Extraction of Goals from Premier League Match Reports using Natural Language Processing Techniques

Student Project

presented by
Jochen Hülß & Matthias Rabus
Matriculation Number 1376749 & 1207834

submitted to the
Chair of Information Systems V
Prof. Dr. Christian Bizer
University Mannheim

Mai 2013

Contents

Chapter 1

Project Summary

1.1 Application Domain and Goals

Our interest in looking into text mining and, particulary, into sport report mining emerges from the overwhelming availability, omnipresence, and ubiquitousness of information on sports in media.
The project originates from a classical text mining problem. Generally speaking, we would like to deal with the question whether or not Natural Language Processing (NLP) tools are able to extract and correctly classify certain events in a given text. More specifically, our intention is machine-reading written reports on Premier League football matches. The particular event we are looking into is the scoring of a goal. At first sight, this appears rather straightforward, but by taking into account that there are numerous or even infinite ways to express that a team marks a goal, the difficulty to find as many of them as possible is challenging. Thus, a classification of text snippets based on manually found patterns indicating a goal is needed. Furthermore, we would like to enhance our search result by using learning algorithms.

Domains:

- Information Extraction w/ help of web crawling, NLP, POS Tagging, NER, manually designed patterns matching football goals - Text Mining: Learning a Classification model on automatically-generated labeled data seeds, bootstrapped negative seeds

part-of-speech taggers are computer programs for assigning contextually appropriate grammatical descriptors to words in texts; these approaches are: taggers based on handwritten local rules, taggers based on n-grams automatically derived from text corpora n-gram are subsets of a text with length n (Voutilainen, 2005, p.219)

Due to the explosion of available textual data, text mining and Information Extraction (IE) from texts have become important topics of study in recent years. In particular, detection of relations between named entities is a challenging task to automatically discover new relationships in texts. (Cellier et al., 2010, p.1)

Some previous works use hand- crafted linguistic IE rules for that task which is time consuming [7,5]. Other methods based on Machine Learning (ML) techniques [10] give good results but run as a black box (Cellier et al., 2010, p.1)

Sequential pattern mining [1] is a data mining technique that aims at discovering correlations between events through their order of appearance. Sequential pat- tern mining is an important eld of data mining with broad applications (e.g., biology, marketing, security) (Cellier et al., 2010, p.1)

Information extraction (IE) is the automatic identification of selected types of entities, relations, or events in free text. (Grishman, 2005, p.545)

The project originates from a classical text mining problem. Generally speaking, we would like to deal with the question whether or not Natural Language Processing (NLP) tools are able to extract and correctly classify certain events in a given text. More specifically, our intention is machine-reading written reports on Premier League football matches. The particular event we are looking into is the scoring of a goal. At first sight, this appears rather straightforward, but by taking into account that there are numerous or even infinite ways to express that a team marks a goal, the difficulty to find as many of them as possible is challenging. Thus, a classification of text snippets based on manually found patterns indicating a goal is needed. Furthermore, we would like to enhance our search result by using learning algorithms.

While looking for goals in football match reports, we are omitting the detection of team that scored as well as the unique recognition of a goal as many reports refer multiple times to the same goal.

Jochen. Einleitung, Uebertragbarkeit auf andere Text Mining Tasks. Steps to goal

1.2 Structure and size of the data set

Since many NLP tools work best with English language, we would like to increase our chance of success by using football reports of the English Premier League. The BBC sports department offers match reports for every game of the currently ongoing season 2012/2013. The amount of reports is sufficient for the purpose of this project. A Premier League season consists of 38 matchdays with 10 games each. BBC provides one report per game. The BBC sports department is a good datasource because the reports are neutrally written and have no preference for any

team. Furthermore, the used language has a higher level and should be, compared to spoken word, easier to analyse.
The first step is gaterhing the data from the BBC homepage. Crawling the data will be described more detailed in the following section about preprocessing. These data was the input for the following classification process. Thus we wanted to classify each sentence wheter it contained a goal or not, we had to generate a sentence-wise input first. The result of our crawling process was one HTML file per report. Like every HTML file, our results had a nested tree structure including some Javascript and CSS, as well. A problem could be that the text itself contained HTML tags, for example references to other reports. As mentioned before we had to extract the plain text from the rest of the document. Then we had to divide each text into the sentences itself. This could be a hard as well, because a puncuation mark could be a not sufficient seperator, for example if we consider the name of the stadium "St. James Park". The extracted sentences will be a sufficient input for the further research.

1.3 Preprocessing

As said in the previous section, the project started with gathering the data. We used RapidMiner to crawl the data for us. RapidMiner is an open source data mining tool developed by rapid-i.com. RapidMiner allows the user to define a so called process, which is a combination of certain operators. The operators are connected by defined in- and outputs and can be parameterized to perform specific actions.
RapidMiner provides an operator to crawl the world wide web. BBC's Premier League result page is structured as follows: It has a main site from which every game report is liked. The reports are grouped by the day of the match. The first task of the crawler is to find the links to the game reports within the results page and then download each linked page into a seperate HTML file. Then the text has to be extracted from the HTML files. Therefor we use RapidMiner as well. The RapidMiner process takes all HTML files and extracts the plain text. This is achieved with the Cut Document-operator and an XPath-query. The extracted text is then written to one file for all reports because the Write as Text-operator can not write into multiple files.
Afterwards we had to do parallely two steps. First we had to seperate the text file that contained the whole text from all reports and second we had to manually generate patterns that described goals.
For both tasks some helpful methods were implemented in Java. To use the text for our classifier, it had to be splitted into sentences and each sentence had to be written into an seperate file. Accomplishing this task we used a build-in Java class:

BreakIterator. The class provides a method that can cut a given text into sentences given a specific language, English in this case. The previous preprocessing step created some additional lines of text, for example for every new processed file, that also have been removed within this step. As mentioned before, the results of the text splitting were saved and written into seperate text files for further use, but also processed to find some sentences containing goals and some counter examples.
If we want to train a classifier to distinguish the sentences for us we first have to create examples. Thus there are seldom two similar sentences describing a goal, you can find certain patterns if you analyze sentences using Natural Language Processing (NLP) tools. "Natural Language Processing is a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications." (Liddy, 2001, p.1) To construct sufficient patterns we needed two elements of Natural Language Processing: Named Entity recognition (NER) and Part-of-Speech tagging (POS). POS assigns each word in its part of speech, it determines if it is a verb, noun, adjective etc. Named Entity recognition determines if a word is an entity, for example a name, a place or a city. The Stanford NLP Group provides a useful Java library with many build-in language processing tools, including NER and POS tagging. Using this library, we were able to print out a sentence including named entities and part-of-speech tags. We did this for three match reports to detect patterns manually.

Jochen: Gate, Jape-Rules, ngrams

1.4 Actual Web Mining

Matze. Verbindung zur Vorlesung. Web mining techniken. Rapidminer prozess. classifier. Suche nach perfect settings. apply model

1.5 Evaluation

Jochen. Resultate von Precision und Recall der Classification. Diskussion. Positive und negative Einfluesse auf ergebnisse. Analog fuer ungelabelte Daten

1.6 Conclusion

Wrap-up.

Bibliography

Cellier, P., Charnois, T., Plantevit, M., and Crmilleux, B. (2010). Recursive sequence mining to discover named entity relations. In Cohen, P., Adams, N., and Berthold, M., editors, *Advances in Intelligent Data Analysis IX*, volume 6065 of *Lecture Notes in Computer Science*, pages 30–41. Springer Berlin Heidelberg.

Grishman, R. (2005). Information extraction. In *The Oxford Handbook of Computational Linguistics*, pages 545–559. Oxford University Press.

Liddy, E. D. (2001). *Natural Language Processing*. 2 edition.

Voutilainen, A. (2005). Part-of-speech tagging. In *The Oxford Handbook of Computational Linguistics*, pages 219–232. Oxford University Press.