Sports Tournament Predictions Using Direct Manipulation

Romain Vuillemot ▪ *Université de Lyon*

Charles Perin ▪ *University of Calgary*

An advanced interface for sports tournament predictions uses direct manipulation to allow users to make nonlinear predictions. Unlike previous interface designs, this proposed technique better matches the way people actually make predictions, such as first choosing the winner and then filling up the rest of the bracket.

Millions of fans consume information related to sports every day, through media or by watching games. The most enthusiastic fans make informed decisions by betting on team game performances or on whole tournaments. Such prediction activity involves a heavy reasoning mental process that usually sums up a whole body of distributed information (such as statistics on players, individual players' performances, and teams tactics) to determine who will win and who will lose. This prediction process can also be subjective, involving human intuition, individual preferences on teams or players, or just randomness.

Despite the strong interest in predictions and their relevance to soccer and many other sports, few interfaces exist that support this process. The interfaces that do exist are text-based and force users to go through a long list of steps, such as predicting each outcome for each team. This process is repetitive and never allows users to predict the big picture first and then refine it. For instance, someone might want to predict who will win the World Cup first and then detail the game outcomes to support this claim.

Improving the support for predictions is difficult because the mental process behind a prediction usually requires guessing unknown values from previous observations. It is also difficult because the guess must comply with the constrained structure of a given tournament. This structure is usually twofold, involving a *group phase* where all the teams or subsets of teams play against each other either once or twice and a *bracket phase* where teams can be eliminated against a specific opponent. Improving predictions will have an impact beyond sports, such as in domains such as weather forecasting or economics.

The goal of our work is to improve a user's prediction creation process. For this purpose, we designed, implemented, and evaluated an interface that supports soccer tournament predictions (see Figure 1). It implements the constraints of the tournament structure and makes it possible for people to predict—and adjust that prediction—for the big picture of a tournament outcome in a single view. Our process involved the following steps:

- We defined a list of design criteria to understand the current needs from soccer tournament prediction input interfaces and the state of the art in human-computer interaction (HCI), including direct manipulation.[1]
- We designed and implemented an advanced visual interface that complies with the design criteria by using direct manipulation principles to let users drag and drop team icons (or badges) toward their final position in a single view.
- We also designed and implemented a set of novel visualization logs to make sense of the 504,307 recorded interaction logs from the 3,029 visitors in order to identify and discuss interesting behaviors.

Overall, we contribute evidence that sport enthusiasts can make sport tournament predictions using an advanced, nonlinear direct manipulation interface. From our analysis of interaction

 Published by the IEEE Computer Society 0272-1716/16/$33.00

logs, we provide a list of strategies that people employ to make predictions. These strategies will help inform the design of prediction interfaces. These results could be applied to various fields involving predictions such as economics, finance, and weather forecasting that are currently run by automated models that could benefit from more user-generated data.

Context

Sports tournaments are competitions involving multiple competitors, such as sports teams or individuals. Thus, we broadly define a *prediction* as the act of determining future, unknown values. We focus on the predictions that are generated in a quantitative, user-generated way, without any model or automated assistance.

Sports Tournaments

We consider sports tournaments with a common configuration: the combination of a ranking phase and a bracket phase (see Figure 2). Usually, the ranking phase precedes the bracket phase (and is called the group stage), during which teams play against each other once or twice. The left side of Figure 2 shows such a group stage occurring at the beginning of a tournament. The best-ranked teams after the group phase enter the second phase where teams are assigned an opponent and are eliminated after each round. The right side of Figure 2 shows such a stage that looks like a bracket converging to the winner of the tournament.

Schedules

Tournaments usually have two ways of planning games for the elimination bracket. Either a draw has decided a schedule of games in advance (for example, the winner of group stage A will meet the second-best ranked team of group stage B) or a draw occurs at each round of the bracket. Some draws are not uniform, because they often include some constraints. For example, there can be country-specific constraints where teams from the same country cannot play against each other.

Tournament Predictions

As we stated earlier, a prediction consists of determining a set of future values. A tournament prediction is the guess of all game outcomes, which indirectly decide of the tournament's winner. This decision is made in two consecutive steps: first the calculation of the group stage results as a ranking, and then the progressive eliminations during the bracket phase.

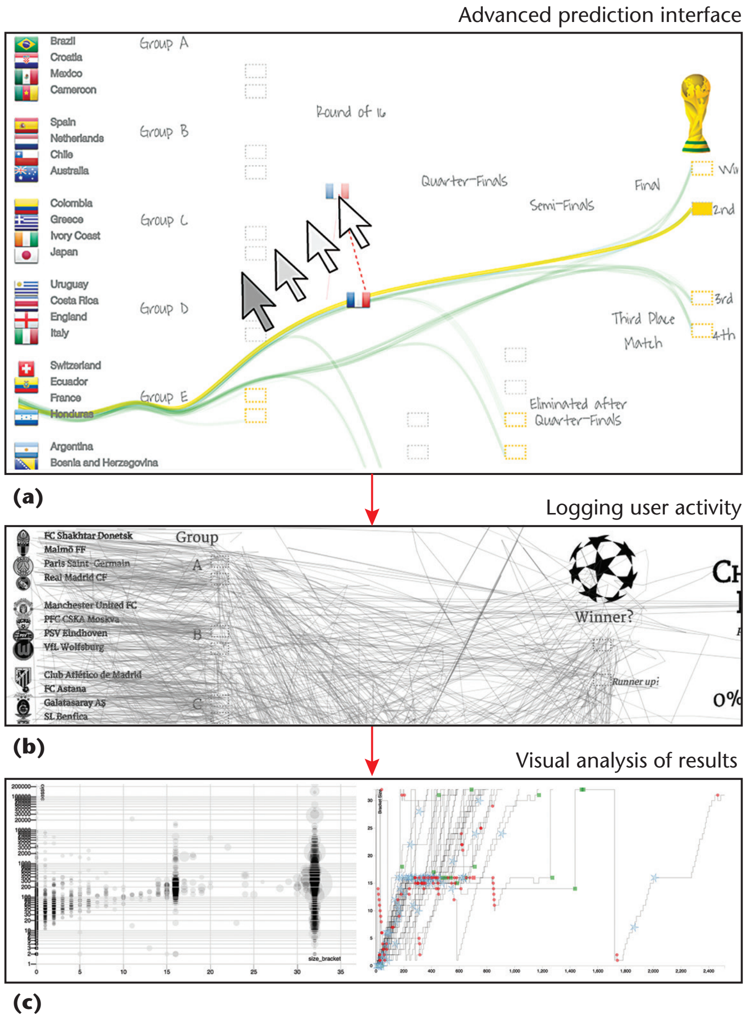

Figure 1. Sports tournament predictions. (a) We present an advanced interface to help users make soccer game predictions by simply dragging and dropping teams badges on the tournament bracket. (b) We logged and plotted user activity and (c) performed a visual analysis of those logs to identify user behaviors.

This is where there is a divergence between the current interfaces and the way users perform predictions. Current interfaces only implement the structure of predictions illustrated in Figure 2 as the sum of a series of games. The user's mental model, however, follows the opposite path: it first concentrates on the tournament's outcome, and then focuses on finding individual game results given the outcome.

Prediction Space

A *prediction space* is the set of all the possible results that may occur in a tournament. The prediction space is usually large because it is the permutation of all possible games and outcomes. However, the further the tournament progresses, the smaller the prediction space becomes. Right before the final game, the prediction space has a size of two because (since all the games have

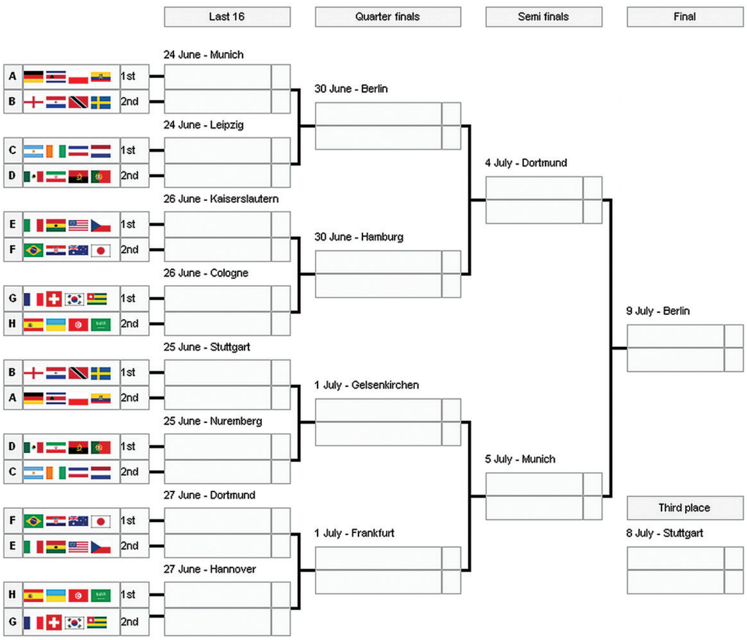

Figure 2. Example of a soccer tournament configuration. On the left, the group stage (column with teams badges) results into a ranking after games have ended. On the right, the elimination bracket phase (for the best group stage teams) determines who the winner of the competition will be.

already occurred except the last) there are only two possible outcomes.

Design Criteria

To the best of our knowledge, there is no best practice or previous work on input interactions to support user-generated predictions. Therefore, we conducted a pre-study to capture any best practice in the building of interfaces to support tournament predictions. Informed by this study and related work in HCI, we derived a set of design criteria to support users in their predictions processes.

Pre-study

We identified 27 websites dedicated to sports betting that support bracket entries. Fifteen of those websites were dedicated to soccer in particular and 12 to betting in general. We found that betting websites sometimes are already populated with prediction suggestions (for example, providing odds) and allow users to enter their predictions. Figure 2 shows the typical interface for sport brackets (both for predictions and results communication) that we observed. Most of the websites used standard HTML widgets for input such as checkboxes, dropdown lists, and text inputs, and they organized team options using a bracket layout. To successfully enter their predictions, users have to manually input data for each team by game and start with the first round of games to end up with the final (from left to right in Figure 2).

When we looked at nonsports-specific websites, such as general-purpose newspapers, we found more innovative input interfaces. For example, Bloomberg offered a soccer bracket that let users predict the game outcomes of the 2014 World Cup (see www.bloomberg.com/visual-data/world-cup/). FiveThirtyEight also used brackets for the same event, but only as a way to communicate its model's probability of outcome for each game and no user interaction was allowed (see http://fivethirtyeight.com/interactives/world-cup/). Finally, a 2012 *New York Times* article communicated predictions for the 2012 US presidential elections by letting users interact with the prediction space; the 512 possible configurations of results could be filtered by various scenarios depending on state-level elections results.[2]

Design Criteria

From both the previous analysis of current systems and our understanding of sports prediction in news articles (written in text), we derived the following set of design criteria for a prediction interface that supports current needs:

- *C1 Partial*: Allow partial predictions, especially ones involving the user's favorite or the best teams (such as a winner or finalist).
- *C2 Levels of detail*: Allow users to fill in the scores, the number of goals scored, or only the outcomes of games.
- *C3 Nonlinear*: Let users fill in the prediction from multiple entry points (such as semifinal or quarterfinals), not necessarily starting with the first games of the tournament.
- *C4 Reversible*: Any action or the whole prediction can be reverted to its previous or initial state in case of mistakes or a change of mind.[1]
- *C5 Structure*: Show the structure, such as the tournament rules, to let users know the connection and dependencies between each round.
- *C6 Suggested interactions*: Show visual cues to tell users what and how they can interact with graphical elements on the screen[3] (especially if widgets are not used).
- *C7 Familiarity:* Keep many aspects of the domain such as team badges and bracket layout to shorten the learning phase.

From our pre-study, we observed that prediction interfaces usually implement C4 and C5. The familiarity criteria C7 is also commonly implemented, mostly by using teams badges or logos (which are small glyphs), along with team names in text.

Direct Manipulation for Predictions

Informed by the previous design criteria, we introduce an interface to better help sport enthusiasts make sports tournament predictions. We focused on making it is easy to learn the input interface so users could focus on the mental process of predicting. The interface provides an overview (C1) of the prediction space (all possible results for teams), which makes the structure of the tournament (C5) visible and is used as a suggested interaction (C6) to inform users about the path a team can follow in the competition. Leveraging this overview, users can start by picking the winning team or semifinalists—that is, they can perform a nonlinear bracket filling (C3).

Figures 1 and 3 illustrate the technique in action. Figure 4 shows the main interaction, which consists of three steps:

1. The user selects a team by clicking on a team's badge and then dragging it toward the bracket.
2. As the user drags the team badge, a visual cue (a line) shows which stages of the tournament the team can be dropped into. The visual cue is green if the team can be dropped and is red otherwise.
3. To make a prediction, the user releases the mouse button to stop dragging the team badge, which snaps it to the closest game it can be attached to.

This series of steps constitutes an extension of the well-known Drag-and-Drop technique that uses direct manipulation[1] of elements (called objects of interest). Following earlier research,[4] we call our extension *Drag-and-Snap*. It uses a team's possible paths as snapping constraints and visual cues for users to identify where team badges can be dropped.

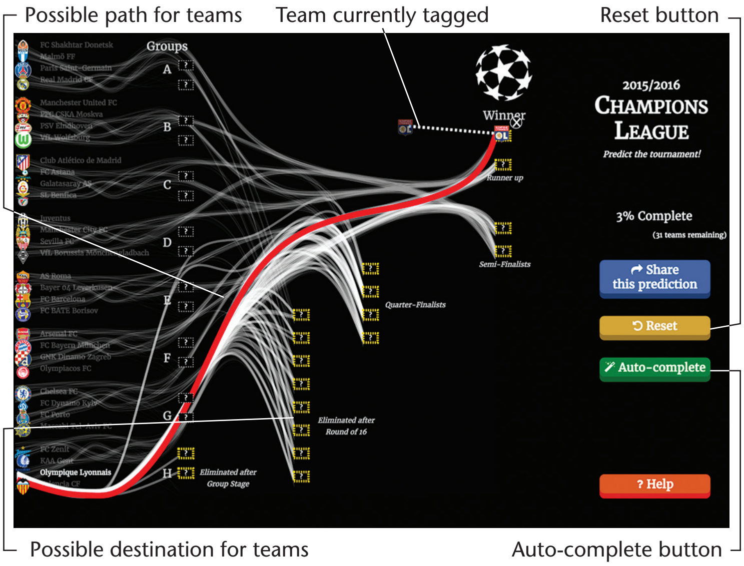

Figure 3. Screenshot of the second prototype's interface. The background shows possible paths for teams, and the interface includes an action button such as a reset button that lets users start over and an auto-complete button to automate the bracket's filling.

Related Direct-Manipulation Techniques

A body of similar techniques also exist that aim to help users when they are dragging elements.

The Drag-and-Pop technique moves potential target icons toward the user's current cursor location, thereby allowing the user to interact with these icons using comparably small hand movements.[5] Drag-and-Pick moves all the icons toward the cursor in the direction of the mouse motion.[5]

In another example, a *New York Times* online article used the Drag-and-Draw technique to let users freely draw a line on a chart to predict how family income determines children's college chances.[6] The horizontal axis is the parent's income percentile (from poorest to richest), and the vertical axis is the percent of children who attend college.

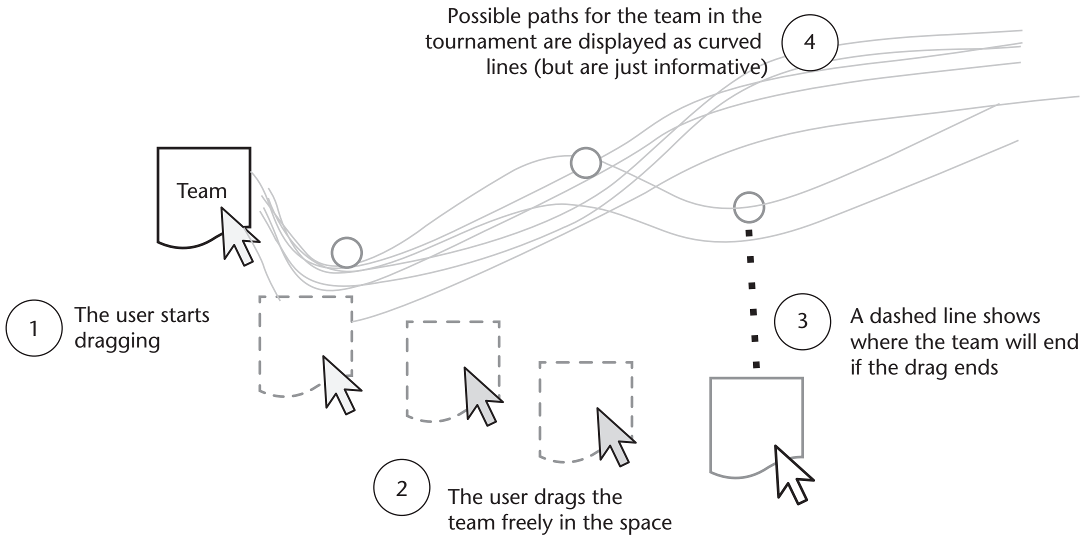

Figure 4. Three main steps to using the Drag-and-Snap interface. Users can make a prediction for a single team at a time.

The Drag-and-Update technique lets users update data graphics by dragging items along a path over time.[7,8] Lastly, the Dwell-and-Spring technique uses the metaphor of springs to enable users to undo direct manipulations.[9]

All these techniques use visual feedback that either connects elements or shows additional information. Our proposed Drag-and-Snap technique is most closely related to Drag-and-Pop because it connects the current team to potential placeholders for the prediction. In addition, it adds an extra layer of information during the dragging phase by showing the full prediction space.

Proxy Object of Interest

When dragging an object, the user's mouse does not necessarily follow the dragged object (for example, if the object can only move within a perimeter). In previous work, it is common to use *proxy elements* as a way of duplicating the object of interest. For example, the Drag-and-Pop and Drag-and-Pick techniques duplicate icons to suggest them as a target destination.

We used a similar design in our technique to ensure that the team badge always follows the mouse (as a proxy) while the original badge remains on the bracket. This is visible in Figure 1a, where the flags are duplicated: one strictly follows the trajectory, the other strictly follows the mouse point, and both are connected with a dotted line to show they are the same entity.

Trajectories Design

Our interface shows all the possible tournament predictions in the background to suggest paths to follow during drag and drop. Previous research has suggested using paths to connect elements to show previous interactions[10] as well as forthcoming ones. The Drag-and-Pop technique uses an elastic rubber band to help users understand what it connects and a visual cue to convey how far away the target is.[5] This is similar to related work[10] where a band connects items using various path styles to simulate motions (such as motion blur or speed lines).

All those designs are not suited to our case, where thousands of trajectories are often packed on a small part of the screen. Thus, we use a visual design similar to the trajectories of chessboard pieces, as in Martin Wattenberg's artwork *Thinking Machine* (see www.bewitched.com/chess.html). Each path is an arc with a jitter that makes it visually unique. This mostly aims to prevent clutter,[11] but it also conveys the dynamics of piece trajectories.

First Prototype

We created our first interface prototype implementing the Drag-and-Snap technique for the 2014 FIFA World Cup tournament (Figure 1a). We released it one week before the start of the first game (12 June 2014) to let enthusiasts predict the outcome of the tournament. The purpose of this release was to validate the design, detect any usability flaws, and collect qualitative feedback for further improvement. The prototype and its source code are available at https://github.com/romsson/worldcup14-interactive-bracket.

Prototype Design

The prototype follows the description of the technique we have provided here and some domain-specific designs (C7). The teams participating in the tournament have an initial position on the leftmost part of the screen, before they enter the group stage. The user makes a prediction by dragging and dropping a team badge from the left to the right until it reaches a game (placeholder). A visual hint indicates the closest trajectory to the dragged team (and not the closest game), and if the users stops dragging the badge, then the team will be assigned to the game at the end of this path.

The interface also contains action buttons (see Figure 3 for their location on the screen), including a reset button (C4) that lets users start over and an auto-complete button to help them fill out the bracket completely. A bracket can be partially completed (C1) and by starting at different levels (C3). As a complement to the background showing the prediction space, we provided animated GIFs[12] that explain the Drag-and-Snap interaction (C6). Finally, we used team badges and the tournament logo for familiarity (C7).

This interface only allows users to predict outcomes of group stages once completed and not the outcome games in those group stages. Thus, we do not offer the lowest level of details possible (C2), which is a tradeoff to keep the interface simple.

Feedback

We released the interface and advertised on forums dedicated to soccer and social media with appropriate hashtags. During the week before the World Cup began, 2,932 unique users visited the interface, with an average session duration of 38 seconds. We observed 141 Tweets, 56 Facebook shares, and 11 Google+ shares. We collected qualitative feedback in an unstructured manner. Overall, the users were enthusiastic, but they noticed three issues.

First, they found that snapping by closest trajectory was not intuitive and was difficult to use, especially when there were multiple trajectories at the initial position, making it hard to select a specific one. Second, they wanted to make predictions like playing a chess game where teams are progressively moved game by game, rather than having to set the final position. Lastly, they wanted to be able reset individual teams rather than the whole interface at once.

Second Prototype

Based on the qualitative feedback we received for the first prototype, we designed and released an improved interface (see Figure 3) for the 2015/2016 Union of European Football Associations (UEFA) Champions League. This prototype and its source code are available at https://github.com/romsson/ucl16-predictions.

In this version, we improved the Drag-and-Snap technique so that teams snap to the closest valid game (placeholder) instead of the closest prediction trajectory. We also added the possibility to reset individual teams (C4) by adding a small close button on each badge so that it can be reset it to its initial position. In addition, we changed the timing of animations and added question marks within empty placeholders and the use of modals to explain features. Those changes aim to make the technique ecologically valid (C7), similar to a standard website. Finally, we adapted the prediction space to the configuration of the UEFA Champion's League, which slightly differs from the World Cup tournament structure: after the group stage, the games for every round are decided by draw, which increases the number of prediction paths for each team.

Technical Notes

We implemented the two prototypes using the JavaScript toolkit dragit,[8] which uses D3 to handle mouse events. We used D3 to represent the trajectories and the snapping guides using SVG. Users can interact with the prototypes with simple mouse clicks and mouse moves, and both prototypes run in any recent Web browser. Because performance is key with direct manipulation interfaces, we generated an image containing all the trajectories and displayed it as a background; we only highlighted user predictions using a SVG overlay because drawing each trajectory line using SVG would have greatly impaired performance.

One last technical decision was to make the prototype *stateful*, which means that after the user interacts with the system, his current prediction can be recorded and retrieved with the page's URL. This makes it possible to share a prediction via a link to the page, which can then be used to complete or modify the prediction or as a starting point by other people. In other words, this featured immediately allowed asynchronous collaborations by users around predictions.

Evaluation Methodology

Once our interfaces were released, real-world users interacted in a natural and comfortable environment similar to using a regular website. Thus, they behaved the way they would have normally, rather than being distracted or feeling "tested" in a formal experimental setting. Moreover, the prediction task cannot be assessed in terms of regular metrics such as time of error. First, there is no baseline for correct answers since the results of the correct prediction (in this case, the UEFA Champions League) were unknown before May 2016 (before this article was submitted for publication). Also, even if this baseline existed, it is perfectly normal to be wrong. Finally, it does not make sense to make predictions for an event that already happened.

Once our interfaces were released, real-world users interacted in a natural and comfortable environment similar to using a regular website.

As a result, we focus on understanding how sport enthusiasts made predictions rather than on assessing whether or not they made accurate predictions using the interface. The only valid baseline would be to ask users in advance what their predictions would have been and if the interface was helpful. However, we believe that a users' predictions can sometimes be progressively formed as they are using the interface and thus cannot be known in advance.

To capture distant user activity, we used *logging*, a nonintrusive mechanism to capture in-the-wild users' activities. Research has shown that this approach is efficient for attempts to learn about user behaviors.[5] Enthusiasts can interact with the system without any interruption or distraction related to the experiment. We instrumented the technique with a server-side mechanism to record key interactions from each visitor.

Table 1 shows the list of events we recorded. We did not record every mouse move because it have

Table 1. List of user interactions events we recorded during the evaluation.

Focus	Type	Details
Webpage	Init (auto)	The user opens the webpage
	End (auto)	The user closes the webpage
Team badges	Hover	Mouse hover on a team badge
	Drag	Dragging starts
	Drag (end)	Dragging ends
	Close	The user clicks on the close button
Teams	Hover	Mouse hover on a team placeholder
Buttons	Click	Reset the prediction
	Click	Auto-complete the prediction
	Click (submit)	Share the whole prediction

would led to too much data and would probably capture noise over signals.

Visual Analysis of Logs

We designed and implemented a set of visualization tools to explore the captured user logs. The goal of those tools is to assist us in better translating user interactions (such as click and drag) into complex behaviors and eventually identify recurring patterns or any flaws in the interaction.

Step One: Plotting All Logs

The first step was to get a big picture of the 519,129 interaction logs we recorded from 4,739 visitors. We cleaned up the logs by removing sessions with durations that were either under 1 second or more than 45 minutes as well as those with less than five interactions. We ended up with 3,029 unique visitors and 504,307 interactions.

Figure 5 provides an overview of all visitors by showing each of them as a circle. The circle's size indicates the visitor's number of interactions (max 4,354 interactions). The horizontal axis is the bracket size (32 is a full bracket), and the vertical axis is the time spent on the website (in seconds with a logarithmic scale). We added a low opacity to circles to better show the distribution of dense areas by reducing overplotting.[11]

Step Two: Grouping Visitors

We visually identified three main groups of visitors (as vertical stripes from left to right in Figure 5):

- *G0* contains 1,079 visitors (36 percent of the total visitors) who did not fill the bracket (the vertical line of circles with zero predictions in Figure 5). Those visitors left the interface unchanged with no interaction and with an empty bracket.
- *G2* contains 287 visitors (9 percent of the total visitors) who performed interactions and partially completed the bracket (the circles with at least one prediction but who did not complete the bracket in Figure 5).
- *G3* contains 1,653 visitors (55 percent of the total visitors) who completed the full bracket (the vertical line of circles with 32 predictions in Figure 5).

We focused our analysis on G3. This group does not necessarily contain users who completed the

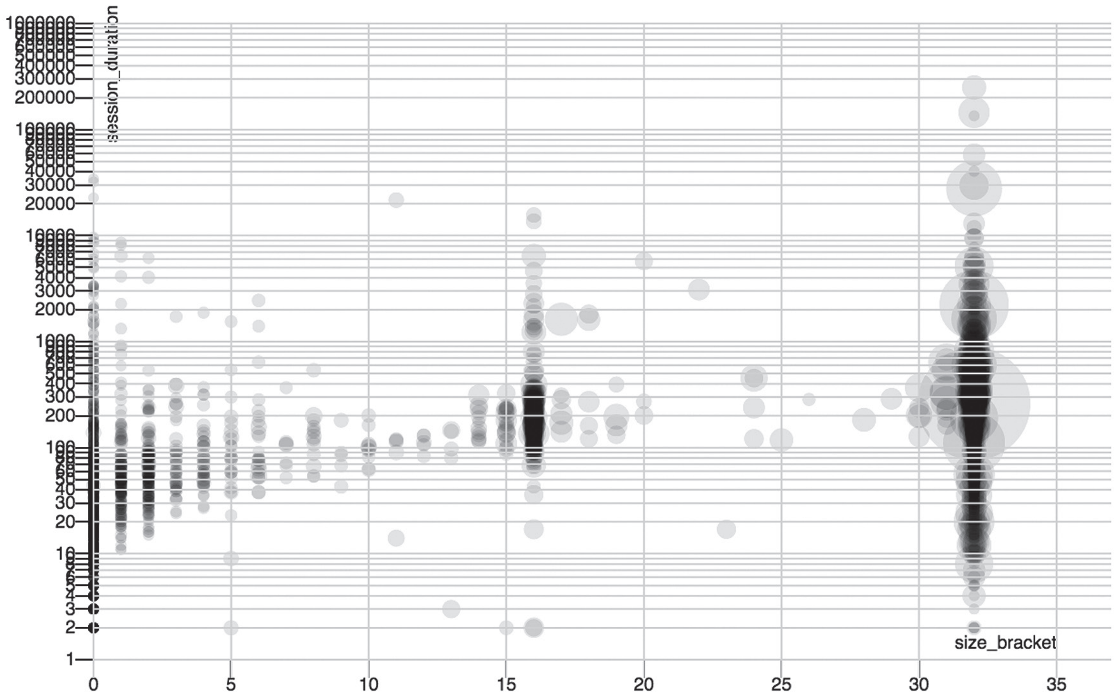

Figure 5. Scatterplot of visitors. Each circle represents a visitor. The circle size indicates the visitor's number of interactions. The horizontal axis is the size of the final bracket (ranging from 0 to 32). The vertical axis is the time spent using the interface (in seconds, logarithmic scale).

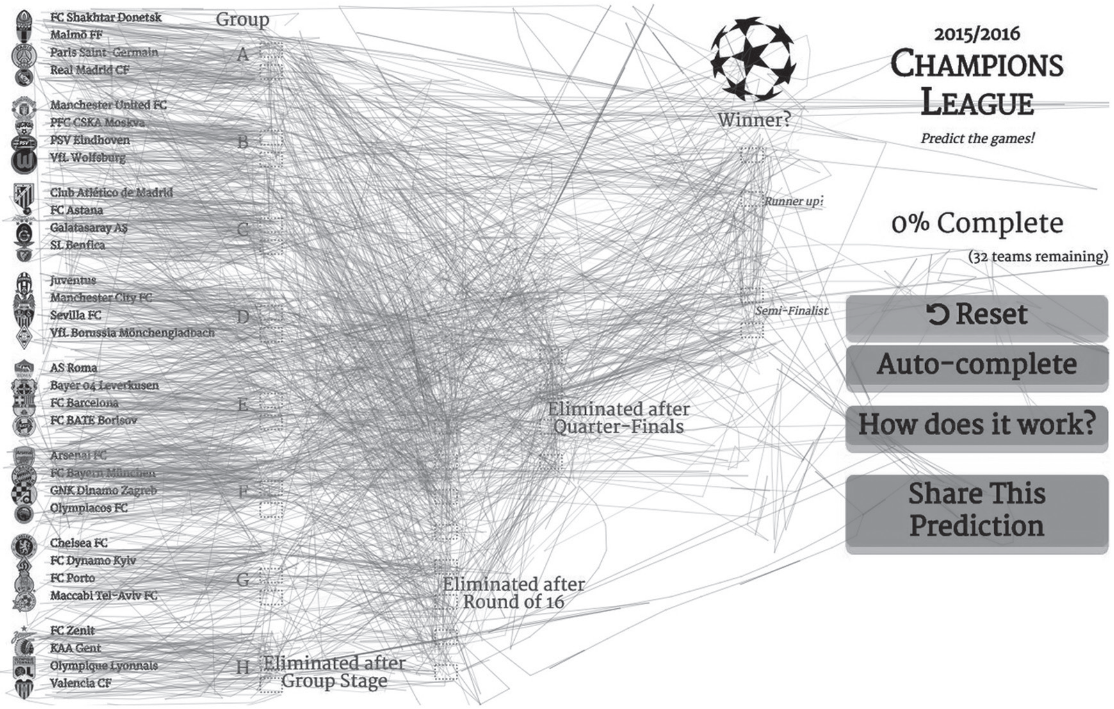

Figure 6. Aggregation of dragging interactions. Each line connects all the visitors' drag and drop interactions for the corresponding team.

prediction themselves. Indeed, since the bracket can be shared, many of these visitors may have started their predictions with an already completed bracket. We found that in G3, 11.8 percent of the visitors (198 visitors) started from scratch, and we call this group G3a. We identified a subgroup of G3a called G3b containing 56 visitors (3 percent of G3) who completed the prediction and also filled out a questionnaire that popped up once the prediction was complete. In addition to demographic information, the questionnaire asked a question about the user's favorite team.

Step Three: Plotting Sequences

To further investigate G3a and G3b, we plotted the interaction sequences for G3a as an overlay on the interface similar to a heatmap (see Figure 6). The result is that user behaviors visually match the prediction space. There are a few scribbles outside the prediction space (such as over the buttons), but most of the interactions remain in the center.

Overall, the flow of dragging remains horizontal or leans toward the center of the interface. A few placeholders seem to not have been the destination of drags (such as the bottom ones for round of 16 and teams in the lower group stages F, G, and H). An interesting result is that the placeholder themselves are not crossed: users tended to release the team badges a bit before reaching the target placeholder.

We tested real-time replays as alternatives of showing the sequential nature of interactions (which is temporal), but those did not provide satisfying results. Replays are difficult to follow (even when using a fast-forward to reduce their duration), and because we did not record mouse drags, they do not show smooth transitions between recorded events.

Step Four: Sequence Abstraction

We decided to abstract interaction sequences and lay them out temporally. Figure 7 shows user interactions from G3b as lines. The horizontal axis is time (in seconds), and the vertical axis is the bracket's level of completion (from 0 to 32). This makes it possible to visualize the complete sessions and all interactions in each session. We observed three notable patterns.

First, the *ladder pattern* shows the cascading behavior of all similar interactions (in that case, moving teams to placeholders). Those interactions are regular and with no interruption. Indeed, interruptions would have been visible as horizontal lines where the user would not have performed any recorded interaction for a significant time (for example, if he stopped to read or drink a coffee). This confirms that the interface helps users focus on their tasks and that users sometimes do not need any external knowledge to complete it.

Second, the *reformulate pattern* occurred when a single team was dragged once and then again. This is interesting because it shows that users tend to rearrange predictions along the way. This confirms our early feedback from the first prototype where users required a step-by-step prediction process, rather than one that is too rigid once predictions are made.

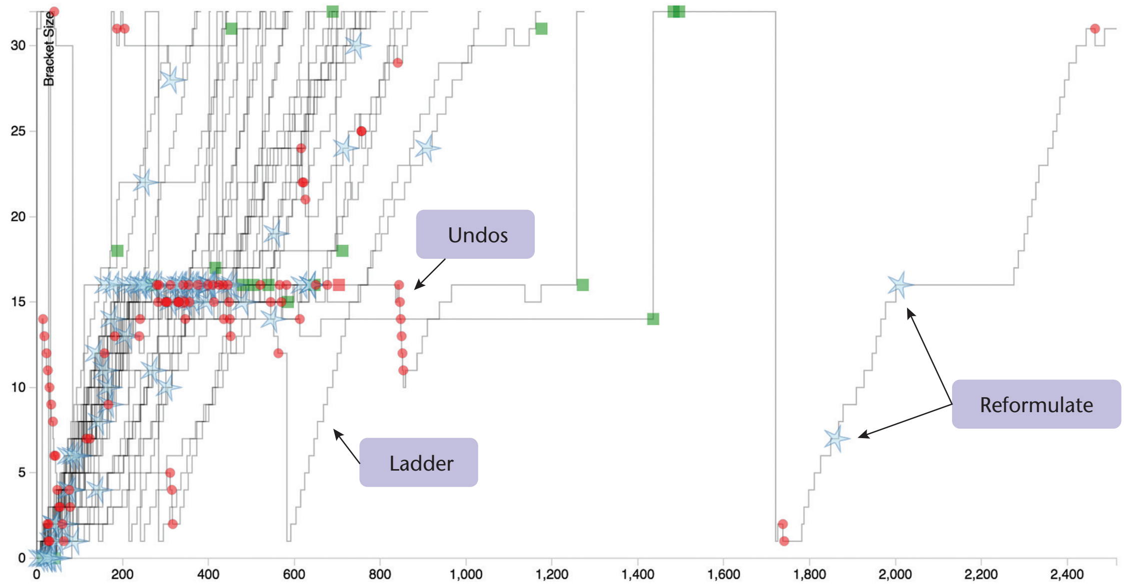

Figure 7. Sequence abstraction. Each line represents a visitor from G3b (who completed the prediction from scratch and filled out a questionnaire). The horizontal axis represents time, and the vertical axis shows the bracket's level of completion (from 0 to 32). Red dots represent clicks on the reset button for teams (not the reset for the whole prediction). Green rectangles represent clicks on the auto-complete button. Blue stars represent the interactions involving the visitor's self-declared favorite team.

Lastly, the *undo pattern* is another type of cascade, but it appears when the user deselects some teams to get back to a previous state of the prediction. This pattern seems to occur when the prediction becomes moderately filled. This confirms both the usability and the need to deselect individual teams, a feature we added in the second prototype.

These early findings paved the way for richer visualizations and interactions to dig into the wealth of log data and support them with data-driven evidence. Still, visual inspection of logs remains powerful because human behaviors are complex and difficult—if not impossible—to automatically query and retrieve such as by using SQL.

Qualitative Results

We received 65 responses to our questionnaire. We removed incomplete submissions and kept the 56 that were completed from G3b (1 female, with an average age of 22). Comments from the questionnaire were very supportive—for example, "I like the site and it's good to make a prediction" and "Awesome!!! Do this every year!" We also collected some more negative comments, such as "I really didn't find it useful. Am I missing something?" These negative comments can be explained by the fact that the interface probably requires more onboarding and interaction discovery, beyond the GIFs and tutorial we provided. Such a technique probably polarized users: either people understood how to use it, or they did not know how to start and got frustrated.

Looking at online forums where the technique had been advertised, we noted some domain-specific discussions and debates that were triggered by the tool. Posts sometimes including a link to a prediction that the person made, for example, "Just tried it out [link to prediction], and I feel if we get to win against Valencia…." Some other posts did not contain a specific link but were based on using the prediction interface: "I think you would be foolish to bet against PSV winning the whole thing," "I think we'll get 1st in the group, and with a good draw I think we can get to the quarters, and then anything can happen, I think Abdennour will play a huge part in any success," and "I pressed auto complete and Roma wins the champions league against Chelsea in the finals."

The feedback we collected from sport enthusiasts who used the interface validates the design of our interface based on the Drag-and-Snap technique for making tournament predictions. We observed predictions completed in full from scratch and users sharing them on forums. Based on both quantitative and qualitative data, we found that sport enthusiasts can effectively use a direct manipulation interface to make predictions and share insights. Among the data we collected, we only analyzed successful predictions; we discarded the unfinished ones. Exploring incomplete prediction data in more details could tell us about design issues that should be addressed to increase the rate of prediction completion.

In the future, we plan to investigate in particular the reasons for visitor drops, as these may be diverse and difficult to hypothesize. We also plan to investigate how other sports (such as basketball and baseball) currently support predictions and determine how applicable our technique and interfaces are in those domains.

In conclusion, our technique is a stepping stone toward making more use of direct manipulation for prediction interfaces. Our future work will focus on more deeply investigating user behaviors, particularly to automate the detection of patterns. The first challenge would be to extract the right features of user behavior to account for them in the detection model. This is nontrivial because interactions are highly contextual, as they have a meaning based on previous interactions. Another important challenge is to enhance the technique with those patterns, either with new interaction features or with the recommendation of predictions with automatic filling of the brackets that anticipates a behavior. We are confident those challenges can be achieved with collective input and collaboration and will lead to improved prediction techniques and interfaces for a broad range of domains.

References

1. B. Shneiderman, "Direct Manipulation: A Step beyond Programming Languages," *Computer*, vol. 16, no. 8, 1983, pp. 57–69.
2. M. Bostock and S. Carter, "512 Paths to the White House," *New York Times*, 2 Nov. 2012; www.nytimes.com/interactive/2012/11/02/us/politics/paths-to-the-white-house.html.
3. J. Boy et al., "Suggested Interactivity: Seeking Perceived Affordances for Information Visualization," *IEEE Trans. Visualization and Computer Graphics*, vol. 22, no. 1, 2016, pp. 639–648.
4. S.E. Hudson, "Adaptive Semantic Snaping—A Technique for Semantic Feedback at the Lexical Level," *Proc. SIGCHI Conf. Human Factors in Computing Systems (CHI)*, 1990, pp. 65–70.
5. P. Baudisch et al., "Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-Operated Systems," *Proc. Human-Computer Interaction (INTERACT)*, vol. 3, 2003, pp. 57–64.
6. G. Aisch, A. Cox, and K. Quealy, "You Draw It: How Family Income Predicts Children's College Chances," *New York Times*, 28 May 2015; http://nyti.ms/1BqOX3h.
7. B. Kondo and C.M. Collins, "DimpVis: Exploring Time-Varying Information Visualizations by Direct Manipulation," *IEEE Trans. Visualization and Computer Graphics*, vol. 20, no. 12, 2014, pp. 2003–2012.
8. R. Vuillemot and C. Perin, "Investigating the Direct Manipulation of Ranking Tables for Time Navigation," *Proc. 33rd Ann. ACM Conf. Human Factors in Computing Systems* (CHI), 2015, pp. 2703–2706.
9. C. Appert, O. Chapuis, and E. Pietriga, "Dwell-and-Spring: Undo for Direct Manipulation," *Proc. SIGCHI Conf. Human Factors in Computing Systems*, 2012, pp. 1957–1966.
10. P. Baudisch et al., "Phosphor: Explaining Transitions in the User Interface Using Afterglow Effects," *Proc. 19th Ann. ACM Symp. User Interface Software and Technology (UIST)*, 2006, pp. 169–178.
11. L. Wilkinson, *The Grammar of Graphics*, Springer, 2006.
12. C. Perin, R. Vuillemot, and J.-D. Fekete, "A Table!: Improving Temporal Navigation in Soccer Ranking Tables," *Proc. SIGCHI Conf. Human Factors in Computing Systems*, 2014, pp. 887–896.
13. E.T. Brown et al., "Finding Waldo: Learning about Users from Their Interactions," *IEEE Trans. Visualization and Computer Graphics*, vol. 20, no. 12, 2014, pp. 1663–1672.

Romain Vuillemot *is an assistant professor of computer science at Ecole Centrale de Lyon, Université de Lyon, France. His research interests include information visualization, visual analytics, and human-computer interactions, with a focus on advanced interaction techniques to include humans in the loop of complex predictive analytics processes. Vuillemot has a PhD in computer science from INSA de Lyon. Contact him at romain.vuillemot@gmail.com.*

Charles Perin *is a postdoctoral researcher in information visualization and human-computer interaction at the University of Calgary, Canada. His research interests include information visualization and human-computer interaction, particularly the design and study of new interactions for information visualization, understanding how people use and interact with visualizations in their everyday lives, and human-centered design for specific applications such as sports and medicine. Perin has a PhD in computer science from Inria and CNRS-LIMSI at Université Paris-Sud. Contact him at charles.perin@ucalgary.ca.*