**3**

This chapter describes how to solve a general problem of extracting knowledge out of natural language sources and defines which steps have to be considered. Therefore, in the following the focus is set on universal approaches to accomplish such a task. Specific ideas and solutions adjusted to the sentiment classification task for chess annotations will be dealt with in chapter 4.

According to estimates, around 80 % (SOURCE) of all information in the internet exist in the form of natural language. Usually, it is hard to evaluate the information contained in this data, because they are not structured in the same way. For example, in product reviews every customer can write his comment in a different kind, so that there is neither a certain order of the information nor a specification, which information the comment should provide. However, on the basis of an additional star rating it is possible to get a fast assessment of the customer’s attitude towards the product. So, if there is a need of further evaluation, it is helpful to have the data in a structured form instead of an unstructured form.

Often it is not desirable or even impossible to get the data already existing in an unstructured form also in a structured form. 🡪 class

Picture processing pipeline of information extraction

In the first step we have to define the input and the output of the classification task. A possible input of natural language could be represented by a whole book, by a (web) page, by a paragraph or just by one sentence. In certain cases, even a single letter is an appropriate input, e.g. for the detection of handwritten letters.

As well as for the input we need to determine the type of output we want to receive. But not only the type, also the precision in the range of values is important for the difficulty of the task. In the case of product reviews already mentioned above the easiest output “good review vs. bad review” could be complicated by using the ten values of a five star rating or by distinguishing between different ratings for the quality, the price-performance ratio, the delivery etc.

**4**

In the context of this work we consider the general problem described in the previous chapter this time in the field of chess annotations.

As data sources a set of files (<http://www.angelfire.com/games3/smartbridge/>) in standard PGN format is used as well as a bundle of commented games that have been extracted from Mega Database 2012 (SOURCE) in ChessBase format. The related user interface offers the possibility to select the desired games and convert them into the standard PGN format.

However, it is not possible to filter the games by the used comment language. For this reason, an additional language detection polyglot (SOURCE) is used to reduce the comments that will be processed to the English ones.

Analyzing a commented file

PGN

PGN is "Portable Game Notation", a standard designed for the representation of chess game data using ASCII text files. PGN is structured for easy reading and writing by human users and for easy parsing and generation by computer programs (<http://www.saremba.de/chessgml/standards/pgn/pgn-complete.htm>, chapter 1). A sample game in PGN notation is shown in figure TODO.

A PGN game contains first a list of tuples with general information of the game (“tag pairs”). Seven of those tags are mandatory (Seven Tag Roster: Event, Site, Date, Round, White, Black, Result), the other tags are optional.

Afterwards the “movetext” section starts. The chess moves themselves are represented using SAN (Standard Algebraic Notation). A move pair (moves of white and black) starts with the move pair number followed by a dot and a blank, then the move of white, another blank and the move of black, e.g. 7. Bg5 a6. Each move contains the piece by a single upper-case letter (see table TODO) followed by the square the piece is moved to (see figure TODO). Hence, the example describes the seventh move of both players in the game; white moves his dark-squared bishop to the square g5 and black moves his a-file-pawn to a6. If a move pair is interrupted by a comment, the move of black is prefaced by the move pair number, an ellipsis and a blank: 27. Nxf4 $2 {doesn't work because of} 27... exf2+ (see chapter 8 of TODO).

Parts of the moves are annotated using comments in braces. A comment can contain information about the opening of the game, about a single move or about the current position. In the last two cases the comment is often prefaced by one or several NAG (see below) or the corresponding chess symbol. Since there is no restriction on the exact position of a comment, comments may refer to the move before or after itself. A comment can also connect two or more moves with each other. On the contrary, a comment can be interrupted by a move such that it is split into two parts, which may only make sense when seen together. All in all, there are four possibilities of comment-move combinations shown in the examples below (TODO).

Besides, by convention there should not be nested braces, however, sometimes nested braces are used to comment different move variants separately.

NAG (see chapter 10 of TODO)

Token definition by PGN

As we have already seen in chapter TODO, there are different comments in a PGN file.

Since a supervised learning approach is used, we need to know the correct class of a comment in the file. Therefore, the comments which are from importance are those connected to a traditional chess symbol or a NAG.