Sentiment classification of chess annotations

Master-Thesis von Florian Beck
1. Gutachten: Prof. Dr. Johannes Fürnkranz
2. Gutachten:

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Informatik
Knowledge Engineering Group

Sentiment classification of chess annotations

Vorgelegte Master-Thesis von Florian Beck

1. Gutachten: Prof. Dr. Johannes Fürnkranz
2. Gutachten:

Tag der Einreichung:

Erklärung zur Master-Thesis gemäß § 22 Abs. 7 und § 23 Abs. 7 APB TU Darmstadt

Hiermit versichere ich, Florian Beck, die vorliegende Master-Thesis gemäß § 22 Abs. 7 APB der TU Darmstadt ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß § 23 Abs. 7 APB überein.

Darmstadt, den March 29, 2019

(Florian Beck)

Abstract

Dies ist der Abstract der Arbeit. Er gibt wertungsfrei, kurz und prägnant den Inhalt der wissenschaftlichen Arbeit wieder.

Contents

1 Introduction **5**
- 1.1 Motivation . . . 5
- 1.2 Problem description . . . 5
- 1.3 Goal of the thesis . . . 5
- 1.4 Structure of the thesis . . . 5

2 Basics **6**

3 Concept **7**

4 Approach **8**
- 4.1 Data set extraction . . . 8
  - 4.1.1 PGN-format . . . 8
  - 4.1.2 Chessbase-DB . . . 9
  - 4.1.3 NAGs . . . 9
  - 4.1.4 Python NLTK . . . 10
- 4.2 Prepare data for classification . . . 10
  - 4.2.1 Feature extraction . . . 10
  - 4.2.2 Generating training instances . . . 10

5 Experiment setup **11**
- 5.1 Selecting classes . . . 11
- 5.2 Tokenizer tuning . . . 11
- 5.3 NLTK Parameters . . . 11
- 5.4 Classifiers . . . 11

6 Evaluation of results **12**

7 Conclusion **13**
- 7.1 Summary . . . 13
- 7.2 Outlook . . . 13

List of Figures

1 Sample PGN game 8
2 Square names in algebraic notation[1] 9
3 Sample arff file 10
4 Sample sparse arff file 10

List of Tables

1 Basic chess notations 9
2 Comment-move combinations 9
3 Meaning of NAGs 10

1 Introduction

1.1 Motivation

1.2 Problem description

1.3 Goal of the thesis

- is it possible to "convert" a chess annotation comment to the appropiate symbol by using a classifier

1.4 Structure of the thesis

bla

2 Basics

Sentiment Analysis and classification, Ordinal Classification, Word Embeddings, TF-IDF mindestens 5 Seiten

3 Concept

- general problem: structured data easier to evaluate than unstructured data, possibility to build statistics (how good is a product rated?, political attitude?, player/game statistics in chess -> average count of good/bad moves in a chess game, average count of good/bad moves by a specific player (-> talent scouting?)) –> EXTRACT KNOWLEDGE OUT OF UNSTRUCTURED INFORMATION - general [in this case]: comments [chess annotations] should be converted to symbols (=classes) [chess symbols] - step 1: define input and output - possible input types (symbol: detection of handwritten letters, word/sentence: detection of language, page/file: detection of author - possible output types (letter, language, author) - "precision": language = language family|language|dialect -> chess: good move vs. brilliant/good/slightly good move - step 2: find a database (or similar source) with sufficient information to extract data from - step 3: define conditions that filtered data sets has to fulfill - language restriction - minimal comment length - supervised/unsupervised learning - in case of

- step 5: tokenize the text - handling of punctuation - step 6: token preprocessing - remove stopwords - lowercase - stemming
- how to handle order of classes?
- how to handle differences in class counts?
- how to handle too many attributes? -> attribute selection

4 Approach

4.1 Data set extraction

Nur Weka-Classification oder auch ersten Ansatz der NLTK-Classification mit reinnehmen?

4.1.1 PGN-format

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 [Edw94, Chapter 1]. A sample game in PGN notation is shown in figure 1.

```
[Event "Deutschland "]
[Site "?"]
[Date "1995.??.??"]
[Round "?"]
[White "Lutz, Ch"]
[Black "Kramnik, V."]
[Result "0-1"]
[ECO "B33"]
[PlyCount "70"]
[EventDate "1995.??.??"]

1. e4 {B33: Sicilian: Pelikan and Sveshnikov Variations} 1... c5 2. Nf3 Nc6 3.
d4 cxd4 4. Nxd4 Nf6 5. Nc3 e5 6. Ndb5 d6 7. Bg5 a6 8. Na3 b5 9. Nd5 Be7 10.
Bxf6 Bxf6 11. c3 O-O 12. Nc2 Bg5 13. a4 bxa4 14. Rxa4 a5 15. Bc4 Rb8 16. b3 Kh8
17. O-O g6 18. Qe2 Bd7 19. Rfa1 19... Bh6 {last book move} 20. g3 {
Consolidates f4} (20. Nde3 20... Be6 $14) 20... f5 $11 21. exf5 gxf5 22. b4
22... e4 {Black wins space.} 23. bxa5 Ne5 24. Rb4 Rxb4 25. cxb4 f4 26. Nd4 e3
27. fxe3 (27. Nxf4 $2 {doesn't work because of} 27... exf2+ 28. Qxf2 28... Bxf4
$19) 27... f3 {He broke from his leash} (27... fxg3 28. hxg3 Qg5 29. Kh2 Nxc4
30. Nf4 $19) 28. Qa2 f2+ 29. Kg2 Qe8 30. Be2 30... Ng4 {
The pressure on the isolated pawn grows} 31. Bf3 $4 (31. Qd2 Qh5 32. Bxg4 Qxg4
33. Nf4 Bxf4 34. exf4 Qh3+ 35. Kxf2 Qxh2+ 36. Ke1 Qxg3+ 37. Kd1 Qg1+ 38. Ke2
Bg4+ 39. Kd3 Qxa1 40. f5 $19) 31... Nxe3+ $19 32. Nxe3 Qxe3 33. Qxf2 $4 {
sad, but how else could White save the game?.} (33. Rd1 Bg7 34. Qb3 Bxd4 35.
Qxe3 Bxe3 36. Be2 $19) 33... Bh3+ $1 {the final blow} 34. Kg1 {
Black now must not overlook the idea Re1} (34. Kxh3 {A deflection} 34... Qxf2)
34... Qc3 35. Re1 Bd2 (35... Bd2 36. Ne2 36... Qxf3 $19 (36... Bxe1 $6 {
is clearly weaker} 37. Nxc3 Bxf2+ 38. Kxf2 $19) (36... Rxf3 $2 37. Nxc3 Rxf2
38. Kxf2 Bxc3 39. Re7 $18)) 0-1
```

blue: comments red: NAGs

Figure 1: Sample PGN game

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 (one move of white and one of 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 except of the pawn (see table 1) followed by the square the piece is moved to (see figure 2). 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 piece of the opponent is placed on the destination square, this piece is captured and in the move an "x" is inserted immediately before the destination

square. In this case, if the capturing piece is a pawn, the lower-case letter of the previous file of the pawn is used at the beginning of the move, e.g. "exd5". Whenever a move pair is interrupted by a comment, the move of black is prefaced by the move pair number, an ellipsis and a blank:

Nxf4 $2 {doesn't work because of} 27... exf2+

Additionally, there are some further moves with a special notation (see table 1). In cases of disambiguation of pieces, an additional letter for the file or a number for the rank is used. In summary, a move can contain between two and seven signs in SAN [Edw94, Chapter 8].

Symbol	Meaning	Symbol	Meaning	Example
K	King	x	Capture	Rxa1
Q	Queen	+	Check	Nf6+
R	Rook	#	Checkmate	Bb7#
B	Bishop	0-0	Castling kingside	
N	Knight	0-0-0	Castling queenside	
blank	Pawn	=	Promotion	fxg1=Q+

Table 1: Basic chess notations

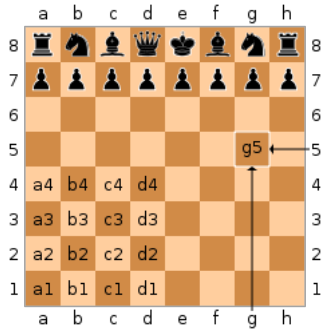

Figure 2: Square names in algebraic notation[2]

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 NAGs (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 of table 2.

Combination	Example
Move, Comment	e4 {Black wins space.}
Comment, Move	{Weaker is} 39. Bxe6
Move, Comment, Move	Nxf4 $2 {doesn't work because of} 27... exf2+
Comment, Move, Comment	{Because of the blunder} 24. Txf8 {Black wins immediately}

Table 2: Comment-move combinations

Besides, by convention there should not be nested braces, however, sometimes nested braces are used to comment different move variants separately. Those variants need not be part of a comment and are written down in parenthesis. The enumeration of the moves proceeds within a variant and is set back before a new variant starts or the game itself continues.

4.1.2 Chessbase-DB

Transformation to PGN, language detection using polyglot

4.1.3 NAGs

Symbols and corresponding NAGs

[2] https://en.wikipedia.org/wiki/Algebraic_notation_(chess)#/media/File:SCD_algebraic_notation.svg, (18.03.2019, 20:56)

NAG	Symbol	Meaning
x	Capture	Rxa1
+	Check	Nf6+
#	Checkmate	Bb7#
0-0	Castling kingside	
0-0-0	Castling queenside	
=	Promotion	fxg1=Q+

Table 3: Meaning of NAGs

4.1.4 Python NLTK

RegExp parsing, tokenization, extraction of comment and class

4.2 Prepare data for classification

4.2.1 Feature extraction

simple features: count(word), advanced features: tf-idf, bigrams, trigrams

4.2.2 Generating training instances

structure of arff-file a brilliant counterattack of white a big mistake of black

```
@RELATION comment
@ATTRIBUTE COUNT(brilliant) NUMERIC
@ATTRIBUTE TFIDF(mistake) REAL
@ATTRIBUTE CLASS {good,bad}
@DATA
1,0.0,good
0,0.06,bad
```

Figure 3: Sample arff file

```
@DATA
{1 2, 3 good}
{2 0.06, 3 bad}
```

Figure 4: Sample sparse arff file

5 Experiment setup

5.1 Selecting classes

distribution how many instances per class, splitting into several problems: 2 classes (!,?), 6 classes (!!,!,!?,?!,?,??), 2 classes (+,-), 7 classes –> introduction of dictionary difference if even or odd number of classes ("neutral class")

5.2 Tokenizer tuning

punctuation, special chess notations (#ce etc.)

5.3 NLTK Parameters

stopwords, stemming, threshold(hapax), lowercase, bigram, trigram

5.4 Classifiers

which classifier to use? –> MCC (x3), OCC, RF, NBM

tables with number of attributes, tables with accuracies, comparison of confusion matrix
each for simple approach, tf-idf, word embedding

7 Conclusion

7.1 Summary

7.2 Outlook

References

[Edw94] Steven J. Edwards. *Standard: Portable Game Notation Specification and Implementation Guide*. available at http://www.saremba.de/chessgml/standards/pgn/pgn-complete.htm, accessed 19.03.2019. 1994.