1 Software

We used EventFlow[1] to analyze event temporal data. Thanks to Megan Monroe we received the URL to download version 2.023 of the software. We are mainly interested in EventFlow because we already have some experience with simpler data formats and are interested to explore temporal data. However, while we learned a lot about temporal data visualization, finding interesting insights and visualizations or even finding a suitable dataset for the task proved to be challenging; in comparison, it is relatively easy to use simple tabular datasets and visualization with standard charts and scatter plots.

2 Data Source

We had three initial candidate datasets to try. One was the airlines dataset which records flights for each airline over a period of time. The other dataset was basketball game events; this dataset contains exact timed events associated with a player. We ultimately decided to study a soccer event log data source from English Premier League, because it captured more events of the game. The dataset is a professional soccer commentaries dataset. The dataset was originally acquired by Disney research group at a considerable cost. This dataset contains timed events associated with players, e.g., passes between players. It contains 19,216 records with 16 attributes. We investigated different aspects of the game and did not just concentrate on the goals as they are infrequent within a game. To give an idea of the event types present in the data, we show the frequencies of the ten most common event types in Figure 1.

To make the soccer data compatible with EventFlow, we had to make several transformations. The data for soccer events is available in a file format where every row is a collection of attributes. For example, one row of the data is type:pass outcome:successful secs:1 pass events:head pass team_id:arsenal player_id:marouane chamaikh body part:non. After converting this data into a tabular format, we then joined the table of events with a file provided with the data containing the positions of all the players from all the teams.

The next step in our preparation process was critical in determining what analysis was possible in EventFlow; that was the decision of extracting EventFlow records from the soccer events. In EventFlow, a record is an identity, e.g., a patient in a drug trial, that is tracked over time and which experiences either instant or interval events. We chose to analyze the data using three different types of records: games, teams and players. Game records tracked the events within an entire game, team records tracked the events for a team within a game, and player records tracked the events for a player within a team within a game. We found that the different record types led to different types of insights from EventFlow. The final step in our data preparation process was to generate record-level and event-level input files in the text file formats supported by EventFlow. We implemented the entire data preparation process in Python using the Pandas library.

1 The software's website is http://www.cs.umd.edu/hcil/EventFlow/
2 available at http://moa.cms.waikato.ac.nz/datasets/
3 available at http://www.basketballgeek.com/data/
Figure 1: The most common event types in the data.

3 Interesting Observations

Initially we looked for insights such as comparative statistics about each team’s strategy, e.g., average time before losing the ball or making a shot; and commons patterns in the plays, e.g., a pass pattern between certain players. However, we found that, once the data was loaded properly, a number of unexpected insights were revealed. Below are some observations:

1. In Figure 2 we see a view of each of the games played by each team. In EventFlow it is common to align its aggregate view on a type of event, but we do not use it here. Because the view is not aligned on any event type, we have a view of the entire game. One way that this view is effective is at finding long sequences of the same type of event, and in this case we see a long sequence of 29 passes in a row by a team in a game. A spectator may be eager for some other type of action after such a long sequence of passes!

2. Next, in Figure 3, we see the actions taken by players in different positions both before and after a corner is awarded. Being awarded a corner may be a very good opportunity for a team to score because the play may start near the opposing team’s goal and the opposing team may have very little time to react to the team’s strategy. In this visualization we observe several things that may not be apparent from watching the game. The most common action before a corner awarded for both midfielders and forwards is an attempt saved (at a short median time of three seconds before the corner awarded); meanwhile the most common action for defenders is a clearance. Defenders and midfielders have many more corners awarded than forwards. Finally and not surprisingly, all actions after a corner is awarded happen after a delay due to the pause in play.

3. In Figure 4 we now align events by fouls and see the events by all players that either precede or follow the foul. Something interesting we can see is the similarity of the distributions of events that precede a foul. The most common is a ball recovery and the second most common is an out. So, it may be surprising to see a substantial change in the frequencies of actions from the first to the second halves, but that is what we see with ball touches, which increase in frequency, and aerials, which become relatively less frequent. Also cards often follow fouls regardless of what half they occur in.

4. In Figure 5 we use data where each record is an entire half of a game played by a single team, and we align on all goals made. Interestingly, the team Arsenal most often precedes a goal with a ball recovery while most other teams have goals following fouls (resulting from the foul, perhaps). Two other teams make goals following outs. Arsenal, however, then proceeds to make fouls after a goal while other teams do not.

5. The dataset was from a paper that was recently published at the reputable UAI\textsuperscript{5} conference [2]. We found the data contained twelve events named ”error” in the datasets. A fact that was not caught by the authors of the paper.

\textsuperscript{5}Uncertainty in Artificial Intelligence
Figure 2: A long sequence of passes by a team within a game. A spectator may find such a long sequence boring!

6. By using the event query tool in EventFlow we searched for players who made consecutive fouls without doing anything else, and we were able to identify one player that acted with such poor behaviour.

7. We also wondered if it was the case that most goals in soccer are made by players who position themselves near the opponent’s goal and wait for the ball to get to them, or if most goals are made by players who participate in the offensive teamwork. By using the aggregate view of EventFlow we found that 65% of the players who scored a goal, on average made a pass within the last 42 seconds of the goal. From that we could infer that they participated in the offensive team play.

8. We also noticed that about 15% (4 out of 27) of goals in the English Premier League were scored at the very beginning of the game and those players had at most 2 passes.

There are many random patterns in the soccer games and finding a definite observation is not straightforward. However, from the above experiments with the data we can suggest the following headlines:

1. Arsenal scores most of its goals after a ball recovery and perhaps on counter attacks.

2. Arsenal makes more fouls after they score a goal comparing to the other teams.

3. 15% of the goals in the English Premier League were scored at the very beginning of the game.
Figure 3: Actions of players in different positions before and after a corner awarded. The most common action before a corner awarded for both midfielders and forwards is an attempt saved; meanwhile the most common action for defenders is a clearance.

4 Tool Critique

We were excited to work with EventFlow as it provides a novel visualization of temporal sequences and patterns. It is obvious that, without such a tool, seeing and capturing many of these events would not be possible; even simple facts like discovering the errors in the dataset could be difficult. We think the variety of tools provided with EventFlow for manipulating, grouping and querying the data are effective. On several occasions we wanted a particular feature and later discovered that EventFlow had it; grouping by attributes is an example of this and turned out to be very useful.

We tested EventFlow on Windows, Mac-OS and Ubuntu. We encountered some issues that were platform specific and some others that may have been due to our limited expertise with the tool. We observed the following.

1. When loading data on Ubuntu, a small empty window opens on the top left side of the screen (which was not easy to find) and everything halts. Figure 6 shows the window re-sized to be seen.

2. F2 key is used to freeze the attribute window. However, on our systems we couldn’t simply press F2 to freeze the attribute window; instead we had to hold the F2 key for a second and then click on the event and release the F2 key.

3. The soccer dataset had many events and we were interested in only a few of them most of the time. We didn’t find a way to deselect all event types at once and select only a few, instead we had to go and deselect them one by one each time.

4. They were many ”pass” events in our dataset. When we selected ”Align by all passes”, EventFlow started working and then halted, so we had to kill the EventFlow process. For very frequent event types there is no warning or ability to cancel the actions that cause this.

5. In aggregate view, the horizontal zoom level can change unexpectedly, e.g., when applying grouping.
Figure 4: Actions of players in either the first or second half surrounding a foul. Ball touches increase in frequency between the first and second halves while aerials become relatively less frequent.

Figure 5: Actions of teams before and after a goal. Arsenal most often precedes a goal with a ball recovery while most other teams have goals following fouls, perhaps resulting from the foul.
6. Event lists are initially sorted based on frequency. This makes finding a specific event in the legend time consuming, i.e., linear with the number of event types. Having the lists be sorted alphabetically would be a valuable option. Also, in aggregate categories, and event search menu, event lists are also not alphabetized where in that view there is no real need for seeing them based on frequency.

7. It would have been helpful if some of the pre-processing data manipulation that we performed outside of EventFlow could have been done inside EventFlow. For example, extracting multiple records based on players, or based on different teams, from a record that contains all information about one game.

References
