Sport Tournament Predictions Using Direct Manipulation

Romain Vuillemot
Université de Lyon
romain.vuillemot@gmail.com

Charles Perin
University of Calgary
charles.perin@ucalgary.ca

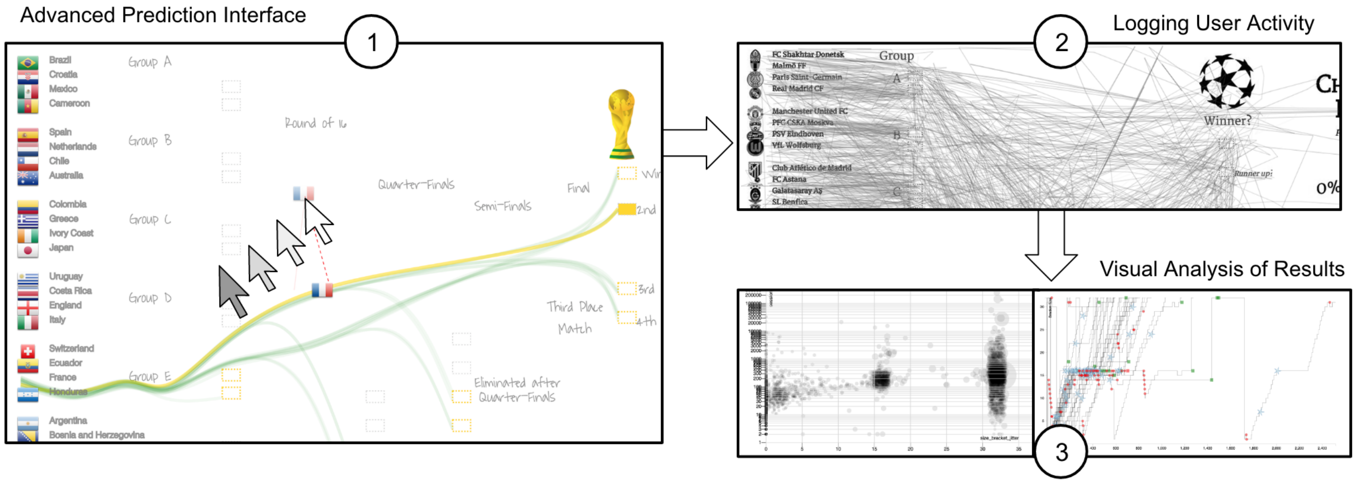

Figure 1. We present ① an advanced interface to facilitate users predicting soccer games with simple drag and drops of teams badges on the tournament bracket. We logged and plotted users activity ②, and performed a visual analysis of those log to identify users' behaviors ③.

ABSTRACT

We present an advanced interface for predicting sport tournaments by direct manipulation. The interface allows users to focus on their prediction tasks, by dragging and dropping teams to their final outcome in the competition, e. g., as winner or semi-finalist. This interface allows predictions to be made non-linearly, such as prediction winner first and filling up the other games. This better matches the way people actually make predictions, while the current interfaces can only be filled linearly, with text fields widgets as input. We released a first version of the interface for the 2014 FIFA soccer World Cup that validated the use of direct manipulation as alternative to widgets. We released an improved version a year later for the 2015/2016 UEFA soccer Champions League, which included a tracking system to understand users interactions. We recorded a total of $504,307$ interaction logs from $3,029$ unique visitors, among them 198 fully completed the prediction from scratch. From our analysis of logs, we provide a list of strategies that people employ to perform predictions, and which will help inform the design of further prediction interfaces that preserve the flow and concentration of users.

INTRODUCTION

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

Despite the strong interest for predictions and their relevance to soccer and many other sports, few interfaces exist that support this process. When such interfaces exist, they 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 a *big picture* first, and then refine it. For instance, one might want to predict who will win the World Cup *first*, and *then* detail each games outcomes to support this claim.

Improving the support for predictions is difficult because the mental process behind a prediction is usually heavy as it requires guessing unknown values from previous observations. It is also difficult since the guess has to comply with the constrained structure of a tournament. This structure is usually twofold: first a group phase where all the teams or subsets of teams play against each others either once or twice; and a bracket phase where teams can be eliminated against a specific opponent. Improving prediction will have impact beyond sport, such as to many domains such as weather forecast or economics projections.

This paper aims at improving the predictions creation process by users. For this purpose, we designed, implemented, and evaluated an interface that supports predictions of soccer tournaments. It implements the constraints of the tournament structure and makes it possible for people to predict–and adjust the prediction–for the big picture of a tournament outcomes in a single view, and its details. Our process outline follow the steps below, which are the key contributions of this paper:

1. We defined **a list of design criteria** to understand current needs from soccer tournaments predictions input interfaces and the state of the art in HCI, including direct manipulation [9].
2. 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 teams towards their final position in a single view.
3. We also designed and implemented **a set of novel logs visualization** to make sense of the $504,307$ recoreded interaction logs from the $3,029$ visitors in order to identify and discuss interesting behaviors.

Overall, we contribute evidence that sport enthusiastics *can* predict sport tournaments using an advanced direct manipulation interface that is non-linear. From our analysis of interaction logs, we provide a list of strategies that people employ to perform predictions. These strategies will help inform the design of prediction interfaces. Such technique could be applied to various fields involving predictions such as economics, finance, or weather forecasts that are currently run by automated models that could benefit from more user-generated data (e. g., parameters, customization of models, constraints).

CONTEXT

The context of our work is sport tournaments which are competitions involving multiple competitors, such as sport teams or individuals. We define a *prediction* in a broad way as the act of determining future, unknown values. We focus on the ones that are generated in a quantitative, user-generated way, without any model or automated assistance.

Sport Tournaments

We consider sport tournaments with a very common configuration: as the combination of a *ranking* phase and a *bracket* phase (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. Figure 2 (left) 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. Figure 2, right 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 of a schedule of games ahead (e. g., 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. Not some draws are not uniform, as they often account some constraints. For example, there can be country-specific constraints where teams from the same country cannot play against each other.

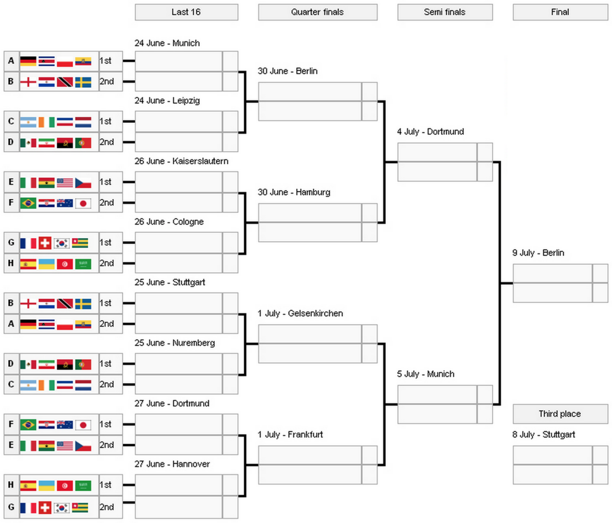

Figure 2. Example of a soccer tournament configuration. Left: group stage (column with teams badges) that results into a ranking after games ended. Right: the elimination bracket phase (for the best group stage teams) to decide who the winner of the competition will be.

Tournament Predictions

The general problem of *prediction* can be formulated as follows: it consists in determining a set of future values. A tournament *prediction* is the guess of all games outcomes, which indirectly decide of the tournament's winner. This decision in 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.

This is where there is a divergence between the current interfaces, and the way users perform predictions. Current interfaces only implement the above structure of predictions, as the sum of a series of games. While users' mental model follows the opposite path: it focuses on the tournament's outcome first, and then focuses on finding individual games 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 as it is the permutation of all possible games and outcomes. However, the more the tournament progresses, the smaller the prediction space is. Right before the final game, the prediction space is of size two, since all the games already happened but the last one, which has 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 for their predictions process.

Pre-Study

We identified 27 websites dedicated to bet and that support brackets entry in the area of sport online media. 15 websites were dedicated to soccer in particular and 12 to bets in general. We found that bet websites, are sometimes already populated with predictions suggestions (e. g., using bets odds), also allow users to enter their predictions. Figure 2 shows the typical interface for sport brackets (both for predictions and results communication) we observed. Most of the websites used standard HTML widgets as input widgets such as checkboxes, dropdown lists, and text inputs and organized on a bracket layout. To successfully enter their predictions, users have to manually input data 1) for each team by game, and 2) starting with the first round of games to end up with the final (from left to right on figure 2).

It is only looking at non sport-specific websites, such as general-purpose newspapers that we found innovating input interfaces. Bloomberg (`http://www.bloomberg.com/visual-data/world-cup/`) displays a soccer bracket to let users predict the games outcomes of the 2014 World Cup. FiveThirtyEight (`http://fivethirtyeight.com/interactives/world-cup/`) also uses brackets for the same event, but as a way to communicate their model's probability of outcome for each game (with no user interaction allowed). Finally, The New York Times article *512 Paths to the White House* (`http://nyti.ms/WhmfT7`) communicated predictions for the 2012 U.S. elections by letting users to interact with the prediction space (i. e. the 512 possible configurations of results) which can be filtered by various scenarios depending on state-level elections results.

Design Criteria

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

C1 Partial: Allow partial predictions, especially the ones about the favorite or best teams (e. g., winner, finalist).

C2 Levels of details: Allow to fill in the scores, or the number of goals scored, or only the outcomes of games.

C3 Non-linear: Let the user fill in the prediction from multiple entry points (e. g., semi-final, quarter-finals), 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 change of mind [9].

C5 Structure: Show the structure, such as the rules of the tournament, to let the user know the connection and dependencies between each round.

C6 Suggested interactions: Such as visual cues to tell the user what and how he can interact with graphical elements on the screen [4] (especially if widgets are not used any longer).

C7 Familiarity: Keep many aspects of the domain such as team badges and bracket layout to reduce the learning phase.

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

DIRECT MANIPULATION FOR PREDICTIONS

Informed by the previous design criteria, we introduce an interface to better assist sport enthusiastics in making sport tournament predictions. We focused on making it is easy to discover and learn, to let users focus on the mental process of predicting rather than being distracted by the input interface. The interface provides an overview (**C1**) of the **prediction space** (all possible results for teams) which makes visible the structure of the tournament (**C5**) and is used as a **suggested interaction** (**C6**) to inform users on what path a team can follow in the competition. Leveraging this overview, users can start with the winning team as well as with the semi-finalist or teams quickly eliminated, i. e. can perform a non-linear bracket filling (**C3**).

The technique in action is illustrated on figure 1 and figure 4. The main interaction is illustrated below on figure 3 and consists of three steps:

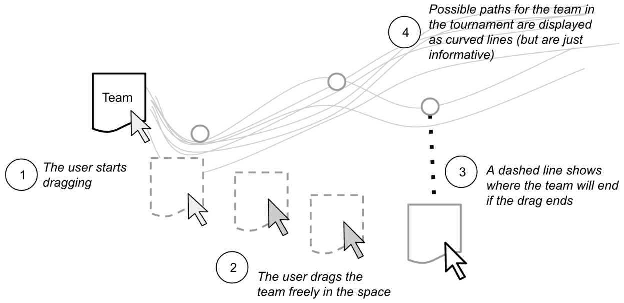
Possible paths for the team in the tournament are displayed as curved lines (but are just informative)
4
Team
1
The user starts dragging
3
A dashed line shows where the team will end if the drag ends
2
The user drags the team freely in the space

Figure 3. The main three steps to use DRAG-AND-SNAP and perform a prediction for a single team at a time.

Step 1: The user **starts dragging** a team by clicking and moving the mouse over the teams' badge.

Step 2: The user **keeps dragging** the team badge and a visual cue (a line) shows to which stage of the tournament the team can be dropped. The visual cue is green if the team can be dropped or is red otherwise.

Step 3: In order to make a prediction, the user releases the mouse button to **stop dragging** the team, which snaps it to the closest game it can be attached to.

This series of step constitute an extension of the well-known DRAG-AND-DROP that uses direct manipulation [9] of elements (called *objects of interest*) of the interface (in our case, soccer teams). We call this extension DRAG-AND-SNAP following [6] using teams possible paths as snapping constraints for teams, and visual cues for users to inform where teams can be dropped.

Related Direct Manipulation Techniques

This interaction relates to a body of similar techniques beyond [6] that aim at assisting users in space or time navigation when dragging elements.

DRAG-AND-POP [2] moves potential target icons towards the user's current cursor location, thereby allowing the user to interact with these icons using comparably small hand movements. DRAG-AND-PICK [2] makes all icons in the direction of the mouse motion come to the cursor. DRAG-AND-DRAW from The Upshot's article *You Draw It: How Family Income Predicts Children's College Chances* (`http://nyti.ms/1BqOX3h`) lets users freely draw a line on a chart on the prediction of how family income predicts children's college chances. The horizontal axis is the parent's income percentile (from poorest to richest) and the vertical one the percent of children who attend college. DRAG-AND-UPDATE [7, 10] lets users update data graphics by dragging items (e. g., countries) along their path over time. DWELL-AND-SPRING [1] uses the metaphor of springs to enable users to undo direct manipulations.

All these techniques use visual feedback that either connect elements or show additional information. The DRAG-AND-SNAP technique relates the most to DRAG-AND-POP as 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 (e. g., if the object can only move within a perimeter). From previous work, it is common to use *proxy* elements as a way of duplicating the object of interest. For example, DRAG-AND-POP and DRAG-AND-PICK duplicate icons to suggest them as target destination. We used a similar design in our technique in order to ensure that the team badge always follows the mouse (as a proxy) while the original badge remains on the bracket. This is visible on figure 1 ① where the flags are duplicated: one strictly follows the trajectory, the other one 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 predictions of the tournament in the background to suggest paths to follow during drag and drops. Using paths to connect elements has already been suggested to show previous interactions [3], as well as forthcoming ones. DRAG-AND-POP [2] uses an elastic rubber band to help users understand what it connects and a visual cue to convey how far away the target is. This is similar to [3], where the band also connects items using various path styles to simulate motions (e. g., motion blur or speed lines). All those designs are not suited to our case with thousands of trajectories, often packed on a small part of the screen. Thus, we use a visual design similar to a chess board's pieces trajectories as in Thinking Machine [11]. Each path is an arc with a jitter that makes it visually unique. This mostly aims at preventing clutter [12], but it also conveys the dynamics of pieces trajectories.

FIRST PROTOTYPE

We created a first interface implementing DRAG-AND-SNAP for the 2014 FIFA World Cup tournament (Figure 1, left). We released it one week before the start of the first game (June 12^{th}, 2014) to let enthusiastics predict the outcome of the tournament. The purpose of this release was to validate the design, detect any usability flaw and collect qualitative feedback for further improvement. The prototype and its source code are available at `https://github.com/romsson/worldcup14-interactive-bracket`.

The prototype follows the above description of the technique and some domain specific designs (**C7**). The teams participating to the tournament have an initial position on the leftmost part of the screen, before they enter the group stage. The user can do a prediction by drag and dropping a team 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 the drag then the team will be assigned to the game at the end of this path.

The interface also contains action buttons (See figure 4 for their location on the screen), including a reset button (**C4**) to start over and an auto-complete button to ease the complete filling. A bracket can be completed partially (**C1**) and starting at different levels (**C3**). As a complement to the background showing the prediction space, we displayed animated GIFs [8] that explain the drag and snap interaction (**C6**). Finally, we used team badges and the tournament logo for familiarity (**C7**).

This interface only allows 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 trade-off to accept 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 noticed the following issues:

- They found that snapping by closest trajectory was not intuitive and difficult to use, especially when there were multiple trajectories at the initial position, making it hard to select a specific one.

- 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.

- They needed to reset individual teams rather than the whole interface at once.

Focus	Type	Details

webpage	`init (auto)`
	`end (auto)`
		When the user opens the webpage
		When the user closes the webpage

teams badges	`hover`
	`drag`
	`drag (end)`
	`close`
		Mouse hover on a team badge
		Dragging just gets started
		Dragging just ende
		The user clicks on the close button

teams	`hover`	*Mouse hover on a team placeholder*

buttons	`click`
	`click`
	`click (submit)`
		Reset the prediction
		Auto-complete the prediction
		Share the whole prediction

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

SECOND PROTOTYPE

Based on the qualitative feedback we received for the first prototype, we designed and released an improved interface (Figure 4) for the 2015-2016 UEFA Champions League available at `https://github.com/romsson/ucl16-predictions`.

In this version, we improved the DRAG-AND-SNAP 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 to reset it to its initial position. We also changed the timing of animations and added question marks within empty placeholders and the use of modals to explain features. Those changes aim at making the technique ecologically valid (**C7**), e. g., as 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 [10] which uses D3 to handle mouse events. We used D3 for the representation of 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. As performance is key with direct manipulation interfaces, we generated an image containing all the trajectories and displayed it as background; we only highlighted user predictions using a SVG overlay (drawing each trajectory lines using SVG would have greatly impaired performance).

One last technical decision was to make the prototypes stateful, which means that after the user interacts with the system, his current prediction can be recorded and retrieved wit 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 to be used as a starting point by other people. In other words, this featured immediately allowed asynchronous collaborations by users around predictions.

EVALUATION METHODOLOGY

As our interfaces have been released *in-the-wild*, users interacted in a natural and confortable environment similar as when 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 task of prediction 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 (six months before this paper was submitted). 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. As a result, we focus on understanding *how* sport enthusiasts predicted rather than on assessing whether or not they make more 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 claim that a user's prediction can sometimes be progressively formed as he or she is using the interface, and thus cannot be known in advance.

To capture distant user activity, we used logging, a non-intrusing mechanism to capture in-the-wild users activity. This approach has been shown efficient to learn about users 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 moves as it would lead to too much data and would probably capture noise over signal.

VISUAL ANALYSIS OF LOGS

We designed and implemented a set of visualization tools to explore user logs. The goal of those tools is to assist us in better translating user interactions (e.g., click, 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 which duration was either under 1 second or over 45 minutes, and those with less than 5 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 size of a circle encodes the number of interactions of the visitor (max $4,354$

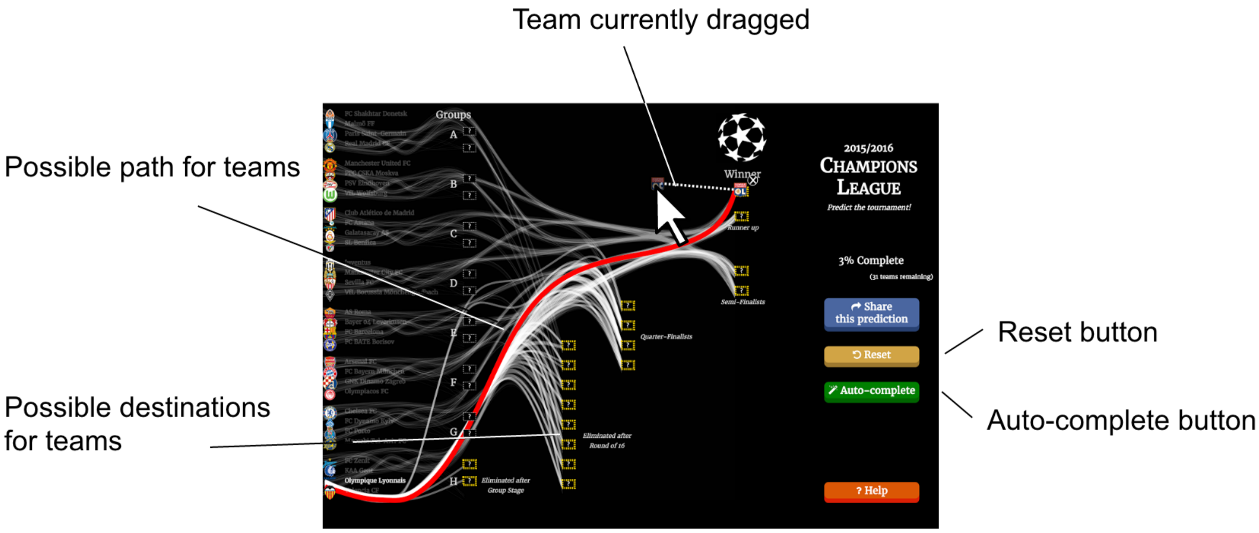

Figure 4. Screenshot of the second prototype's interface, with and emphasize on the background showing possible paths for teams, and action button such as a reset button to start over, and an auto-complete button to automate the bracket's filling.

interactions). The horizontal axis is the size of the bracket (32 is a full bracket), and the vertical axis is the time spent on the website (in second with a logarithmic scale). We added a low opacity to circles to better show the distribution of dense areas by reducing overplotting [12].

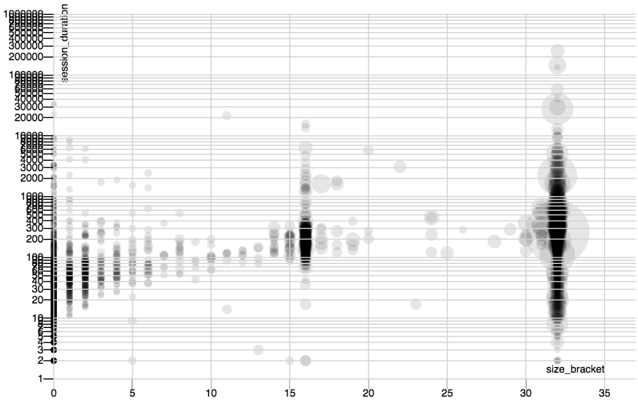

Figure 5. Scatterplot of visitors. Each circle represents a visitor. The size of a circle encodes the number of interactions of the visitor. The horizontal axis is the size of the final bracket (ranging from 0 to 32). The vertical axis is the time spent using the interface (in seconds, logarithmic scale).

Step Two: Grouping Visitors

We visually identified three main groups (**G0**, **G2**, **G3**) of visitors (as vertical stripes from left to right in Figure 5):

- **G0** contains 1,079 visitors (36% of total visitors) who did not fill the bracket (the vertical line of circles with zero prediction in Figure 5). Those visitors left the interface unchanged with no interaction and with an empty bracket.
- **G2** contains 287 visitors (9% of 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% of 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 prediction themselves. Indeed, since the bracket can be shared, lots of visitors may have started their prediction with an already completed bracket. We found that in **G3**, 11.8% of the visitors (198 visitors) started from scratch and we call this group **G3a**. We identified a sub-group of **G3a** called **G3b** of 56 visitors (3% of **G3**) who additionally to completing the prediction, also filled a questionnaire that popped up once the prediction was complete. Among many demographics information, the questionnaire also contained a question on the user's favorite team.

Step Three: Plotting Sequences

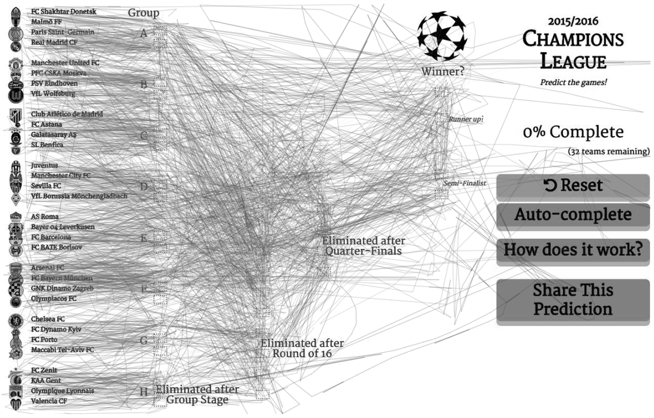

Figure 6. Aggregation of of dragging interactions, where each line connects all the drag and drop of a team, by a visitor.

To further investigate **G3a** and **G3b** we plotted the sequences of interactions of **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 scribble outside the prediction space (e. g., over the buttons) but most of the interactions remain in the center. Overall, the flow of dragging remains horizontal or leaning towards the center of the interface. 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, H). An interesting result is that the placeholder themselves are not crossed: users tend to release the teams a bit before reaching the target placeholder. We tested real-time replay 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 with a fast-forward to reduce their duration) and since we did not record mouse drag, it does not show smooth transitions between recorded events.

Step Four: Sequences Abstraction

We decided to abstract sequences of interactions and to 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 level of completion of the bracket (from 0 to 32). This makes it possible to visualize the complete sessions and all interactions in each session. We observed the following notable patterns:

The first one is a LADDER pattern which 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 ave performed any interaction we record for a significant time (e. g., he might read or drink a coffee). This confirms that the interface helps users focus on their task and that users sometimes do not need any external knowledge to complete it.

The second pattern is a RE-FORMULATE pattern, which occured when a single team had been dragged once, and then dragged again. It is interesting as it shows users tend to rearrange a current prediction 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 were too rigid once predictions were made.

The last pattern is an UNDOS pattern. This pattern is another type of cascade but appearing when the user de-selects 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 for being able to un-select individual teams that we added in the second iteration of the technique.

Those early findings pave 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 as human behaviors are complex and difficult – if not impossible – to automatically query and retrieve such as 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, average age 22). Comments from the questionnaire were very supportive, e.g., *"I like the site and its good to make a prediction."*, and *"Awesome!!! Do this every year!*. We also collected some more negative comments, e.g., *I really didn't find it useful. Am I missing something?* These negative comments can be explained by the fact that the interface would probably require more onboarding and interaction discovery, beyond the GIFs and tutorial we provided. Such technique probably polarized users: either people got 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, e.g., *"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, e.g., *"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"*, *"I pressed auto complete and roma wins the champions league against Chelsea in the finals."*

DISCUSSION AND CONCLUSION

The feedback we collected from sport enthusiasts who used the interface validates the design of our DRAG-AND-SNAP-based interface for making tournament predictions. We observed fully completed predictions from scratch and their sharing on forums. Based on both quantitative and qualitative data, we found that sport enthusiasts can effectively use a direct manipulation interface to perform predictions and share insights. Among the data we collected, we only analyzed successful predictions: we have discarded the unfinished ones. Exploring incomplete predictions data in more details would inform us on design issues that should be addressed to increase the rate of predictions completion.

In the future, we plan to investigate in particular the reasons for visitors drop, as these may be diverse and difficult to hypothesize. We also plan to investigate how other sports (Basketball, Baseball, ..) currently support predictions, and how applicable are our technique and interfaces to those.

In conclusion, our technique is a stepping-stone towards making more use of direct manipulation for prediction interfaces. Our future work relies in the deeper investigation of user behaviors, particularly to automate the detection of patterns. The first challenge would be to extract the right features of user's behavior to account into the detection model. This is non-trivial since 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 by anticipation of a behavior. We are confident those challenges can be achieved with collective inputs and collaboration, and

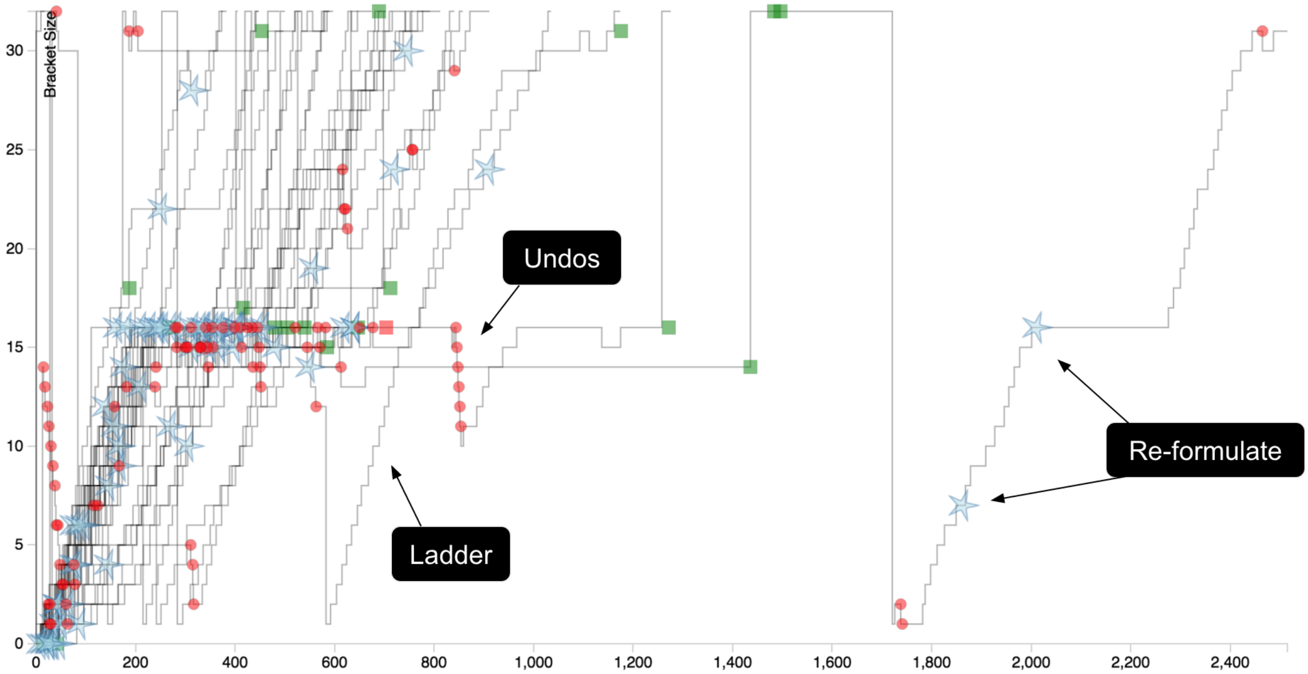

Figure 7. Each line represents a visitor from G3b (who completed the prediction from scratch and filled a questionnaire). The horizontal axis represents time and the vertical axis the level of completion of the bracket (from 0 to 32). Red dots represent clicks on the reset button of teams (not the one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

will lead to improved prediction techniques and interfaces for a broad range of domains

REFERENCES

1. Appert, C., Chapuis, O., and Pietriga, E. Dwell-and-spring: undo for direct manipulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2012), 1957–1966.
2. Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P., Bederson, B., Zierlinger, A., et al. Drag-and-pop and drag-and-pick: Techniques for accessing remote screen content on touch-and pen-operated systems. In *Proceedings of INTERACT*, vol. 3 (2003), 57–64.
3. Baudisch, P., Tan, D., Collomb, M., Robbins, D., Hinckley, K., Agrawala, M., Zhao, S., and Ramos, G. Phosphor: Explaining transitions in the user interface. In *In Proc. of UIST'06* (2006).
4. Boy, J., Eveillard, L., Detienne, F., and Fekete, J.-D. Suggested interactivity: Seeking perceived affordances for information visualization.
5. Brown, E. T., Ottley, A., Zhao, H., Lin, Q., Souvenir, R., Endert, A., and Chang, R. Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics 20*, 12 (2014), 1663–1672.
6. Hudson, S. E. Adaptive semantic snaping&mdash;a technique for semantic feedback at the lexical level. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, ACM (New York, NY, USA, 1990), 65–70.
7. Kondo, B., and Collins, C. M. Dimpvis: Exploring time-varying information visualizations by direct manipulation. *Visualization and Computer Graphics, IEEE Transactions on 20*, 12 (2014), 2003–2012.
8. Perin, C., Vuillemot, R., and Fekete, J.-D. À table!: Improving temporal navigation in soccer ranking tables. In *Proc. CHI '14*, ACM (2014), 887–896.
9. Shneiderman, B. Direct manipulation: A step beyond programming languages. *Computer 16*, 8 (1983), 57–69.
10. Vuillemot, R., and Perin, C. Investigating the direct manipulation of ranking tables for time navigation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, ACM (New York, NY, USA, 2015), 2703–2706.
11. Wattenberg, M. Thinking machine, 2003.
12. Wilkinson, L., Wills, D., Rope, D., Norton, A., and Dubbs, R. *The grammar of graphics*. Springer Science & Business Media, 2006.