Tamil Summary Generation for a Cricket Match

J. Jai Hari Raju, P. Indhu Reka, K.K Nandavi, Dr. Madhan Karky
Tamil Computing Lab (TaCoLa),
College of Engineering Guindy, Anna University, Chennai.
jaihari1989@gmail.com, p.indhu@gmail.com, ashwathas@gmail.com, madhankarky@gmail.com

Abstract

Cricket is one of the most followed sports in the Indian subcontinent. There is a wide requirement for natural language descriptions, which summarize a cricket match effectively. The process of generating match summaries from statistical data is a manual process. The objective of this paper is to propose a framework for automatic analysis and summary generation for a cricket match in Tamil, with the scorecard of the match as the input. Data analytics is performed on the statistical match data, to mine all frequently occurring patterns. The paper proposes a parameter called Interestingness, which quantifies the interestingness of the match. The paper also proposes a customization model for the summary. We propose an evaluation parameter called humanness, which quantifies the similarity between the output and a manually written summary. Discussing the results and analyzing the summaries generated for matches based on scorecards, this paper concludes with proposing some extensions for future developments.

1. Introduction

The number of websites which facilitate people to follow and analyze sports has increased manyfold. Among them, there are an exceptionally large number of sites devoted to Cricket. Mostly these involve participation of experts, who present their views and summaries in English about cricket matches. There are no such sites which provide similar services in Tamil. In this case it is also desirable if there is an alternative for human creativity. As a solution the paper proposes an automated Tamil summary generation framework which is capable of analyzing and generating a Tamil summary about a cricket match, provided the scorecard as the input. This paper discusses the overall architecture and implementation details of such a framework.

The large amount of data in this domain makes it possible to apply data mining and data analytics techniques. The input scorecard is analyzed to construct feature vectors, which are then subjected to data mining. Based on the various parameters identified, the interestingness of the match is quantified.

The summary generation part involves extraction of key players and events from a match. Appropriate sentences are then synthesized to express these selected events. The sentence constructs and the vocabulary used are chosen based on the linguistic ability specified by the user. Then the sentences are combined into a meaningful summary.

The results of the system, i.e. the summaries, are evaluated based on the Humanness parameter. This parameter gives the degree of similarity between the generated summary and the manually written summary, with which it is compared. This value helps us decide, the level of creativity achieved by
the system. In section 2 we provide an overview of the literature survey conducted. In section 3 we discuss the design of the various modules of the framework. In section 4 we discuss the implementation of the proposed framework and the results obtained from the analysis. Finally we conclude in section 5 with extensions to the current framework and directions for further studies in the field of Tamil Summary Generation Systems.

2. Background

In the literature there are existing works on summary generation from statistical data. Alice Oh et al. generated multiple stories about a single baseball game based on different perspectives using a reordering algorithm [1]. Ehud Reiter et al. in their book building natural language generation systems explain the difference between natural language generation and natural language processing and also describe the various steps involved in the natural language generation process with examples [2]. Jacques Robin et al. presented a system (called STREAK) for summarizing data in natural language. It focuses on basketball game to design and evaluate the system [3]. L. Bourbeau et al. came up with the FoG (Forecast Generator) using the streamlined version of the Meaning-Text linguistic model. This system was capable of generating weather forecasts in both English and French [4].

3. Summary Generation Framework

The Tamil Cricket Summary Generator consists of the following major components:

- Data Gathering and Modeling module
- Data Mining and Data Analytics module
- Summary Generator
- Evaluator

Figure (1) given below depicts the Summary Generation framework.

Figure 1: Tamil Cricket Summary Generator Framework
3.1 Data Gathering and Modeling Module

Data gathering is the first step of the system. The data to be gathered is present in internet. This module has a custom designed parser, for the tag structure of the site. The user must provide the URL from where the particular match’s data can be obtained. The module checks whether the match has already been processed. If not the parser parses the page and retrieves the statistical data. The statistical data is then modeled in the form of the predefined feature vectors.

3.2 Data Mining and Analytics Module

Modified version of Apriori algorithm is used to find the association rules from the feature vectors. After performing mathematical analysis using correlation of variance (CoV), CoV is plotted against average to give an idea about how consistent the player is. The interestingness of the match is calculated based on the weighted average of the scores assigned to the factors identified, they include the Winning margin, Team history, Individual records made, High run rate, Series state, Relative position in international ranking, Reaction in social networks etc.

3.3 Summary Generator

The summary generator part of the framework consists of the following sub modules Content Determiner, Aggregator, Tamil Morphological Generator and Layout Determiner. The events to be included in the summary are not predefined and are not the same for every match. Based on the interestingness of the total match, the interestingness of the individual events and the expert level chosen by the user, particular events are chosen to be included in the summary. The content determiner is responsible for identifying those facts which are worth mentioning in the summary.

Aggregation of relevant events from other matches in the summary will make it more readable and interesting. The aggregator performs this function. It chooses events based on their similarity and coherence and aggregates them with the key events selected in the content determiner module.

As a next step, the sentences used to describe the events are synthesized. The sentence which is the most apt to the current event under consideration is selected. The vocabulary used in the sentence and the depth to which an event is discussed is also varied based on the expert level of the user. The nouns in the key events are passed to the morphological generator along with the desired case endings and the generated variants are added to the sentences.

The layout determiner module chooses the layout of the summary to be generated. The layout is varied based on the interestingness of the match. The sentences are aggregated in the fashion of the layout selected and the final output summary is passed to Evaluator.

3.4 Evaluator

The summary generated by the system is evaluated based on its degree of similarity with human written summaries. The summaries are compared based on two parameters, the Nouns Mentioned and the Events Mentioned.

The nouns and the events in the summaries are extracted along with their absolute positions. The events in the summary are modeled as a set consisting of, one or more Performers (the persons who takes part in the event), Numeral (the numeric part involved in the event e.g. 4 wickets) and a
Descriptor (the action connecting the Performer and the Numeral). Their absolute positions refer to the sentence number in which they are mentioned. Then these absolute positions are normalized based on the total number of sentences present in the summary. Three different scores are calculated they are,

- **Similarity Score**: The ratio of the number of nouns and events mentioned in both the summaries to the total number of nouns and events mentioned at least in one summary.
- **Count Score**: The ratio of the number of nouns and events mentioned in the system generated summary to the number of nouns and events mentioned in the human written summary.
- **Closeness Score**: The degree of closeness, in terms of the normalized positions of the nouns and events mentioned in both the summaries.

A weighted average of these three scores yields the final humanness score.

4. Implementation

To implement the proposed framework, espncricinfo a reliable and prominent site for Cricket data is chosen as the source of input. The framework was implemented in java. The URL of the match for which the summary is to be generated is obtained from the user. The feature vectors designed for modeling a match are stored as rows with unique identities, in the back end oracle database. The patterns which are generated as a result of data mining are validated based on the support and confidence parameters. As a design decision all nouns are stored in English and are translated on the fly using a constantly updated look up database. This decision was taken to allow interoperability and easy extension of the system to other languages in future. The sentence pattern files are stored external to the system, so as to allow modifications without changes in the system. The summary generated for the match is stored in the back end, indexed with the unique identity assigned already. The user interface is designed to be simple and robust. It allows the users to search matches based on various parameters and also to save their preferences.

4.1 Results

Score cards of 90 One Day International matches where retrieved and their summaries were generated. These include matches between 9 countries. Both individual matches and series were considered. A large number of hidden patterns in cricket domain have been retrieved based on the algorithm used. The patterns have been validated and the ones which are interesting have been reported. The factors contributing to the interestingness of the match have been identified and the weights associated with them have been found. The consistency of a player has been modelled and consistency analysis of a player is done to analyse his performance.

The difference in the language used and the events mentioned in the summary is pronounced when the user opts for an expert level. Similar facts occurring in the past have been identified and added to the summary. Each summary was compared with two human written summaries, one an expert summary and other an average summary, their cumulative scores were considered. The humanness score of the summaries tend to be in the range of 70% to 85%. The recurrence of layouts is also minimal, which reflects the fact that the summaries generated are not monotonous.
5. Conclusion and future work

In this paper we have proposed the framework for an Automated Tamil Cricket Summary Generator. The current implementation of the system can be enhanced by adding machine learning capabilities to make the summaries more human and interesting. The system can be extended to produce summaries in multiple languages apart from Tamil. The system can be enhanced to generate summaries about the match in real time. As a next level the system can be modified for summary generation in other sports too.

Figure 2: Screenshot of the Tamil Cricket Summary Generation System

The framework can be used as a guideline to develop summary generation systems, which can be applied for any domain where frequent numerical reports are used. (Weather Prediction, Industrial Quality Testing etc)

References