chess_web_page Documentation

Release 0

Juan I. Perotti

August 12, 2016

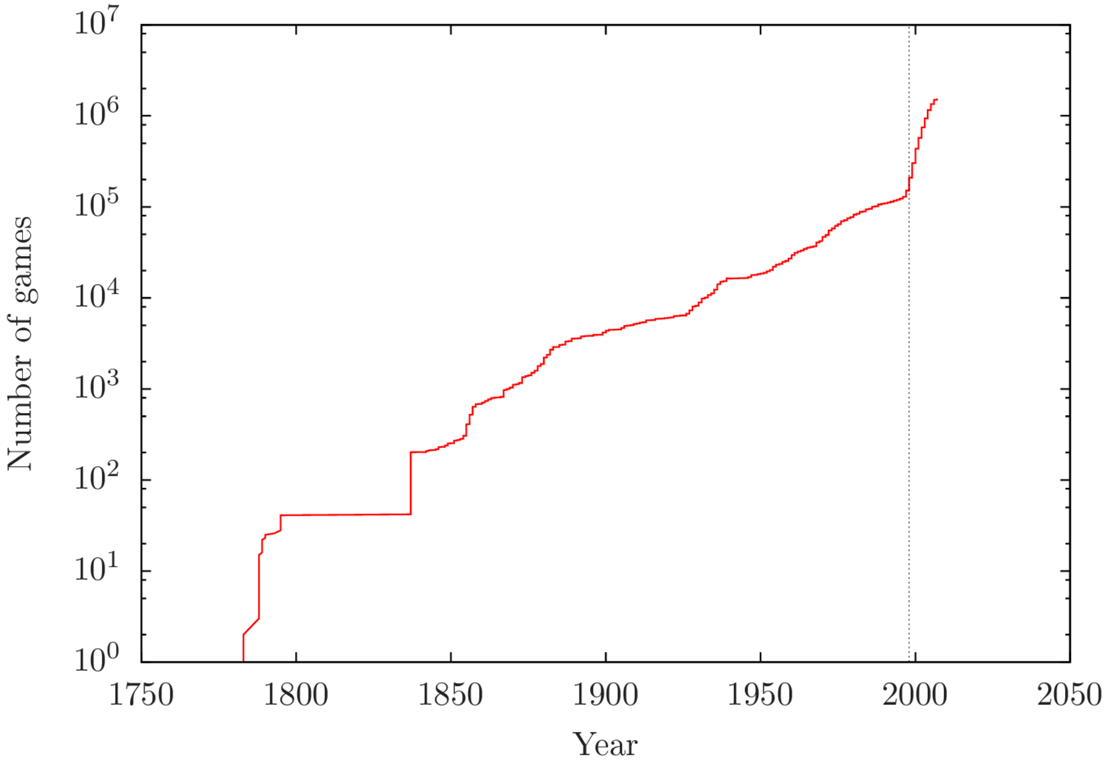
Original dataset

Here, we provide the chess dataset in two convenient formats (see below). However, if you are interested, you can download the original dataset from [1]. The original dataset is stored in an efficient (in terms of space and access speed) but weird format that cannot be easily transformed into PGN (Portable Game Notation) format without using some specific program. If you want to work with the original data, anyway, I recommend you to install the chessDB software [1] that allows you to download the original data and then transform it to PGN format.

[1] http://chessdb.sourceforge.net/

Description of the dataset and data curation

In the original dataset there are around **3.5 million games**. If you study the cumulative number of games played as a function of time (following plot)



you can see that the oldest game in the database is from the year 1783 (a blindfold by Philidor). After that, several periods can be appreciated, roughly. The first period consisting in old games where the data is very sparse. The second period, starting at the end of 1800, is when chess became a popular game. A third period occurs around 1960 during the cold war, when chess became a serious matter; i.e. it became a highly competitive and profesional sport. Finally, a fourth period starts around 1998 with the masification of the Internet. This last period contains most of the games and is the most consistent one. After filtered out games without a complete date (day,month,year), around **1.4 million**

games remains. This is the dataset we used for the calculations in our papers [2] [3] [4]. A similar dataset has been used in [5].

For more information about the dataset, please read the papers [2] [3] [4].

[2] Memory Kernel in the Expertise of Chess Players, A.L. Schaigorodsky, J.I. Perotti, O.V. Billoni, submitted (2015) arXiv:1504.06611

[3] Memory and long range correlations in chess games, A.L. Schaigorodsky, J.I. Perotti, O.V. Billoni, Phys. A 394, 304-311 (2013) arXiv:1307.0729

[4] Innovation and Nested Preferential Growth in Chess Playing Behavior, J.I. Perotti, H.-H. Jo, A.L. Schaigorodsky, O.V. Billoni, Europhys. Lett. 104, 48005 (2013) arXiv:1309.0336

[5] Bernd Blasius and Ralf Tonjes. Zipfs law in the popularity distribution of chess openings. Phys. Rev. Lett., 103, 21, 218701 (2009) APS.

CHAPTER 3

Download the dataset

We provide the following files, each of which contains all and the same information but in different formats,

All games in the original dataset transformed to .PGN format

all.pgn.zip [831 MB]

All games in a **simplified format** (created by us) with labels for easy filtering of games according to their attributes

all_with_filtered_annotations.txt.zip [747 MB]

Description of the simplified format

We converted the data in PGN format to a **simplified format** which we find convenient to work with. This format can be easily handled with a combination of different linux bash commands. Below we provide a few examples on how to do this but, first, let us describe the **simplified format**.

In the file `all_with_filtered_annotations.txt`, lines starting with the character # correspond to comments or the description of the columns. For example:

the first 8 first lines of this file looks like. The last 3 lines correspond to the first 3 games in the file:

```
# #
# datetime 2013-08-10 22:32:16.640552
# program programs/formats/pgn_to_filtered_very_basic_format_plus_info_filtering_info.py
# filein original_data/scidbase/all.pgn
# 1.t 2.date 3.result 4.welo 5.belo 6.len 7.date_c 8.resu_c 9.welo_c 10.belo_c 11.edate_c 12.setup 1
1 2000.03.14 1-0 2851 None 67 date_false result_false welo_false belo_true edate_true setup_false fe
2 2000.03.14 1-0 2851 None 53 date_false result_false welo_false belo_true edate_true setup_false fe
3 1999.11.20 1-0 2851 None 57 date_false result_false welo_false belo_true edate_false setup_false fe
```

After the comments and description of the columns, each line corresponds to one and only one game. The first columns describe attributes of the game, such as the date in which it was played, the name of the players, etc. The last columns, from 17 onwards after the token ###, contains the sequence of the game moves. Let us provide a description for the columns of the game attributes:

1. Position of the game in the original PGN file.
2. Date at which the game was played (the format is `year.month.day`).
3. Game result specified inside brackets in the PGN file. The value can be `1`, `0` or `-1` corresponding to white win, draw or loose, respectively.
4. ELO of withe player (an integer number).
5. ELO of black player (an integer number).
6. Number of moves in the game (for some games it may be zero!)
7. `date_c` = date (in `year.month.day`) is corrupted or missing? the label should be `date_true`, meaning the date is corrupted, or `date_false`, meaning the date is NOT corrupted. The same logic applies to the following attributes ending in "_c" (i.e. _corrupted).
8. `resu_c` = result (`1-0`, `1/2-1/2`, or `0-1`) is corrupted or missing?
9. `welo_c` = withe ELO is corrupted or missing?
10. `belo_c` = black ELO is corrupted or missing?
11. `edate_c` = event date is corrupted or missing? The event where the game was held (if there is one).

12. `setup` may be `setup_true` or `setup_false`. If it is true then the game initial position is specified. This is used when playing Fischer Random Chess for example.
13. `fen` may be `fen_true` and `fen_false`. It is related to column 12.
14. In the original file the result is provided in two places. At the end of each sequence of moves and in the attributes part. This flag indicates if the result is (is not) properly provided after the sequence of moves (just for checking consistency in the PGN file).
15. `oyrange` may be `oyrange_true` or `oyrange_false`. This flag is false only for games with dates in the range of years [1998,2007]. The `oyrange` means `out of year range`.
16. `bad_len` (or bad len) flag indicates, when `blen_true` (`blen_false`), if the length of the game is (is not) good.
17. Finally, after the token `###`, you can find the sequence of moves. Each move has a number and a letter `W` (white) or `B` (black) indicating the th-move of the white or black player, respectively.

The sequence of moves for one game (the first game in the dataset, in this case) looks as follows:

```
W1.d4 B1.d5 W2.c4 B2.e6 W3.Nc3 B3.Nf6 W4.cxd5 B4.exd5 W5.Bg5 B5.Be7 W6.e3 B6.Ne4 W7.Bxe7 B7.Nxc3 W8.
```

This is the string that should be parsed, e.g. using Python, to process the first game on the **simplified format** of the database.

IMPORTANT: In the file `all_with_filtered_annotations.txt` the games are NOT in chronological order because some games have missing date in the original database.

CHAPTER 5

Easy filtering of undesired games and sorting by date: examples using bash commands

You can use a simple linux(bash) command to quickly filter games with undesired properties from this file. For instance, the command:

```
cat all_with_filtered_annotations.txt | grep 'welo_false' > all_with_filtered_annotations_with_wELO.t
```

will generate the file “all_with_filtered_annotations_with_wELO.txt” with games where the white ELO is provided. Of course you can concatenate the command `greep` to filter more than one property. For instance:

```
cat all_with_filtered_annotations.txt | grep 'welo_false' | grep 'belo_false' > all_with_filtered_ann
```

will kept the games with both, white and black ELO specified. Another example,:

```
cat all_with_filtered_anotations.txt | grep 'oyrange_false' > all_with_filtered_annotations_since1998
```

will generate a file with games after the year 1998.

If you want to sort games by date, first generate a file filtering games with corrupted date by:

```
cat all_with_filtered_anotations.txt | grep 'date_false' > all_with_filtered_annotations_with_dates.t
```

and then use the following linux command to sort the games by date:

```
sort -k2 all_with_filtered_annotations_with_dates.txt > all_with_filtered_annotations_sorted_by_date.
```

CHAPTER 6

References