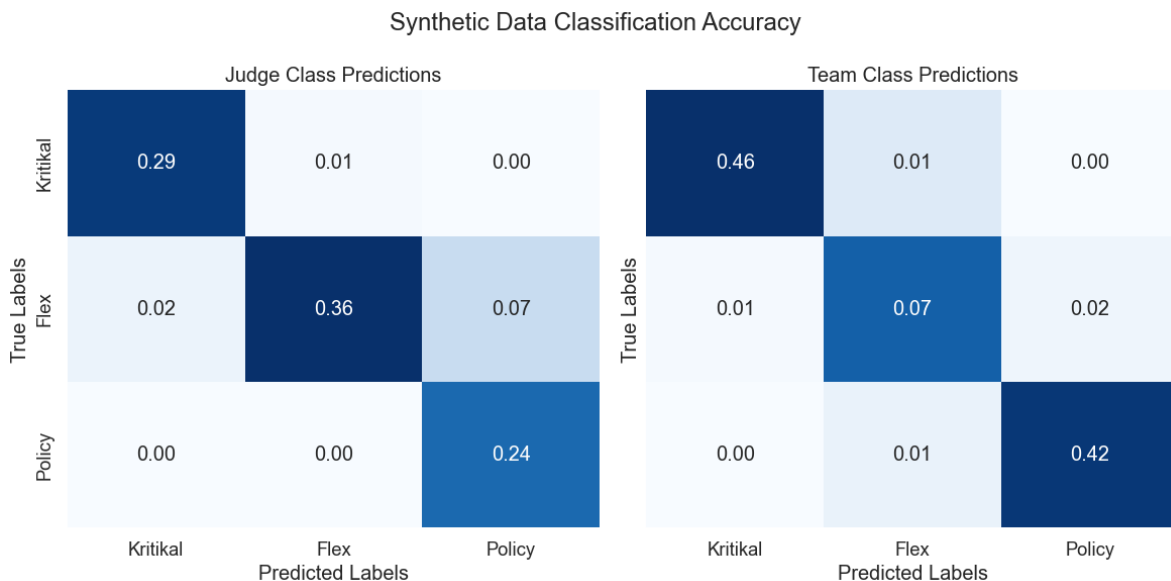


Figure 3: Confusion matrices for synthetic data classification. Annotations represent the proportion of teams or judges in each cell. I achieve a 90% prediction accuracy on judges and a 95% prediction accuracy on teams in synthetic data.

of true ideal points. The mean average error when predicting exact judge ideal points is 0.05 and for teams is 0.067.

Interestingly, this model is worse at classifying judges but better and estimating their real ideal points. In practice, it tends to send the ideal points of teams with a large number of observed pairings towards the margins. On the other hand, judges with a large number of observed pairings tend to have the mean of their estimated ideal point distribution converge to the ground truth value. It is unclear to me why this is the case and warrants further investigation.

When trying to fit the model on real data, I only use pairings from varsity rounds. This is necessary because in novice and junior varsity debate, teams are often too green or not competitively-minded enough to strategically fill out their preference sheets. At the varsity level, it is much safer to assume that preference sheets are organized according to model assumptions.

The impetus for designing this model is the infeasibility of manually categorizing teams and judges. However, by manually classifying a small subset, it is possible to evaluate the quality of the approximation. It is also necessary to use this subset as indicators to find the regions of the unit interval corresponding to different ideological categories, as there is no guarantee that teams and judges will be evenly ideologically distributed or that the perceived categories will align with an obvious division of the $(0, 1)$ interval. I pick about thirty “indicator” teams and judges whose ideological leanings I am familiar with, selected to ensure diversity of ideological position, demographic, and geographic location. As a member

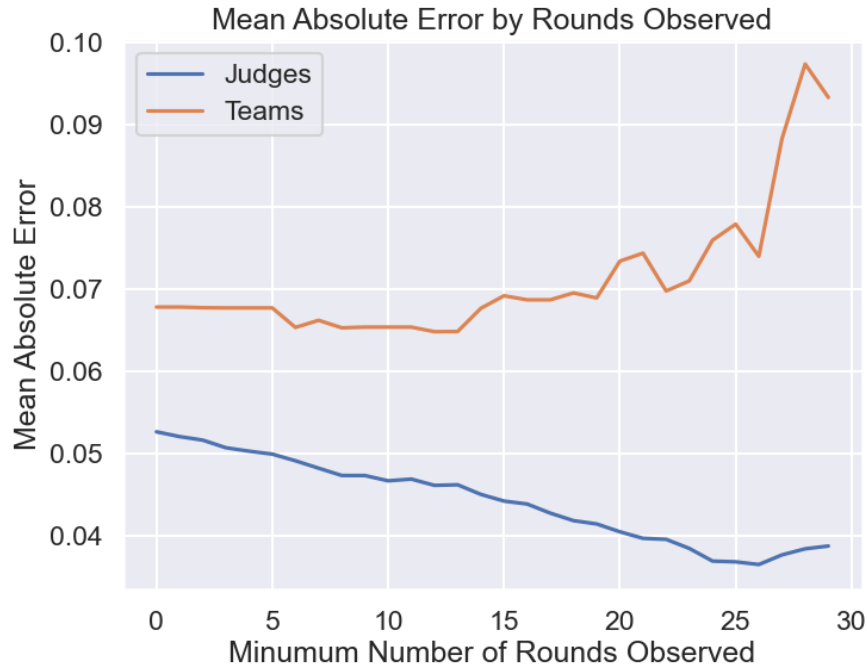


Figure 4: Strange behavior in model fit. Teams observed many times have worse estimates, while judges observed many times have better estimates.

of the University of Houston coaching staff, I can also compare the ideological rankings provided by the model to the actual preference sheets of Houston teams.

Fitting the real data is much more difficult than fitting toy data. Presumably, this is because the real data conforms worse to the key assumptions made about preferences. Although ideologically stratified, preferences have quite a bit of variability within categories. It’s easy to sort judges into categories but quite difficult to finely distinguish between hundreds ideologically similar judges. Teams’ opinions of the quality of judges or established relationships between teams and judges typically break perceived ties between rankings. When teams do not have an established relationship with a judge, they are often hesitant to rank them highly, preferring predictability in judging over marginal potential “improvement” in their rankings.

Including an additional parameter to account for “nudges” up or down the pref sheet for each team does not improve the fit. Adding additional noise to the model does not seem to help either, slowing learning without substantially changing the learned distributions.

The most notable failing is the very strange behavior in inferring team ideal point distributions. Based on the fits of indicator teams, a team’s ideal point distribution appears learned almost at random. It seemingly has nothing to do with either who judges a team or their “ground truth” ideal point, although is generally consistent between model fits. Only about 45% of teams are classified into the category that judges them most disproportionately. Were the model learning correctly, one would expect this to be near 100%. It should identify the

teams almost never be judged by individuals on one side of the ideological spectrum as belonging to the other side. In one egregious case, a team is classified as kritikal despite having been judged almost exclusively by policy judges and only a single time by a kritikal judge.

Luckily, despite a general failure to classify teams, the model does a good job of classifying judges. Looking at the classification of indicator judges, every single one of them is placed sensibly. Similarly, the ideal points align very well with the actual preferences of Houston teams across the ideological spectrum. This means that, although the model has explicitly failed in one aspect, it is still possible to use the successful part to impute the failed part.

By tabulating the proportion of rounds judged by judges of each category at each tournament a team participates in, it is possible to establish baseline categorical weights b_t over judge categories. Were the team to write their preference sheets at random, one would expect their judging to be drawn Multinomial(n_t, b_t) where n_t denotes the number of debate rounds they have participated in. If the observed distribution of a team’s judging is substantially skewed away from the baseline distribution towards a particular category, they can be classified as such. Although this approach is not perfect, it leads to correct classifications of all thirty indicator teams. Now, with categorizations for teams and judges, it is possible to evaluate the effect of team and judge types on round outcomes.

5 Estimating Effect on Round Outcome

Before discussing the effect of round types, it is necessary to establish how they are determined. In my analysis, there are two ways I will determine them. The first is the team-focused round type, which classifies rounds based on only the teams classes. Because debate is asymmetric, the three categories result in nine team-focused round types. However, I choose to not to explore the effects of round types defined by flex teams. There are simply too few of them to gain any useful information. This results in four team-focus round types: policy v policy, policy v k, k v policy, and k v k. The second way is the comprehensive round type, which creates twelve distinct types by also taking the judge classes into account.

Given these classifications, one might be content with a basic exploration of the round types. For each debate type, a binomial test, treating affirmative wins as successes, would suggest that the null hypothesis of a fair game can be rejected with a 95% confidence threshold in policy v k and k v policy rounds with non-neutral judges. When judged by a policy judge, these rounds appear decided nearly 2 : 1 in favor of the policy team. Interestingly, these results are fairly symmetric across side assignments.

However, this approach doesn't help build an understanding of how these interactions have changed over time. Furthermore, these results are only meaningful under the belief that these debates were evenly matched. If the sample of policy teams is more skilled than the sample of kritikal teams, then it makes sense that they should generally win in the head-to-head with neutral judging. This also tells us little about what performs better at different skill levels. It's possible that one style crushes the other in low quality debates, but fails at high levels of competition. In order to definitively determine if a particular style confers a strategic advantage, there must be some notion of who *should* win a given debate.

5.1 Skill Estimation

In a perfectly predictable fair game, the more skilled player always wins. The simplest case is the two player game. Although Intercollegiate Policy Debate has two participants on each team, it is natural to treat teams as units. This makes debate a two-player game as well, which substantially reduces the complexity in modelling its outcomes..

Imagining some quantification of skill for players s_1 and s_2 , a simple paired comparison model $P(y = 1 \text{ wins} | s_1, s_2) = \delta(s_1 - s_2 > 0)$ can be constructed. However, games are generally not deterministic in practice. Even an unskilled player can occasionally play a perfect game. The Bradley-Terry paired comparison model incorporates random chance into the model, instead generating a Bernoulli probability for victory based on the skill values of the competitors [4]. Let y denote the game outcome, such that $y = 1$ denotes a win for player one. Then, the Bradley-Terry paired comparison model is defined:

$$P(y = 1|s_1, s_2) = \frac{e^{s_1}}{e^{s_2} + e^{s_1}} = \text{logit}^{-1}(s_1 - s_2)$$

From this formulation, it is possible to fit skill values for each competitor through maximum likelihood estimation. Despite no closed form solution for optimal s_i , it is relatively easy to fit iteratively and is in fact a convex optimization problem [28]. This model can be further generalized as follows:

$$P(y = 1|s_1, s_2) = f(s_1 - s_2)$$

It is not necessary that $f(\cdot)$ is the inverse logit function. As long as it some type of sigmoid function, such that $f(x) + f(-x) = 1$, the choice of $f(\cdot)$ changes the scale of the skill space but is otherwise equivalent [9].

Adding a bit more detail to the description of variability in game outcomes allows for the formulation of a more useful model. Then, it is necessary introduce the idea of performance variability - a modifier to skill that serves as a quantification of on-the-day performance. The winner of a game will then be determined by which player performs better. For simplicity, it is useful to treat performance as drawn from a normal distribution, $p_i \sim \text{Normal}(s_i, \beta^2)$, where β is an arbitrary scaling parameter. Then, the probability of a win for player 1 becomes:

$$\begin{aligned} P(y = 1|s_1, s_2) &= P(\text{Normal}(s_1, \beta^2) - \text{Normal}(s_2, \beta^2) > 0) \\ &= \Phi\left(\frac{s_1 - s_2}{\sqrt{2}\beta}\right) \end{aligned}$$

Because the normal distribution is symmetric, its CDF Φ is a sigmoid. As such, the above is still the same Bradley-Terry paired comparison model with formalized performance variability. As β denotes the standard deviation of the normal distribution which performance is drawn from, $s_1 - s_2 = \beta \iff P(y = 1 \text{ wins}) = 0.76$.

However, this model suffers from a major flaw - it treats skill as static. In reality, skill is something that varies over time. This is important to the setting because I am not interested in evaluating who had the most dominant career. Instead, I care about the odds of particular debate rounds.

The most notable attempt to remedy this is the Elo rating algorithm, still used by many competitive leagues like FIDE in chess. To allow variation over time, the Elo algorithm introduces an update rule that is shown to converge to a “true” skill after enough iterations [17].

$$s_1 = s_1 + y\alpha\beta\sqrt{\pi}\left(\frac{y+1}{2} - \Phi\left(\frac{s_1 - s_2}{\sqrt{2}\beta}\right)\right)$$

With where $y = 1$ for win, 0 for tie, -1 for loss and α dictates update rate, chosen as a function of rating difference. Necessarily, there is also a “base rating” s_0 .

This update rule allows a player’s skill to actually evolve over time, but is still unsatisfactory. Elo is very inaccurate when players only have a small number of games on record. To remedy this, early games are considered “provisional.” This provisional process involves introducing tons of exceptions to the normal update rules in order to encourage faster learning and to prevent too much damage to the ratings of established players. This is an unsatisfying solution, and doesn’t really resolve the core complaint.

Furthermore, Elo doesn’t account for change over time when players are inactive. There is little reason to expect a player who has been inactive for a while to be exactly the same skill level as they were in their last competition. FIDE attempts to keep the skill estimates accurate by requiring continuous activity. If a chess player goes longer than a year without competing in a rated event, they lose their rating and become provisional again. This is annoying, difficult, and inelegant. It fails to solve the actual problem while creating extra book-keeping chores.

Glicko attempts to resolve these two problems by approaching the problem from a more explicitly Bayesian perspective [13]. Glicko’s first innovation is that it treats a player’s skill as a random variable rather than a fixed value. It also introduces a new parameter τ which denotes the uncertainty in skill introduced by the passage of time between a player’s matches. Then, skill at time $t + k$ is determined as follows:

$$\begin{aligned} s_i^{(t)} &\sim N(\mu_i^{(t)}, \sigma_i^{(t)2}) \\ s_i^{(t+k)} | s_i^{(t)} &\sim N(s_i^{(t)}, \tau^2 k) \\ \text{with marginal } s_i^{(t+k)} &\sim N(\mu_i^{(t)}, \sigma_i^{(t)2} + \tau^2 k) \end{aligned}$$

This establishes an explicit dependence structure between skill levels at each timestep. Although its update formula is somewhat similar to that of Elo, it is too complicated to justify its space on the page. Still, it is designed to be simple enough for pen and paper and calculator in order to be usable at chess tournaments.

This algorithm is much better suitable for use, effectively solving two of the main problems present in Elo by quantifying uncertainty. However there are two problems remaining in Glicko. First is that it is designed for a live-update setting where the only relevant question is each player’s current skill. As such, it only propagates information forward through time. Due to its relation to the Kalman filter, this forward propagation of information will be termed “filtering” [9]. In this setting, I want to be able to determine the odds for each debate round in the past decade. Then, it is necessary for us to propagate information backwards, which will be termed smoothing, in order to better understand the evolution of skill over time. The second problem is that Glicko is sensitive to the order of updates. Without changing who beats who, by scrambling the order updates are processed, Glicko results in different estimates.

TrueSkill Though Time is designed to resolve these issues [5]. Developed to generate reliable estimates of the skill level of chess players at any point in history, it uses the same basic structure as Glicko. For clarity, the process that determines who wins a game at time $t + 1$

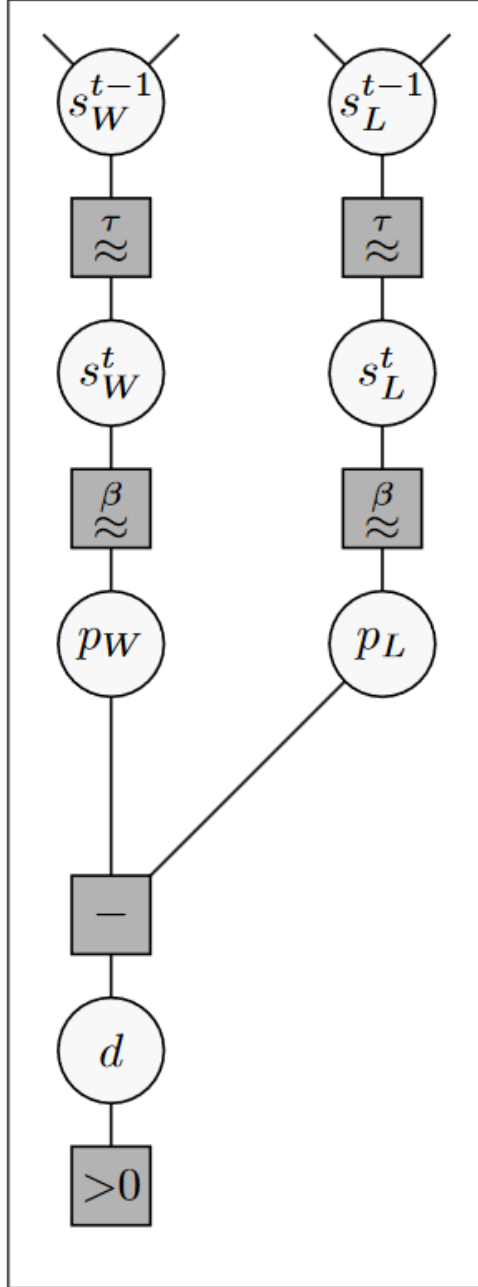


Figure 5: Factor graph of TrueSkill Through Time, taken from the original paper [5]. Starting from skill distribution at previous match at time $t - 1$, it incorporates time deviation τ to get skill distribution at time t , then incorporate performance deviation β to get match performance. The difference of match performances determines the winner.

is as follows:

1. draw player skills at time t : $s_i^{(t)} \sim N(\mu_i^{(t)}, \sigma_i^{(t)2}), s_j^{(t)} \sim N(\mu_j^{(t)}, \sigma_j^{(t)2})$
2. add noise from passage of time: $s_i^{(t+1)} = s_i^{(t)} + N(0, \tau^2), s_j^{(t+1)} = s_j^{(t)} + N(0, \tau^2)$
3. add performance modifier: $p_i = s_i^{(t+1)} + N(0, \beta^2), p_j = s_j^{(t+1)} + N(0, \beta^2)$
4. declare winner $y_{i,j}^{t+1} = \delta(p_i - p_j > 0)$

Because I don't have the same practical constraints as in Glicko, I don't care about ease of updating. As such, it is reasonable to update the model through message passing forward and backwards in time instead of just approximate forward filtering. By assembling the factor graph of this new model, it is apparent that this structure is very amenable to the message passing algorithms.

In the abstract, message passing can get quite messy. With $f(\cdot)$ a factor, v_i a random variable, and $n(x)$ the neighborhood of x on the factor graph, the sum-product algorithm gives us three equations that define message passing updates [22].

$$\text{marginal: } p(v_k) = \prod_{f \in n(v_k)} m_{f \rightarrow v_k}(v_k) \quad (1)$$

$$\text{factor to R.V.: } m_{f \rightarrow v_j}(v_j) = \int \dots \int f(n(f)) \prod_{i \neq j} m_{v_i \rightarrow f}(v_i) dv_{\setminus j} \quad (2)$$

$$\text{R.V. to factor: } m_{v_k \rightarrow f}(v_k) = \prod_{\bar{f} \in n(v) \setminus f} m_{\bar{f} \rightarrow v_k}(v_k) \quad (3)$$

Although these appear daunting, and can be with complicated factor graphs, their interpretation is relatively straightforward. The first equation says that the marginal probability at a variable is the product of messages to the variable. The second equation says that the message from a factor f to random variable v_j is the integral of the product of the factor evaluated at all other adjacent random variables and the marginals of those random variables. The third equation says the message from a random variable v_i to a factor f is the product of the messages of all other factors adjacent to v_i .

With the nice, tree-like structure of the factor graph for matches (Figure 5), messages are very simple. Every node has at most two neighbors, drastically reducing the complexity of the messages. When everything is Gaussian, it is easy to use the updated marginals from the message passing algorithm to update the Gaussian parameters. Propagating through time induces cyclicity in the factor graph, but this isn't a showstopper. It just requires updating iteratively, making filtering forward and smoothing backward through the graph until convergence. While in other settings this may be a problem, here it is an acceptable tradeoff.

Unfortunately, not everything is Gaussian. The comparison factor at the base of the graph, which determines the winner of a match, sends a non-Gaussian message to the performance

difference variable d_i . According to the message passing update rule, the message it sends is a binary 1 or 0. This means that the marginal probability $p(d_i)$ isn't Gaussian, destroying any ability to have nice updates using Gaussian conjugacy properties.

To resolve this, one approach is to adopt expectation propagation [27]. Instead of evaluating the naughty term directly, it is possible to proceed by using a Gaussian approximation in its place. This involves two approximations. First, it is necessary to approximate $\hat{p}(d) \approx d(x)$ using moment matching to minimize the KL divergence between the Gaussian approximation and non-Gaussian marginal. Substituting message passing equation 3 into equation 1:

$$\hat{p}(d) = \hat{m}_{f \rightarrow d}(d)m_{d \rightarrow f}(d) \iff \hat{m}_{f \rightarrow d}(d) = \frac{\hat{p}(d)}{m_{d \rightarrow f}(d)}$$

Now, there is a usable, conforming approximation of the faulty message and marginal. It is possible to proceed with message passing using nice Gaussian updates.

Now, in order to evaluate the effect of round type on outcome, it is necessary to incorporate covariates into the model. The natural way is to treat them as additive factors to player skill. TrueSkill is designed to be flexible to account for variable team sizes, by treating team performance as the sum of all player performances on that team. Then, we just need to include a skill effect "teammate" e_i , resulting in the following process generating game results:

1. draw player skills at time t : $s_i^{(t)} \sim N(\mu_i^{(t)}, \sigma_i^{(t)2})$, $s_j^{(t)} \sim N(\mu_j^{(t)}, \sigma_j^{(t)2})$ and draw skill effect $e_i^{(t)} \sim N(\mu_{e_i}^{(t)}, \sigma_{e_i}^{(t)})$
2. add noise from passage of time: $s_i^{(t+1)} = s_i^{(t)} + N(0, \tau^2)$, $s_j^{(t+1)} = s_j^{(t)} + N(0, \tau^2)$, $e_i^{(t+1)} = e_i^{(t)} + N(0, \tau_e^2)$
3. add performance modifier: $p_i = s_i^{(t+1)} + N(0, \beta^2)$, $p_j = s_j^{(t+1)} + N(0, \beta^2)$
4. calculate team performances: $t_i = p_i + e_i$, $t_j = p_j$
5. declare winner $y_{i,j}^{t+1} = \delta(t_i - t_j > 0)$

Covariates are treated the same as players with one exception: $\beta = 0$ for covariates to prevent them from injecting unnecessary variance into estimates. Although in my setting this is simple because my covariates are all indicator variables representing match type, a more complex version of this approach is implemented in TrueSkill2 [26].

5.2 Experimentation and Results

When fitting the TrueSkill model, I treat each debate team as an individual player. Before I include round type in the model, I use maximum likelihood estimation to learn model hyperparameters. I learn time deviation parameter τ and optimal prior distributions for

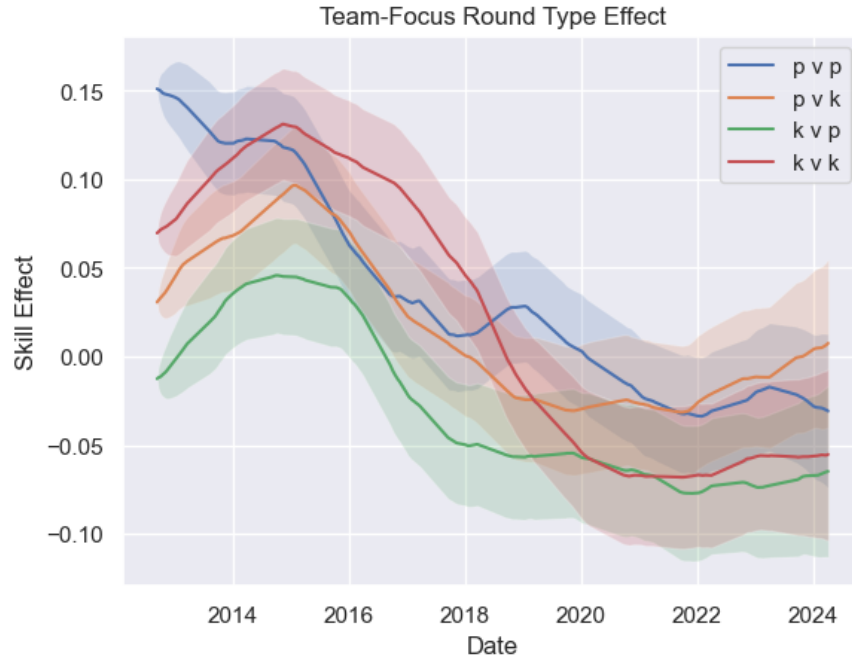


Figure 6: Team-focus round type effect. Positive values denote an increase in the affirmative team’s performance. The affirmative team’s advantage can be seen declining over time across each round type.

the skill of teams starting their careers in the Novice, Junior Varsity, and Varsity divisions respectively. Once tuned, the model predicts the results of nearly 79.95% of debate rounds successfully before considering round types.

Then, fixing the hyperparameters for teams, I learn hyperparameters for the different round types. As a reminder, I only look at the effect of round type on debates in the varsity division, as that is where it is possible to reliably classify teams and where the impact of round type is most relevant. I learn optimal hyperparameters for the team-focus approach, the team-focus approach with different effects for different skill levels, and the comprehensive approach.

The team-focus approach marginally increases the retrodiction accuracy of the model. Now, it correctly calls 80.01% of rounds. This improvement does not appear to be attributable to a large difference between biases across round types. Rather, across all round types, there appears to be a marginal bias towards the affirmative side before 2016. After 2020, there appears to be marginal negative side bias for all round types. However, taking skill level into consideration, the story seems different.

The next approach is stratify the team-focus approach across skill levels. I divided debates into five categories using the skill scores from the first model, which ignores round type. I classified each debate round based on the skill of the teams involved. If both teams are at least in the fifth quintile, the round is labelled 5. Then, if both are at least in the top

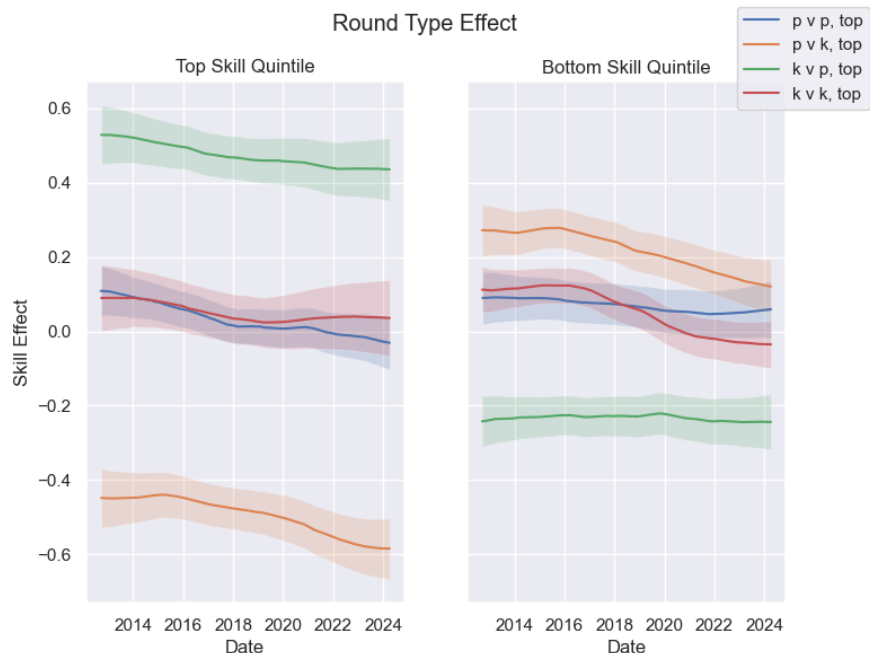


Figure 7: Round type effect in top and bottom skill quintiles. Notably, in debates between kritikal and policy teams advantage the kritikal team in high-skill debates, but advantage the policy team in low-skill debates.

two quintiles, the round is labelled 4, and so on. Quintiles are chosen because roughly 40% of teams advance to elimination rounds, which aligns them with a useful natural cutoff. Compared to the bare skill-focus approach, this marginally improves prediction accuracy and provides more interesting results. It seems that the effectiveness of the kritik increases with skill. In the uppermost quintile, the kritik has a substantial advantage against policy strategies regardless of side selection. With evenly skilled teams, a modifier of 0.5 suggests the kritikal team will win nearly 64% of the time. This aligns with community sentiment that there are a large number of upsets in debates between top policy teams and kritikal teams. The advantage of the kritik becomes increasingly smaller in each quintile. At the bottom, kritikal teams are disadvantaged against policy teams. This also makes sense, as the kritik is a complex argument that unskilled debaters are likely to struggle to explain.

Lastly, the comprehensive approach takes judge type into consideration as well as team type. This approach is the most useful in predicting debate round outcomes, marginally increasing retrodiction accuracy to 80.08%. Parameter estimates show that clash judges are, in aggregate, essentially unbiased when deciding rounds between policy v kritik and kritik v policy debates. With the largest estimated skill effect at 0.15, which equates to a 54% winrate for the affirmative with teams of equal skill, clash judges are fairly impartial. On the other hand, policy judges and kritikal judges do appear biased towards ideologically aligned teams. Policy judges also appear more biased than kritikal judges. This is especially apparent when considering kritik v policy debates, which show a skill effect of nearly -0.4 . With equal skill, a kritikal affirmative team against a policy negative team is expected to

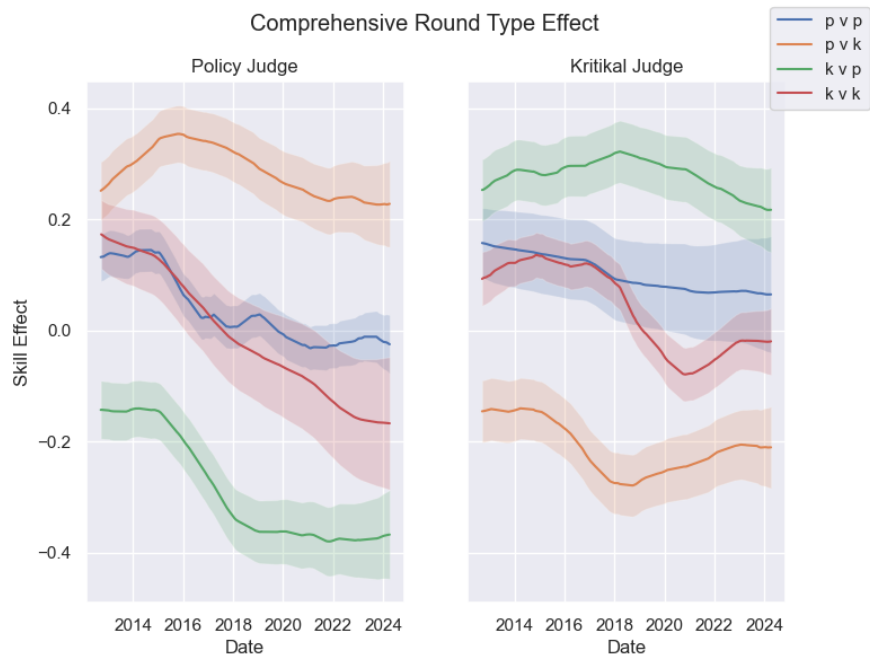


Figure 8: Comprehensive round type effect. Policy judges appear more heavily biased against the kritik than kritikal judges are against policy arguments.

win only 39% of the time in front of a policy judge. This makes sense, as kritikal affirmative strategies typically deviate further from the plan-focus model of debate than kritikal negative strategies.

Finally, it's worthwhile to note that there is no longer substantial bias in favor of the affirmative team in policy v policy or kritik v kritik debates. There appears to have been a sharp decline in side bias within styles of Intercollegiate Policy Debate. This decline, which starts in the 2016-2017 academic year and levels out around the 2019-2020 academic year, does not appear to be an artifact of the model. Simply aggregating ballots, without examining round type, shows a similar rapid shift in favor of the negative. Although inexplicable without further research, it is reasonable to believe that this is caused by a mix of conscious correction in judging, a shift towards resolutions more favorable for the negative, and a political climate that makes political action both more necessary and more difficult.

6 Conclusion

By leveraging knowledge the Mutual Preference Judging process, it is possible to determine the ideological leanings of teams and judges. This allows for the classification of teams and judges into the categories germane to the debate community: “policy”, “kritikal”, and “flex/clash.” From these categories, and the results of debates, it is possible to see how their interaction actually effects competitive outcomes. It appears that, although struggling compared to policy styles at the skill level of most debaters, kritikal debate styles are incredibly effective at the top levels of competition. I also find that a judge’s ideology substantially influences which team is declared the winner of a debate round.

There are two big takeaways for teams and coaches bent on competitive success. Teams should absolutely rank judges according to ideological alignment. Because judging greatly affects the odds in policy vs kritik and kritik vs policy debates, any alternative risks gifting opponents a sizeable advantage. Second, it is necessary for skilled policy teams to rethink their strategies against the kritik. Although standard strategies against the kritik seem to be effective at lower skill levels, highly skilled policy teams substantially underperform when faced with kritikal opponents.

Last, it’s important to note that aff side bias is not set in stone. The winrate of the affirmative is constantly in flux and is dependent on the topic, the skill level of debaters, and the ideological leanings of teams and debaters. In modern policy v policy and kritik v kritik debates, neither the affirmative nor the negative is consistently disadvantaged. Although side bias has existed, it has rarely been substantial and should not be treated as inherent to the structure of Intercollegiate Policy Debate.

References

- [1] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman. Pyro: Deep Universal Probabilistic Programming, Oct. 2018. arXiv:1810.09538 [cs, stat].
- [2] D. M. Blei, A. Ng, and J. Michael. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 2003.
- [3] M. Blondel, O. Teboul, Q. Berthet, and J. Djolonga. Fast Differentiable Sorting and Ranking, June 2020. arXiv:2002.08871 [cs, stat].
- [4] R. A. Bradley and M. E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4):324–345, 1952. Publisher: [Oxford University Press, Biometrika Trust].
- [5] P. Dangauthier, R. Herbrich, T. Minka, and T. Graepel. TrueSkill Through Time: Revisiting the History of Chess. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

- [6] W. D. Decker and J. T. Morello. Some Educational Difficulties Associated with Mutual Preference Debate Judging Systems. *The Journal of the American Forensic Association*, Mar. 1984. Publisher: Taylor & Francis.
- [7] T. Dillard-Knox. Against the grain : the challenges of black discourse within intercollegiate policy debate. *Electronic Theses and Dissertations*, Dec. 2014.
- [8] T. Dozat. Incorporating Nesterov Momentum into Adam. Technical report, Stanford University, 2016.
- [9] S. Duffield, S. Power, and L. Rimella. A State-Space Perspective on Modelling and Inference for Online Skill Rating, Sept. 2023. arXiv:2308.02414 [stat].
- [10] R. L. Geiger. *The History of American Higher Education: Learning and Culture from the Founding to World War II*. Princeton University Press, 2015.
- [11] L. Giella. “Mr. Tabroom” doesn’t let data and code interfere with debate’s human side, Sept. 2023. Section: News.
- [12] R. Glass. The DebaterCast Episode 03 - Gary Larson, Aug. 2018.
- [13] M. E. Glickman. Paired Comparison Models with Time-Varying Parameters:. Technical report, Defense Technical Information Center, Fort Belvoir, VA, May 1993.
- [14] A. Grover and J. Leskovec. node2vec: Scalable Feature Learning for Networks, July 2016. arXiv:1607.00653 [cs, stat].
- [15] C. C. Henson and P. R. Dorasil. An Empirical Analysis of Judging Bias by Sex, Region & Side, July 2011.
- [16] C. C. Henson and P. R. Dorasil. Judging bias in competitive academic debate: the effects of region, side, and sex. *Contemporary Economic Policy*, 32(2):420–435, Apr. 2014. Publisher: Blackwell Publishers Ltd.
- [17] R. Herbrich, T. Minka, and T. Graepel. TrueSkill™ : A Bayesian Skill Rating System. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.
- [18] T. Hofmann. Probabilistic Latent Semantic Analysis, Jan. 2013.
- [19] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An Introduction to Variational Methods for Graphical Models. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 105–161. Springer Netherlands, Dordrecht, 1998.
- [20] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, Jan. 2017. arXiv:1412.6980 [cs].
- [21] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes, Dec. 2022. arXiv:1312.6114 [cs, stat].

- [22] F. Kschischang, B. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, Feb. 2001.
- [23] E. Lee. *Debate and Self-Authorship: A Narrative Case Study of Competitive Intercollegiate Debate and the Development of the (Cross) Examined Life*. PhD thesis, The University of Georgia, ATHENS, GEORGIA, Dec. 2017.
- [24] J. D. Leeuw, K. Hornik, and P. Mair. Isotone Optimization in *R* : Pool-Adjacent-Violators Algorithm (PAVA) and Active Set Methods. *Journal of Statistical Software*, 32(5), 2009.
- [25] A. Loudon. “Permanent Adaptation” - The NDT’s Last 50 Years. *Speaker and Gavel*, 50(2), 2013.
- [26] T. Minka, R. Cleven, and Y. Zaykov. TrueSkill 2: An improved Bayesian skill rating system. 2018.
- [27] T. P. Minka. Expectation Propagation for approximate Bayesian inference, 2001. arXiv:1301.2294 [cs].
- [28] M. E. J. Newman. Efficient Computation of Rankings from Pairwise Comparisons. *Journal of Machine Learning Research*, 24, June 2023.
- [29] C. Palmer. The Massacre of the Novii, July 2012.
- [30] R. J. Park. Muscle, Mind and “Agon”: Intercollegiate Debating and Athletics at Harvard and Yale, 1892-1909. *Journal of Sport History*, 14(3):263–285, 1987. Publisher: University of Illinois Press.
- [31] B. Perozzi, R. Al-Rfou, and S. Skiena. DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, Aug. 2014. arXiv:1403.6652 [cs].
- [32] R. Ranganath, S. Gerrish, and D. Blei. Black Box Variational Inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pages 814–822. PMLR, Apr. 2014. ISSN: 1938-7228.
- [33] S. R. Reid-Brinkley. Voice Dipped in Black: The Louisville Project and the Birth of Black Radical Argument in College Policy Debate. In N. S. Eidsheim and K. Meizel, editors, *The Oxford Handbook of Voice Studies*, page 0. Oxford University Press, July 2019.
- [34] S. R. Reid-Brinkley. Celebrating the 20th Anniversary of the Louisville Project. *The Journal of the Cross-Examination Debate Assosiation*, 38(Identity, Performance, & Debate), June 2023.
- [35] M. Schmidt, G. Palm, and F. Schwenker. Spectral graph features for the classification of graphs and graph sequences. *Computational Statistics*, 29(1-2):65–80, Feb. 2014. Num Pages: 65-80 Place: Heidelberg, Netherlands Publisher: Springer Nature B.V.

- [36] M. Stannard. Speculations on the Effects of Mutual Preference Judging on Debate Communities, 2000.
- [37] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1139–1147, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.
- [38] W. F. University. A Century of Intercollegiate Debate.