Linking Event Mentions from Cricket Match Reports to Commentaries

Manish Gupta
Microsoft, India
gmanish@microsoft.com

Abstract. We focus on the problem of linking event mentions in cricket match reports to instances from temporal commentary data. The problem is challenging because depending on the event type, event mentions could be linked to a single data instance, or to a set of instances. The complexity of the natural language in the reports along with a lack of canonical names or verbose descriptions of the data instances to be linked, add to the difficulties. Given a user-highlighted event mention from a cricket match report, we solve the problem of linking it to a relevant set of balls from the corresponding match commentary. Our approach encodes cricket-specific intuitions broadly into the system design and more specifically as features for multiple classifiers for candidate detection. Further, we leverage techniques such as structured match, context similarity, iterators and sequential proximity of linked entities to perform the linking. Using a dataset of 2828 event mentions across 187 match reports related to 30 matches from the 2011 Cricket World Cup, we show the effectiveness of the proposed methods.

1 Introduction

Every game of any sport leads to a temporal data series, e.g., minute-by-minute commentaries of soccer, ball-by-ball commentaries of baseball or cricket, move-by-move commentaries of chess games. In this work, we focus on event linking for sports related temporal data series, specifically cricket. With hundreds of millions of fans worldwide, cricket is one of the most popular sports, second only to soccer. The 2015 Cricket World Cup was watched by 288 million viewers. 26 million unique visitors made up more than 225 million page views on the official World Cup 2015 website during the world cup alone. 800 million tweets were sent during group stages of the world cup while in the same period, 36 million people generated 341 million interactions on Facebook [1].

Cricket Linking Problem: For each cricket match, online portals publish ball-by-ball commentaries which describe what happened when a ball was bowled including the name of the bowler, the batsman, number of runs scored, the type of the ball delivered, the type of shot, and sometimes comments on the form of the bowler or the batsman. Thus, every one-day cricket match has this detailed description of the match in the form of a maximum of 600 balls (plus the extras). Also, multiple experts write articles (or reports) describing the events that occurred during the match. In this paper, we focus on providing an ability to the user to zoom in on a particular event mention from a match report, and read the ball commentaries most relevant to the event.

Challenges: Compared to traditional entity linking work, this problem is different at least from two perspectives. (1) Compared to traditional entity knowledge bases, the total number of basic ball entities are quite small. (2) Mentions represent events and can link to a set of ball entities. These salient differences from traditional settings naturally

[1] http://tinyurl.com/iccwc2015OfficialWebsite

bring in challenges as follows. (1) Mentions could link to one ball or a set of balls. (2) Multi-ball event entities (i.e., the set of balls) do not have a canonical name making the candidate detection itself quite challenging. Also since there are no well-defined semantic groups of ball entities, potential candidates could be exponentially large in number. (3) Cricket reports use a versatile way of representing the game and thus there is a heavy use of synonyms, idioms and writing styles.

Brief Overview of the Proposed Approach: Our solution consist of two main components: candidate detection, and candidate ranking and linking. As part of candidate detection, mention type and mention sub-class detection is also performed. We encode cricket-specific intuitions as features to learn a mention-type detection classifier to first classify a mention as a single-ball mention versus a multi-ball mention. Single-ball and multi-ball mentions are further classified into multiple sub-classes. For single-ball mentions, candidate ranking is performed based on similarity between mentions and ball entities with respect to the values taken by "slots" relevant to the particular mention sub-class. For linking to multi-ball mentions, first derived entities are extracted by grouping together balls semantically. Then, candidate ranking is performed by computing sub-class specific slot-based similarity between mentions and derived entities. Further, multi-ball mentions could refer to a part of the derived entities and hence iterators are extracted from such mentions and applied on the top matching derived entity to extract the most relevant subset. Finally, for certain sub-classes, sequential proximity between linked entities for mentions in the same paragraph could be exploited if we have a list of already detected mentions. The proposed system provides a recall@5 of ∼69% for single-ball mentions and an F1 of ∼57% for multi-ball mentions.

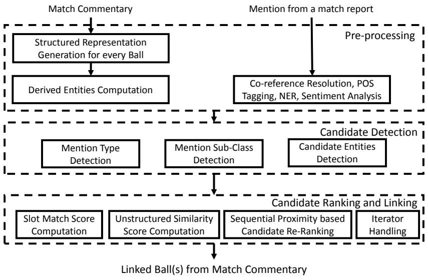
Match Commentary
Mention from a match report
Pre-processing
Structured Representation Generation for every Ball
Derived Entities Computation
Co-reference Resolution, POS Tagging, NER, Sentiment Analysis
Candidate Detection
Mention Type Detection
Mention Sub-Class Detection
Candidate Entities Detection
Candidate Ranking and Linking
Slot Match Score Computation
Unstructured Similarity Score Computation
Sequential Proximity based Candidate Re-Ranking
Iterator Handling
Linked Ball(s) from Match Commentary

Fig. 1. System Diagram

Figure 1 shows a system diagram for the proposed system. The system has three main components: Pre-processing, Candidate Detection, and Candidate Ranking and Linking. We discuss details of these components in Sections 3 and 4.

In summary, we make the following contributions in this paper. (1) We propose an interesting problem of linking event mentions to instances from temporal data series. We study an instantiation of the problem in the cricket domain. (2) We build classifiers for mention type detection and mention sub-class detection for effective candidate detection. We propose slot match score as the similarity measure for effective candidate ranking and discuss the importance of iterators. We also show that sequential proximity is a helpful signal for selected mention sub-classes. (3) Using a dataset of 2828 event mentions across 187 match reports related to 30 matches from the 2011 Cricket World Cup, we show the effectiveness of the proposed methods. The dataset and the code are made publicly available [2].

Paper Organization: We discuss related work in Section 2. In Section 3, we discuss methods for candidate detection which include identifying derived entities and learning classifiers for mention type and mention sub-class detection. Section 4 focuses on various approaches towards ranking candidates for linking. In Section 5, we present

[2] https://github.com/blitzprecision/CricketLinking

the insights from analysis of results obtained by performing extensive experiments. We conclude with a summary in Section 6.

2 Related Work

Our work is related to two main areas of research: sports data mining and entity linking.
Sports Data Mining: Sports data mining has focused on various aspects of popular sports like cricket, football, basketball, hockey, chess, etc. These aspects include player performance analysis [15, 17, 21], player performance prediction [17, 26], finding patterns and performing association rule mining [1, 20], scouting or player selection [19, 25], analyzing player dropouts [3], outcome prediction [7, 9, 23], retrieval of similar chess positions [6] or similar movements from soccer game streams [8], and predicting player recovery times [14]. To the best of our knowledge, this is the first work in the IR community focused on applying entity linking methods in the domain of sports. Although we experiment with entity linking in cricket, similar efforts could clearly be useful for data related to other sports like football, baseball, etc.
Entity Linking: Entity Linking has become a hot topic in recent years [2, 5, 10–13, 24, 27, 28]. Shen et al. [22] provide a thorough overview of the main approaches to entity linking. A typical entity linking system consists of mention detection, candidate entity generation and candidate entity ranking. Popular methods for candidate entity generation include name dictionary based techniques [16, 28], surface form expansion from the local document [29], and methods based on search engines [4]. For candidate entity ranking, both supervised and unsupervised methods have been proposed. We model the event mention linking problem to cricket commentary balls as an entity linking problem where entities are balls or sets of balls. Compared to traditional entity linking task, our problem is different at least from two perspectives: (1) The number and type of entities are very different. Total number of basic ball entities are quite small in number, compared to millions of entities in knowledge bases for traditional entity linking. Also, unlike traditional scenarios, we have derived entities too. (2) Mentions represent events and can link to any group of entities. So, the task is closer to event linking where an event is identified by a set of ball entities. However, this set does not have a canonical name making the task more challenging.

3 Detecting Candidate Entities

In this section, we first describe basic pre-processing steps, and then discuss various components of the candidate detection module: mention type detection which classifies a mention as a single-ball or a multi-ball mention, and mention sub-class detection. Finally, we combine these to identify candidate entities for a mention.

3.1 Pre-processing Commentaries and Reports

In a typical one-day match of cricket, there are 2 innings of 50 overs each, and each over consists of 6 balls. For basic introduction to cricket, we direct the reader to `https://en.wikipedia.org/wiki/Cricket`. The ball-by-ball commentaries are available in a semi-structured form for each ball of both the innings. For accurate linking of mentions from match reports, we need to obtain a structured representation of each ball which contains various fields: ball number, bowler name, striker name, commentary text, etc. Match reports are unstructured summaries of the match. When a user selects a mention, the mention and its paragraph are processed by performing linguistic analysis like part-of-speech tagging, named entity resolution, coreference resolution on the paragraph, and sentiment analysis, using Stanford CoreNLP [18]. Further, balls are semantically grouped into various derived entities. These derived entities are useful for linking to multi-ball mentions as we will discuss in Section 4. Hence, beyond the 600 (50 overs×6 balls/over×2 innings) ball entities (plus extras), we also obtain linkable derived entities. Derived entities include all balls faced by a batsman, all balls bowled by a bowler, all balls on which a four/six was hit in the first/second innings by a particular batsman, etc.

3.2 Mention Type Detection

Given a mention, the first task is to identify whether it is a single-ball mention or a multi-ball mention. We model this as a binary classification task and experiment with various classification algorithms using the following feature sets.
Dictionary Features: We build multiple cricket related dictionaries as follows: Shot (batsman action) words, Bowling type words, Partnership words, Single-ball event words, Multi-ball event words, Extra-balls words, Powerplay words, and Derived entity names. The feature values are computed as the number of dictionary words occurring in the mentions. We expect single-ball event words, batsman/bowler action words to be representative of the single-ball mentions. On the other hand, multi-ball event words, powerplay words, partnership words represent the multi-ball mentions.
Entity Features: These features capture the number of words of a particular entity type in the mention. We consider the following entity types: player names from the first team, player names from the second team, countries, numbers, dates, locations, and all named entities.
Features Capturing Similarity with any Ball: Similarity can be computed between the mention and the structured or unstructured part of the ball commentary. We compute similarity separately for both the cases. For these features, similarity means number of matching words. The features included are maximum similarity with any ball, ratio of first maximum to second maximum similarity and first maximum-second maximum similarity. Thus, this feature set consists of 6 features.
Other Features: This feature set includes the following features: (1) Mention contain names of both bowler and the victim for some wicket ball? (2) Position of the mention in the report. (3) Mention contains the match score ("\d+(runs)? for \d+(wickets)?")? (4) Mention length in characters and in words. (5) Sentiment score of the mention. (6) Number of plural words (POS tags: nns and nnps). We observed that the single-ball mentions are on an average $\sim$1.5 times longer and have a higher negative sentiment compared to the multi-ball mentions.

3.3 Mention Sub-Class Detection

After classifying a mention as single-ball or multi-ball, it is important to identify its sub-class. As we will show in Section 5, sub-classes are critical in determining the slots to be matched when computing similarity during candidate ranking.

Sub-class detection needs to be done separately for single-ball and multi-ball mentions. Sub-classes were chosen based on the frequent type of event mentions occurring in match reports. For single-ball mentions, we consider these classes: OUT (a dismissal), LASTBALL (the last ball of either innings), BALL (the score of a team or of a player), DROPPED (fielding team lost a chance to dismiss one of the batsman on the field), SIX (a "six" shot), FOUR (a "four" shot), REFERRAL (a review appeal), OTHERS (injury event, etc.). For multi-ball mentions, we consider the following classes: BAT (batting of some player), BOWL (bowling of some player), BATBOWL (a player's batting when facing bowls from a particular bowler), FOUR ("four" shots), SIX ("six" shots), PARTNERSHIP (partnerships between two batsmen), WICKETS (multiple dismissals including hat-tricks), OVERS (specific overs of either innings), POWERPLAY (any powerplay), REFERRAL-DROPPED (review appeals or lost chances), EXTRAS (extra balls like wides, etc.), and OTHERS.

For sub-class detection, we use similar features as used for mention type detection except that we add more Sub-class dictionary features to the "Dictionary" feature set. These features are based on dictionaries manually curated for each sub-class, and the feature values indicate the number of words from the dictionaries appearing in the mention. For example, dictionary for OUT sub-class includes "stump", "dismissal", etc. Dictionary for REFERRAL sub-class includes "review", "verdict", etc.

3.4 Candidate Entities Detection

Balls and derived entities are assigned automatically to various sub-classes based on simple rules on certain fields in the structured representation. For example, OUT sub-class could contain balls with the event field in the structured ball representation set to "out." Similarly, BAT sub-class could contain derived entities related to batting of various players. Certain sub-classes like OVERS and OTHERS contain all balls/derived entities in the match.

4 Ranking Candidate Entities and Linking

Given the mention text and the candidate balls or derived entities, in this section, we focus on multiple methods for candidate ranking by similarity score.

4.1 Sub-class Unaware Similarity

This approach is syntactic in nature and does not exploit the sub-class semantics. A similarity measure is computed between the mention text and the ball details using multiple variations as follows. (1) Similarity Measure: Jaccard similarity vs. cosine similarity using TFIDF. (2) Coreference Resolution: Original mention text vs. the coreference-resolved one. (3) Commentary Context: We consider different context windows around ball b as follows: b alone (± 0), 1 ball before and after b (± 1), 2 balls before and after b (± 2), the over containing b ($over$). (4) Mention Context: Mention text itself vs. the sentence containing the mention. (5) Ball Representation: Structured representation vs. unstructured representation.

For single-ball mentions, after ranking the candidate balls, the mention is linked to the ball with the maximum score. But for multi-ball mentions, the mention needs to be linked to multiple balls. We sort the balls by similarity score, detect a knee of the curve and select all balls with value greater than the knee point.

4.2 Sub-class Aware Slot-based Similarity

This approach is semantic in nature and exploits the sub-class semantics and the derived entities.
<u>**Single-ball Mentions**</u>: Let T be set of all sub-classes. Let m_t be the sub-class for mention $m.$ In this approach, we compute the similarity score between mention text m and the candidate ball b as a linear combination of the unstructured similarity between the mention and the ball commentary text, and the slot match score between the ball and the mention using a Bayesian approach (Eq. 1).

$$Score(m,b) = \sum_{t\in T}(P(t|m)(\lambda\, UnStructuredSim(m,b) + (1-\lambda)\, SlotMatchScore(m,b,t))) \quad (1)$$

where $P(t|m)$ is classifier output probability of mention m belonging to sub-class t.
The unstructured similarity score in Eq. 1 is computed in a sub-class unaware way. Slot match score in Eq. 1 is computed as follows. First, all person names, country names, fielding positions, and numbers are extracted from the mention. Also, depending on the mention sub-class, a set of fields (or slots) which could take these values are recognized. Slot match score between the mention m and the ball b under the sub-class t is then computed as the ratio of the number of matching slot values between m and b under t to the number of extracted values from m (Eq. 2).

$$SlotMatchScore(m,b,t) = \frac{\#Matching\ Slot\ Values(m,b,t)}{\#Values\ in\ m} \quad (2)$$

We consider all balls of the match as candidates, rank them using Eq. 1 and return the top one.
<u>**Multi-ball Mentions**</u>: For multi-ball mentions, we use derived entities as candidates. Computing unstructured similarity between mention text and aggregated ball commentary text for all balls within a derived entity will intuitively give a poor match. Hence, we do not compute the unstructured similarity score but consider only the slot match score for the case of multi-ball mentions.

4.3 Iterators

Sometimes, the multi-ball mention must actually link to a part of the derived entity rather than the entire entity. For example, the mention “he started off in a frenzy, scoring 12 off his first six balls” should be linked only to the first six balls of the derived entity “BAT(V Sehwag)” rather than to the entire derived entity. This requires iterator extraction from the mention. An iterator is a phrase which contains three parts: iteration units, start and end. For cricket, iterator units could be “balls”, “wickets”, etc. To extract iterators from mentions, we mainly depends on regular expression patterns. We leave the study of complex iterators (Composite iterators, e.g., “5th ball of the tenth over”; Relative iterators, e.g., “Sehwag had scored 12 off his first six balls and 13 off his next 24.”) as future work. After identifying iterators, they are applied on the derived entities.

4.4 Sequential Proximity

Usually mentions (and hence linked entities) in a paragraph follow a temporal order. Given top ranked candidate entities for a mention, this intuition can be used to re-rank the candidates based on the sequential proximity of the linked balls. Let $m_1, \ldots, m_D$ denote D mentions in a paragraph. Let each mention have a maximum of K candidate entities. Let e_{ik}, s_{ik} and c_{ik} denote the k^{th} candidate entity, its score, and its central ball respectively for the mention m_i. Then we could link entities for these mentions in the following three ways. Note that the difference $c_{ik_i} - c_{(i-1)k_{i-1}}$ in the following is computed in terms of number of balls between c_{ik_i} and $c_{(i-1)k_{i-1}}$.

- minDiff: Select entities for the mentions such that $\sum_{i=2}^{D} |c_{ik_i} - c_{(i-1)k_{i-1}}|$ is minimized where k_i could take any value from 1 to K.
- minRankDiff: Select entities for the mentions such that $\sum_{i=2}^{D} k_i \times |c_{ik_i} - c_{(i-1)k_{i-1}}|$ is minimized where k_i could take any value from 1 to K.
- minScoreReciprocalDiff: Select entities for the mentions such that $\sum_{i=2}^{D} \frac{1}{s_{ik_i}} \times |c_{ik_i} - c_{(i-1)k_{i-1}}|$ is minimized where k_i could take any value from 1 to K.

5 Experiments

In this section, we describe our dataset, metrics and extensive experiments to analyze relative accuracy of various proposed methods for the cricket linking problem.

5.1 Dataset

We crawled 187 match reports, scorecards and commentaries for both innings of 30 matches of the 2011 Cricket World Cup from espncricinfo [3]. The dataset is about 207 players from 14 countries, and contains 15718 balls, 5461 derived entities. We manually labeled 2828 mention phrases in the reports and then labeled the mention type, mention sub-class and the balls that can be linked to the mention. The dataset and the code are made publicly available [4]. 1561 of the 2828 mentions are single-ball mentions and the remaining 1267 are multi-ball ones. Single-ball mention distribution: OUT (515), LASTBALL (383), BALL (304), OTHERS (137), DROPPED (72), SIX (67), FOUR (46), REFERRAL (37). Multi-ball mention distribution: BAT (291), PARTNERSHIP (205), BOWL (186), WICKETS (181), OVERS (126), POWERPLAY (102), FOUR (43), REFERRAL-DROPPED (38), OTHERS (37), SIX (22), BATBOWL (20), EXTRAS (16).

5.2 Mention Type Classifier Analysis

We report 10 fold cross validation accuracy using various classifiers in Table 1. Boosted trees perform the best with an overall accuracy of $\sim$85%. Top ten features based on information gain are the number of single-ball event words, number of multi-ball event words, maximum similarity with any ball, sentiment score of the mention, mention length in words, number of country names, mention length in characters, number of numeric values, and the number of player names. We also performed experiments using boosted decision trees by having only a particular feature set or removing the individual feature sets to understand their relative importance. Results show that the "Dictionary features" are the most important while "Similarity with any ball" are the worst. Also, none of the individual feature sets alone can get accuracy comparable to using all the feature sets.

[3] http://www.espncricinfo.com/ci/engine/series/381449.html

[4] https://github.com/blitzprecision/CricketLinking

Classifier	P (MB)	R (MB)	P (SB)	R (SB)	Acc.
Boosted Decision Trees	0.841	0.823	0.861	0.874	0.852
Logistic Regression	0.836	0.709	0.791	0.887	0.807
Binary Neural Network	0.779	0.572	0.716	0.865	0.735
Linear SVM	0.811	0.688	0.782	0.854	0.780
Random Forests	0.815	0.803	0.844	0.852	0.830
Weighted Ensemble (Bagging)	0.836	0.697	0.785	0.889	0.803

Table 1. Mention Type Classifier Accuracy (MB=Multi-ball, SB=Single-ball), (P=Precision, R=Recall)

Method	Multi-ball Acc.	Single-ball Acc.
One-vs-All with LinearSVM	0.649	0.735
One-vs-All with Boosted Decision Trees	0.730	0.769
Pairwise Coupling with LinearSVM	0.505	0.659
Pairwise Coupling with Boosted Decision Trees	0.676	0.759
Multi-Class Logistic Regression	0.642	0.741
Multi-Class Discriminative GMM	0.612	0.733

Table 2. Accuracy of the Mention Sub-class Classification

5.3 Mention Sub-class Classifier Analysis

We experimented with various classifiers as described in Table 2. One-vs-All method with Boosted trees as the underlying binary classifier performs the best for both the single-ball as well as the multi-ball case. For the single-ball classifier, the top few features were the number of BALL words, number of OUT words, number of REFERRAL words, ratio of max similarity with any ball to second best similarity (unstructured), contains score, and the number of numeric values. For the multi-ball classifier, the top few features were number of BAT words, number of POWERPLAY words, number of REFERRAL-DROPPED words, max similarity with any ball - second best similarity (structured), ratio of max similarity with any ball to second best similarity (unstructured), contains score, and the number of Powerplay words. We also performed experiments using boosted decision trees by having only a particular feature set or removing the feature sets to understand their relative importance. We observed that the importance of feature sets is quite similar to that for the mention type classifier.

5.4 Sub-class Unaware Similarity Results

For single-ball mentions, the best setting is no commentary context, no mention context, Cosine-TFIDF similarity function, and structured ball representation. In Figure 2, we show the best recall@K ($K = 1 \ldots 10$) results obtained using each of these settings as compared to the best setting for single-ball mentions. Recall@K is 1 if the golden ball is present within the top K predicted balls, else 0.

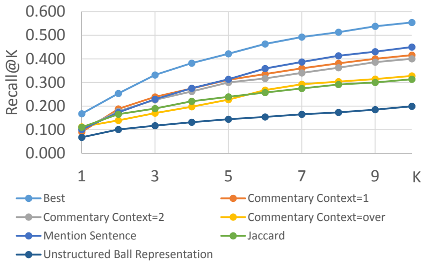

Fig. 2. Recall@K Comparison for Various Settings (Single-ball Mentions, Sub-class Unaware Similarity)

	P	R	F1
Best	0.269	0.369	0.311
Commentary Context=0	0.231	0.456	0.307
Commentary Context=1	0.230	0.444	0.303
Commentary Context=2	0.229	0.442	0.302
Mention Sentence	0.179	0.542	0.269
Unstructured Ball Representation	0.231	0.456	0.307

Fig. 3. Precision, Recall and F1 Comparison for Various Settings (Multi-ball Mentions, Sub-class Unaware Similarity)

For multi-ball mentions, the best setting is commentary context set to over, no mention context, Cosine-TFIDF similarity function, and structured ball representation. Again coreference resolution did not make any noticeable difference. We summarize the precision, recall and F1 results obtained using each of these settings as compared to the best setting in Table 3.

5.5 Sub-class Aware Slot-based Similarity Results

For single-ball mentions, we observed that the best setting is commentary context set to 0, mention context set to mention only, similarity measure as cosine-TFIDF, Coreference resolution set to yes, and λ=0.7. This provides a recall@1 of $\sim$0.47, recall@5 of $\sim$0.69, and recall@10 of $\sim$0.75. By varying λ from 0 to 1, we found that the method is not very sensitive to λ. The only requirement is that λ should not be very close to 0 or 1. Thus, both the unstructured similarity and the slot match score are important. For multi-ball mentions, the best setting is commentary context set to over, mention context set to mention only, similarity measure as Cosine-TFIDF, Coreference resolution set to yes. Table 3 shows the accuracy obtained using various score computation methods.

Method	Precision (P)	Recall (R)	F1
Sub-class_Aware	0.527	0.587	0.556
Sub-class_Aware+Iterator	0.568	0.566	0.567
Sub-class_Aware+Iterator+Seq. Proximity	0.578	0.569	0.573

Table 3. Comparison of Various Methods for Multi-ball Mentions

5.6 Impact of Iterators and Sequential Proximity

129 of the 2828 instances involve iterators. As shown in Table 3, applying iterators over derived entities for multi-ball mentions leads to improvement in precision and hence F1 leading to a best F1 of 56.7%.

We observed that among the three sequential proximity methods, minScoreReciprocalDiff performs better than minRankDiff which in turn performs better than minDiff. Compared to the best accuracy for single-ball mentions so far, sequential proximity improves the accuracy by at least 1% as shown in Table 4. Table 3 shows that the sequential proximity heuristic leads to similar improvement in F1 for multi-ball mentions.

	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10
SA	0.466	0.568	0.619	0.652	0.672	0.693	0.709	0.719	0.728	0.737
SP	0.477	0.580	0.630	0.664	0.687	0.708	0.727	0.737	0.745	0.758

Table 4. Recall Comparison between Sub-class Aware Method (SA) and Sequential Proximity (SP) for Single-ball Mentions

6 Conclusion

In this work, we proposed an interesting problem of linking mentions to instances from temporal data series. We proposed various methods for candidate detection and candidate linking for an instantiation of this problem for the cricket domain. We noticed that single-ball mention linking involves very different challenges compared to linking of multi-ball mentions. We observed that mention sub-class identification, slot match scoring, iterators, coreference resolution, and mention/commentary context expansion provide gains over a basic unstructured match baseline. The proposed system provides a recall@5 of $\sim$69% for single-ball mentions and an F1 of $\sim$57% for multi-ball mentions. The system can be very useful for quick referencing of commentary balls when reading match reports on various cricket portals. In the future, we plan to work on generalizing the solution to other sports, and on linking across multiple temporal data series.

References

1. Guillaume Bosc, Mehdi Kaytoue, Chedy Raıssi, and Jean-François Boulicaut. Strategic Pattern Discovery in RTS-games for E-Sport with Sequential Pattern Mining. *Machine Learning and Data Mining for Sports Analytics*, 2013.
2. Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. ZenCrowd: Leveraging Probabilistic Reasoning and Crowdsourcing Techniques for Large-scale Entity Linking. In *Proc. of the* 21^{st} *Intl. Conf. on World Wide Web (WWW)*, pages 469–478, 2012.
3. Fabrice Dosseville, François Rioult, and Sylvain Laborde. Why do Sports Officials Dropout? In *Machine Learning and Data Mining for Sports Analytics*, pages 10–19, 2013.
4. Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. Entity Disambiguation for Knowledge Base Population. In *Proc. of the* 23^{rd} *Intl. Conf. on Computational Linguistics (COLING)*, pages 277–285, 2010.
5. Paolo Ferragina and Ugo Scaiella. Tagme: On-the-fly Annotation of Short Text Fragments (by Wikipedia Entities). In *Proc. of the* 19^{th} *ACM Intl. Conf. on Information and Knowledge Management (CIKM)*, pages 1625–1628, 2010.
6. Debasis Ganguly, Johannes Leveling, and Gareth JF Jones. Retrieval of Similar Chess Positions. In *Proc. of the* 37^{th} *Intl. ACM SIGIR Conf. on Research & Development in Information Retrieval (SIGIR)*, pages 687–696, 2014.
7. Cristian Georgescu. Data mining in sports betting. *Risk in Contemporary Economy*, pages 102–105, 2013.
8. Jens Haase and Ulf Brefeld. Finding Similar Movements in Positional Data Streams. *Machine Learning and Data Mining for Sports Analytics*, 2013.
9. Maral Haghighat, Hamid Rastegari, and Nasim Nourafza. A Review of Data Mining Techniques for Result Prediction in Sports. *Advances in Computer Science*, 2(5):7–12, 2013.
10. Xianpei Han and Le Sun. A Generative Entity-mention Model for Linking Entities with Knowledge Base. In *Proc. of the* 49^{th} *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT)*, pages 945–954, 2011.
11. Xianpei Han and Le Sun. An Entity-topic Model for Entity Linking. In *Proc. of the 2012 Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 105–115, 2012.
12. Xianpei Han, Le Sun, and Jun Zhao. Collective Entity Linking in Web Text: A Graph-based Method. In *Proc. of the* 34^{th} *Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 765–774, 2011.
13. Yuzhe Jin, Emre Kıcıman, Kuansan Wang, and Ricky Loynd. Entity Linking at the Tail: Sparse Signals, Unknown Entities, and Phrase Models. In *Proc. of the* 7^{th} *ACM Intl. Conf. on Web Search and Data Mining (WSDM)*, pages 453–462, 2014.
14. Stylianos Kampakis. Comparison of Machine Learning Methods for Predicting the Recovery Time of Professional Football Players after an Undiagnosed Injury. *Machine Learning and Data Mining for Sports Analytics*, 2011.
15. Theodoro Koulis, Saman Muthukumarana, and Creagh Dyson Briercliffe. A Bayesian Stochastic Model for Batting Performance Evaluation in One-day Cricket. *Journal of Quantitative Analysis in Sports*, 10(1):1–13, 2014.
16. Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. Collective Annotation of Wikipedia Entities in Web Text. In *Proc. of the* 15^{th} *ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 457–466, 2009.
17. Ananda BW Manage and Stephen M Scariano. An Introductory Application of Principal Components to Cricket Data. *Journal of Statistics Education*, 21(3), 2013.
18. Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

19. John McCullagh. Data Mining in Sport: A Neural Network Approach. *International Journal of Sports Science and Engineering*, 4(3):131–138, 2010.
20. Sérgio Nunes and Marco Sousa. Applying Data Mining Techniques to Football Data from European Championships. In *Actas da 1^a Conferência de Metodologias de Investigação Científica (CoMIC06)*, 2006.
21. Sujeet Kumar Sharma. A Factor Analysis Approach in Performance Analysis of T-20 Cricket. *Journal of Reliability and Statistical Studies*, 6(1):69–76, 2013.
22. Wei Shen, Jianyong Wang, and Jiawei Han. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Jun 2014.
23. Shiladitya Sinha, Chris Dyer, Kevin Gimpel, and Noah A Smith. Predicting the NFL using Twitter. *Machine Learning and Data Mining for Sports Analytics*, 2013.
24. Veselin Stoyanov, James Mayfield, Tan Xu, Douglas W. Oard, Dawn Lawrie, Tim Oates, and Tim Finin. A Context-aware Approach to Entity Linking. In *Proc. of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 62–67, 2012.
25. Pamela Wicker and Christoph Breuer. Analysis of Problems using Data Mining Techniques – Findings from Sport Clubs in Germany. *European Journal for Sport and Society*, 7(2):131–140, 2010.
26. LI Yingying, Silvia Chiusano, and Vincenzo Delia. Modeling Athlete Performance Using Clustering Techniques. In *The 3rd Intl. Symposium on Electronic Commerce and Security Workshops (ISECS)*, pages 169–171, 2010.
27. Wei Zhang, Yan Chuan Sim, Jian Su, and Chew Lim Tan. Entity Linking with Effective Acronym Expansion, Instance Selection, and Topic Modeling. In *Proc. of the 2011 Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 1909–1914, 2011.
28. Wei Zhang, Jian Su, Chew Lim Tan, and Wen Ting Wang. Entity Linking Leveraging: Automatically generated Annotation. In *Proc. of the* 23rd *Intl. Conf. on Computational Linguistics (COLING)*, pages 1290–1298, 2010.
29. Zhicheng Zheng, Fangtao Li, Minlie Huang, and Xiaoyan Zhu. Learning to Link Entities with Knowledge Base. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 483–491, 2010.