1st May, 2022

Football Match Report Generator using NLG

Samyukt Sriram

Nishant Mahesh

Ashoka University

Contents

1. Introduction

1.1 Motivation

Natural Language Generation has interesting applications in many fields. Our initial vision for this project was to use NLP/ NLG models on event data from a football match, to create entertaining match summaries in natural language. To best paint this picture, consider the following sample we generated at the outset of the project with GPT3:

Prompt to DaVinci:

```
Create a match summary for a football match based on the following data (in
csv format) about the events in the game:
minute play_pattern team type player
9 From Throw In Barcelona Shot Sergi Roberto Carnicer
11 From Corner Real Madrid Shot Mateo Kovačić

...<Data omitted for readability>...

92 From Free Kick Barcelona Shot Jordi Alba Ramos
The tone should be professional, enthusiastic and engaging.
```

Output:

```
In a close match that saw both teams have their share of chances, Real Madrid
ultimately came out on top, beating Barcelona 3-2.

After an early shot from Barcelona's Sergi Roberto Carnicer, Real Madrid
took the lead in the 11th minute when Mateo Kovacic found the back of the
net from a corner.  Barcelona responded soon after, with Lionel Messi scoring
from a free kick in the 17th minute.

The two teams exchanged shots throughout the remainder of the first half,
but it wasn't until the 52nd minute that either team found the back of the
```

```
net again.  Neymar set up Luis Suarez for the goal, which brought Barcelona
level at 2-2.

Barcelona took the lead in the 67th minute when Neymar scored himself, but
Real Madrid responded just a minute later when Marco Asensio found the back
of the net from a throw in.

The two teams continued to exchange shots, but it wasn’t until the 89th minute
that either team found the back of the net again.  Luka Modric set up Sergio
Ramos for the goal, which put Real Madrid ahead for good.
```

Note that the output mimics what one would expect from a post-match summary, but the narrative generated and events mentioned are inconsistent with the initial scoreline. However, the sentences are coherent and structured and there is a progression in time.

Our goal was to improve upon this, by developing an NLG model that accurately understood the context and events in a football game based on structured event data, and generated a report in natural language that would engage and inform the reader about what happened in last night’s game.

1.2 Problem Statement

We restricted our problem statement to what we thought was a good first step towards generating match summaries from events data - exploring ways to generate random, real-sounding match reports by fine-tuning state-of-the art language models using football match report data.

Our end goal was to be able to input a simple prompt to our model such as *"Arsenal scores 4 against United"*, and have it build a real-sounding match report around the prompt.

We split up our project workflow into 3 major phases:

(a) **Exploration:**
This phase will involve exploring and understanding the theory behind various algorithms currently being used for natural language generation tasks.

(b) **Data Gathering:**
In this phase, we will get a better understanding of the type of data that will best suit our task and identify ways to procure and build this data.

(c) **Model Construction and Fine-tuning:**

This phase will involve reproducing some of the NLG methods explored in the exploration phase, and fine-tuning state-of-the-art models using the data we gathered.

2. Exploration

2.1 RNNs: Shakespeare Bot

For the task at hand, we quickly realised that conventional feed-forward neural networks were poorly suited. These models are unable to generate varying output lengths, or take into account the order of input data. For generating language, Recurrent Neural Networks (RNNs) [15] are able to overcome this by keeping certain amounts of information in 'memory.' At a higher-level, this just means that RNNs are able to incorporate more input data from earlier in a sentence when generating a prediction about what the next word or character is likely to be.
There are 3 RNN types:

1. Vanilla RNNs
2. Gated Recurrent Units (GRUs)
3. Long Short-Term Memory layers

The architecture of each unit varies, but the essential gist is that each unit is able to hold some information and add to it as new information is fed, and based on both old and new information, is able to make a prediction.

To start off our hands-on exploration, we followed a basic guide to implementing an RNN to generate long lengths of Shakespearean prose and poetry (***Create Your Own Artificial Shakespeare in 10 Minutes with Natural Language Processing***) [18]. This model uses 3 layers, including 1 GRU layer. The data used is a corpus of Shakespearean poetry and prose that is 1115394 characters long. We were introduced to the pre-processing workflow through this example. For a given list of characters, the model uses characters 0-100 to predict the 101st character. Then it uses 1-101 to predict the 102nd and so on. At a fundamental level, the patterns in the language of the corpus can be modelled and predicted. The output for the model in the article can be referenced. To explore, we ran the model again with an LSTM layer instead of the GRU. We also fed in some of the data we scraped (more details in the Data Gathering) section. The Colab Notebook: [10]. Here are some samples, generating 750 characters:

> **Input phrase:** *"Arsenal defeat Manchester United"*, `num_generate = 750`
> **Output:**

```
Arsenal defeat Manchester United group.  Ofter his clever Haners jumping
from Matt Oels and Leicester is one of the best goalkeeper, that lacks in
```

```
the corresposed the first 10 minutes of the second half.  Leicester managed
to meet as a fourth brot on 26 minutes.  "He was exciting in the previous
five minutes of the restart.  Byliverpool would have embarrassed previously
as being absences.  Sky Sports' Gary Neville:  "It was a really good job.
Alpress Watford appeared bungled for his save.  Saints substitute Jimenez's
min-out of the glar on a good run of form, we've seen the ball slowled to
the football at ped and Watford really not been back against one of the best
teams in the league so it's not a bad soft clearer when he was replaced by
Andriy Yarmolenko.  I think the grass wa
```

We made sure to read up a bit on the loss functions of this model and the optimizer AdamW, just to get a better sense on how these models are trained and evaluated. We can see the results are not particularly impressive, there is very little coherence beyond mentioning football related terms.

2.2 Attention-based Models with HuggingFace Transformers

An issue with the above RNNs is that they have to take every item of the input data, but are unable to weigh certain pieces of information more heavily than others. A better way to model natural language is to use Attention-based Models that can weigh certain key words / phrases more than others. (***Understanding Attention Mechanism: Natural Language Processing***) [1].
Intuitively, this kind of model is able to 'understand' natural language better, atleast in a way that is more human. As we read, certain key words give us more information and context than others, so it makes sense that weighing certain pieces of input data more heavily would lead to better results.
From our research and reading, it became clear to us that this kind of architecture is more advanced and better suited to most NLG tasks than the RNNs we were working with. Moreoever, we found a library called HuggingFace Transformers that allowed us to implement and experiment with a variety of attention-based NLG model architectures. [5].

2.3 Causal Language Modelling with Distil-GPT2

After exploring HuggingFace Transformers' library, we concluded that Causal Language Modelling (CLM) was the closest task for our purpose. CLM is specifically best suited to generate long passages on topics, and is commonly used for tasks like story generation, etc. It is unidirectional, so it predicts the next word solely based on what has come before. This is in contrast to Masked Language Modelling, which 'fills in' the blanks with information before and after the word it needs to predict [8].
We chose to use Distil-GPT2 for the project, which is a lighter version of GPT2 [3]. There are many other

models and architectures that are suitable for the task. One of our suggestions for future work would be to implement some of these models and see what changes.

2.4 Neural Machine Translation: An Alternate Approach

Our focus in this project was primarily to implement a Causal Language Modelling approach to generate long narratives, hopefully about football matches. Another approach could have been similar to the paper ***Generating Football Match Summaries with NMT***[2]. The paper aims to convert an event data point (virtually identical to our initial vision for input data) into a single line of commentary in natural language, using Neural Machine Translation (NMT). While reading this, we also considered approaching an entire game this way. What if we passed an entire game's event data as an input vector, and this corresponded to the match report for the same game?

This is a radically different approach. Instead of generating text, these NMT models would learn how to 'translate' event vector data into natural language. This approach is one that we believe can be explored further. However, assembling a dataset like this would be very demanding. We would need event data similar to ***Statsbomb Open Data*** [17], but find a way to match it to the actual human-generated post match reports and summaries. Also, big picture information like season ambitions for a team would be missing from the event data of a game. Nevertheless, the paper had mixed results for the smaller task of generating single lines of commentary, and it would be an interesting task to implement this kind of model for our goal.

3. Data Gathering

Clearly the most important component for any model to work well is data. While looking for datasets related to football, we found a couple of versatile events datasets such as the ***Statsbomb Open Data*** [17] and the ***Kaggle Football Events Dataset*** [7] which compiled a wide list of match events (such as passes, shots taken, goals scored, successful interceptions etc.) tagged by timestamp during the match, names of players involved etc.

However, the kind of data that we needed was more a repository of match summaries, similar to the ones we'll eventually want our model to generate. While searching for this, we came across a paper titled ***Generating Football Match Summaries with NMT***[2], which scraped match reports from Sky Sports [16] for their model. We decided to do something similar to build our dataset.

3.1 Sky Sports Match Reports

3.1.1 Structure

The Sky Sports football results page consists of match results across a wide range of competitions, most importantly all the major European leagues. We limited our data to just the Premier League.

Although the structure of each report page was not deterministic, most reports consisted of the following information in some order:

(i) A summary of the key events in the match

(ii) Excerpts from both teams' managers' and players' post-match interviews

(iii) Impact of the match's result on the larger context of the season

We decided this is a good place to start and designed a scraper to scrape this data across a series of matches.

3.1.2 Scraping and Formatting

We used an open-source python framework called ***scrapy*** [14] to build our scraper. The scraper that we finally designed [12] scraped the following details from each match report

(i) **Title of the report**

(ii) **Name of home team**

(iii) **Name of away team**

(iv) **Date of match**

(v) **All the text within the report**

We scraped the above data for **200** matches a season across **8 seasons** (2013/14 - 2021/22), giving us a total of **1600** matches and **7179778** characters of match report data to work with [13]. However, not all the seasons had the same format for the text within reports. Later seasons, did not include player ratings in plain text within the document. The significance of this will be fully explained later, but it is important to note that the large dataset was not consistent in its format.
In fact, we often saw much better results with models trained only on the most recent season with 200 matches, as opposed to the entire dataset. The inclusion of player ratings as plain text in the input data made the model generate many outputs of only player ratings that repeated indefinitely.

3.2 IPL 2019 Cricket Commentary Dataset

In addition to the scraped Sky Sports reports, we also made use of the ***IPL 2019 Cricket Commentary Dataset*** [6] to test a hypothesis (that will be explained in section 4).

This dataset consisted of ball-by-ball commentary lines tagged by innings-ID and exact ball number along with the runs scored in each ball for all the matches in the 2019 IPL season.

4. Model Building and Fine-tuning

4.1 Fine-tuning GPT2

4.1.1 Attention-based models theory

We covered the basic idea of attention based models earlier in the paper. Now let us expand a little more on some of the theory and hyperparameters.

GPT-2 is an autoregressive model, which means that it makes predictions based on a finite set of available input data. In this case, for CLM, auto-regressive language generation is built on the assumption that the probability distribution of a word sequence can be expressed as a function of the conditional next word distributions [11]. Mathematically this can be expressed as:

$$P(w_{1:T}|W_0) = \prod_{t=1}^{T} P(w_t|w_{1:t-1}, W_0), \text{ with } w_{1:0} = \emptyset$$

There are a couple of different methods of going from these probabilities to a generated output. The most basic is a greedy search, which outputs the highest probability word for a given sequence. A beam search goes a little further depth-wise, taking into consideration high probability words that may be hidden behind low probability ones.

These often generate predictable outputs, so a good practice is to introduce some random word sequences into the consideration of the model. Methods like top-k and top-p sampling do exactly this, and work by restricting the possible output into a pool of sequences that are above a certain likelihood, and then select from them instead. We have used top-p sampling in our models, but top-k sampling could also be an option. The article ***How to generate text: using different decoding methods for language generation with Transformers*** [11] does a fantastic job of describing all these methods with visuals. Let's examine the results of this model when trained on the match report data. Here is the notebook for the model: [9].

4.1.2 Results

Here's an example of a 1500 token report that our model generated. We used top-p sampling = 0.95, a repetition penalty of 1.1, temperature = 1, and a maximum length of 1500 tokens. The input prompt is highlighted in bold:

Arsenal beat Manchester United 4-0 at Old Trafford on Sunday as they put a 2 1/2 win over Real Madrid in their last Premier League game, beating Brentford

3 – two games behind Liverpool who remain unbeaten. Brentford’s recent form showed just how vital City are when playing host to either of the Foxes for Thursday Night Football match between Man Utd and Crystal Palace before unleashing some fury towards Antonio Conte with an equaliser that saw them slide back into second place after 18 minutes under pressure from Jose Mourinho following his late intervention against Chelsea which sent Bruno Fernande offside once more four weeks later than any other team but conceded six goals this season since Jan Havertz did not intervene again because he was penalised by referee Thomas Bachmann twice while Cristiano Ronaldo fired home another yellow card (37) during stoppage time action inside Stamford Bridge Park prior to Goodison’ victory today; further evidence now available pointing the Gunners level beyond claims raised regarding VAR results despite concerns about foul play involving players travelling out without incident due to police involvement within 24 hours’. However, although Burnley dominated possession shortly afterwards it came down to De Bruyne picking up both Guaita Beniteze and Marcos Alonso ahead of Pickout having done so via ball slipback pass attempt only midway through half contact resulting in Joao Cancelo being called upon early throughout – Pep Guardiola even admitted defeat had ’no bearing’ on Watford tonight afternoon rather quickly returning what should have been three points clear thanks to Marc Guehi making good saves himself five years ago! Look forward to seeing you next weekend." Manchester were still looking strong going onto opening day proceedings where Fabianski curled past Hugo Lloris near post alone moments earlier claiming Arsenal played "just like" Southampton 10 yards away above Andros Townsend celebrating Pierre Llorentzer leaving unharmed among visitors waiting outside Selhurst Park seconds too soon heading northwards alongside Meslier whose low cross turned around halfway across goalpost then sealed West Ham top spot all night long courtesy of Ramsey firing wide open seven shots deflected right inches higher overall until James Maddinson made nine catches instead turning one handball narrowly missed Mane perfectly faring Gabriel Jesus deep range superbly inviting Kane upstairs however Wolves managed to stay cleanfooted brilliantly levels thereafter whilst Anthony Martial flicked straight underneath Lacazette headshot barely reaching Salah expertfully beneath Mbeumo well enough defending Joelinton correctly steering Ayew nearly helplessently below Forster though The Hornets found safety quite easily nonetheless allowing him to slot close Pukaku high overhead immediately cutting Kieran Tierney full volley directly beside Dubravka

completely unmarked almost instantly causing Saint John surgery wounds respectively, who might very likely be concerned if Vieira has lost touch? He scored Illi Silva free here somehow becoming unable to get rid of Willy Caballero yet there is no denying Saive breaking Maupay's own goalkeeper award winning effort livethrift seeking first ever winner [34]. Saints began brightly coming round corners expecting Coutinho scorers Raul Jimenez cannoned beautifully finding Kevin McArthur square side Mounted firmly endering Vicente Martinez netting Gallagher scrambling header kindly midrange amid Newcastle boss Bryan Dean delivering sweeping curling final touches along front facing Norwich rivals Aston Villa captain Steven Gerrard slotted deeper fingertips nearer Jorginho threatening emphatically forcing Dignit relegation challengers Leeds closer themselves physically albeit concedishing nothing short of survival position according Toews. Not exactly convincing given Eddie Howe continued searching Cresswell roofward palming Shelvey Sterling shot valiantly parried powerfully headed hard left edge Mason Holgate loopingly racing Fred Van Dijk corneral delicately bobbling wildly backwards seemingly hoping Adams would eventually break Fernandez loose unhurt knowing Emerick could also pick Sanchez safe face otherwise none such heroics find life especially worthy considering Moyes deserved praise earned beforehand: Tottenham finally pulled things rolling fast indeed starting 17+1 eight times including Ederson converting Matip Kante sparingously finishing Mateo Kovacic sprawling tenaciously denied Brighton defender Rashica attempting to move Spurs footily save Sergio Rico intact...but Hodgson gave little detail provided Vitaliy Seguin duly dangled Aaron Ramsdale running wild briefly begging Keane stinging Watkins delivery tame chances ultimately feeding Wilfried Zaha realising Dwight Podence couldn't muster clearance elsewhere cut thumping Harry Maguire scroungued Kasper Schmeichel agonised desperately trying to make way for Matt Targett slipping deadhead unconceivably unnoticed instinctively grazing Hakim Bielsa slightly bending Foster cheaply missing Moussa Sissoko woefully lurking unmoved Ivan Toney clutching Gunn moment after chance saved anyway suggesting Tielemans went nowhere undeserved initially sending Jack Butland wrongended swiftly eroding Charlie Taylor guilty verdict confirming Jonjo Shelvey injured ill will turn Sadio Buendia astray booking fellow winger Kaiessiah unlikely ending injury problems badly tipped Leicester striker Jarrod Bowen culpably failing to fire Tom Cleveron flatkick properly warning Ademola Keita forced Soufal bringing Demarai Grayhill sliding awkwardly toothlessly harm

An interesting thing to note is how frequent team names are in the output. We were aware that the model could probably not focus only on 2 teams throughout, and the first example shows why. As soon as a single team is mentioned, the probably that another team should be included becomes very high. But this same logic applies to the second team, so what we get finally is this unnatural sounding commentary where a large number of teams are mentioned with little to no context. The same applies for players, and this can be seen in the last few lines of the above example.

An explanation for the above could be the repetition penalty we've included in the model generation hyperparameters. However, not including a repetition penalty comes with its own issues. The same model, with a shorter max_length = 500 tokens and no repetition penalty:

> **Arsenal beat Manchester United** 3-2 on Sunday and the Gunners' victory lifts them into the top of the Premier League. Arsenal beat Leicester 1-1 at St Mary's and won the contest 12-1, but it was postponed until after the break. Arsenal boss Rudiger: "It was tough for me, especially on the second half. "In the second half, the game changed and I didn't think we had a better chance of getting a result. "But I still thought we were a game that we deserved to play today and we created more chances in the second half." "At the other end, the game changed dramatically and now it's important we make the right decision." It was an early warning for West Ham as Everton manager: "We did well in the first 20 minutes but we still needed to defend our ball more often than we did in the second half. "We didn't give them anything for a few minutes, they didn't try and score. We didn't defend that right away. We had to defend a goal. We took a couple of chances to be honest. When you think of the type of play you play in, you don't win one game at a time. That's the truth. "The game was difficult to control as the game started to change, but it was important to defend and score. That was the goal that we made. It is not easy today in the Premier League."

This sample has a reasonably convincing impression of a manager's post-game thoughts. There is some amount of consistency in the parts of the 'game' being talked about. It looks like no repetition penalty does help generate slightly more coherent narratives. However, we noticed in a fair few samples, the model began repeating sentences indefinitely. This is why we included a repetition penalty for the larger sample.

To continue our exploration, we tried the same model as above, but set temperature = 0.5 instead of = 1.

> **Arsenal beat Manchester United** 3-0 at Old Trafford on Thursday night. The Clarets were the first Premier League side to score in a game since 1954, and the first of their four Premier League games since the end of the 1970s.

```
The Clarets' victory comes just two days after a thrilling 4-0 win at Manchester
United, which was the first of their previous five Premier League games.  United
boss Frank Lampard said:  "We were really pleased with the performance of
the players.  "We had the ball in the box and the goal was the quickest we've
scored in a game.  "The goal was really important.  The players were really
good, we were really pleased with the performance of the players.  "The players
were really happy with the performance of the players.  "The players were
really happy with the performance of the players.  "We were really pleased
with the performance of the players.  "The players were really happy with
the performance of the players.  "The players were really happy with the
performance of the players.  "The players were really happy with the performance
of the players.  "The players were really happy with the performance of the
players.  "The players were really happy with the performance of the players.
"The players were really happy with the performance of the players.  "The
players were really happy with the performance of the players.  "The players
were really happy with the performance of the players.  "The players were
really happy with the performance of the players.  "The players were really
happy with the performance of the players.  "The players were really happy
with the performance of the players.  "The players were really happy with
the performance of the players.  "The players were really happy with the
performance of the players.  "The players were really happy with the performance
of the players.  "The players were really happy with the performance of the
players.  "The players were really happy with the performance of the players.
"The players were really happy with the performance of the players.  "The
players were really happy with the performance of the players.  "The players
were really happy with the performance of the players.  "The players were
really happy with the performance of the players.  "The players were really
happy with the performance of the players.  "The players were really happy
with the performance of the
```

We can see that another element that contributes to the repetitiveness is temperature. A higher temperature squeezes the probabilities of different outputs closer together. A higher temperature output will talk about more different things, while low temperature will tend to sound 'bland' or repeat itself. This is especially true when there is no repetition penalty, like the above example shows.

4.1.3 Criticisms and Analysis of Our Model

Even with tweaks to the hyper-parameters, we found that the output was rarely able to present a section that focused on any one aspect of a game; be it player performances, manager interviews, big-picture narratives, etc. We hypothesised that the reason for these results is that the scraped sky sports data is not as focused on just the summary of a match as we needed. The match summary was only a small part of the entire report, which dealt more with placing each match in the larger context of the season and quotes of players and managers during match interviews.

Given that we built our scraper based on the webpage structure of these match reports, we were forced to blindly scrape the entire text of the report, since there was no rigid structure imposed by the HTML source. Additionally since the model is based on this indiscernible structure, we hypothesised that if our data was more focused on just the summary of each match instead of what it currently was, our model would produce better results, or at-least results that were more focused.

To test out this hypothesis, we began searching for data from other sports that had more focused data. During this search we found the ***IPL 2019 Cricket Commentary Dataset*** [6], which consisted of ball-by-ball commentary lines for the whole of the 2019 IPL season.

4.2 Generating Cricket Commentary Lines

4.2.1 Why we moved to Cricket

The reason we thought this dataset could be helpful was that it consisted of just commentary lines for each ball and nothing else. Our aim was to try and generate similar commentary lines by training on this dataset. The Causal Language Models are best suited to learning one 'kind' of language pattern; the football match reports had many forms of summarization, reporting new information, quotes from managers, player ratings in strange formats, etc. All these made the language a poor input to model with CLM.

The cricket dataset had only commentary, no other kind of language was present. We believed this would be much easier to model and generate, so we gave this dataset a try by configuring the same model to work with the cricket commentary dataset instead.

4.2.2 Results

Here are some of the samples that our model generated. To be consistent we used the same hyper-paramters as we did for the football model.

Sample 1:

Starts off with a boundary slower ball, full outside off, and cut behind square past the fielder at point! Allowing Watson to free his arms before nudging him away again This was an easy one, quick-kicked but powerful extra run. Ishan Kishan the fielder. Lucky break for Chennai Super Kings again another risky diving this time they'll get four more because there's no one backing up. Faf had to put in two flicks as he attempted to duck back down on Mitch

Sample 2:

Starts off with a boundary full and Watson hops back to keep that out length ball, just outside off, punched by Chahar. Angling into his pads, he's pumped! Mumbai batsmen get away quick on this occasion but the umpire says they'll only go for four because there aren't enough left behind in play. Rohit is pumped, Krunal is pumping again. There's life inside Jharkhand who looks set upon himself like some super early hitter? Nah...he could've been walking across from Bumrah earlier or something - it wasn't working for him. Just eludes Krunal at deep point! Oh my God. This was right where Mr Raina had gone wrong! He went down without incident before tripping two more times. Stopped comfortably and nudged Humberto Ishan one of India's few runs. Pressure building. Couldn 'bouncer any better' an extra run against Chennai Super Kings 150 since mid-on

The generated text isn't spaced to reflect the nature of the commentary. At some points in the middle of the lines, commentary on a new ball begins. Keeping this in mind, the model has fairly interesting output. The lines are plausible and hold some level of consistency within themselves, but this is not frequent.

However, let's look at another sample generated, under the exact same hyperparameters as the previous two:

Starts off with a boundary slower ball, full outside off, and worked to the back foot of keeper Watson. Tight first over from Bumrah back of a length, just past Chahar at point! Solid enough for an easy single as he defends this flatter delivery across. Lovely hit, done in by the zip behind square - on deep midwicket (120kph) length ball o middle (60mh+) still had room but fended off some slower balls. Pressure building. There's life inside out there: wide called before IPK started...this was bowled nicely through extra cover full between cover and point, and Gould pulls away once again ;-) Good news is that not one who sees it because his hands are so long gone. Rohit has been placed under Krunal police control, despite no response yet given. Ishan Kishan the fielder after batting too hard? Nah, we can’t put up any further compelling theories about him or what transpired during TSN Dhoni’s late run today. Ian Bishop apologises very much on TV for South Africa needing help batsman Rahul Gandhi unavailable earlier than usual when asked whether they needed medical attention After reviewing all these reviews I believe Thackeray wasn‘t happy with Bowler simply observing sizzling down short third man wintryly into bunt leg stump This isnt working for Mr Raina either, since Chennai Super Kings 150 have held four more meetings which may be indicative merely due to rain time In terms even if Faf looked upon rather disappointingly deformed lbws It appears clear That Pollard believes fast bowlers need regular follow-up care But only until then does changes would’ve got to go beyond standard stuff like THAT AND HARDFULLY MOUTHA SWEER Umpire looks around nervously To get carried away With halfway through their match against West Indies’ Noorai Jadeja drops two deliveries onto deck Atletico Madrid legend Anjem Chahar makes huge strides forward The decision to keep bowling faster takes chances wider where ever Headingley wants you closer to handball Mitch Fuller celebrates three straight runs Ball flies above his head directly towards Quinton Maroishewan Two bouncer pads fill the gap Backed away early; Watto spotted that QdK fan walking backward Oh my God, dear reader. Think your shot selection went wrong Too soon For All These Times columnist Ravi Shankar says "too many" arenś getting fair offers yorker bunted tightly while running low speed ahead of Ravindra Jadeja Onwarden arrives alone Yes, almost nonchalantly, our ’goats guy’." Even though neither side were standing firm immediately let alone brought along such quality swanshee We will never forget them One fine swipe goes right back!’ Replays suggest excellent movement Bytes somehow came within sight of Bravo himself First choice... good feet work together here Crunched through cramped spaces Competing ends direct hits toward home Plateaus gather pace Fastest strike Ever complete via throwaway Drilled through narrow gaps Cut loose near crease Endeavour meets arms round face value Smashes hope lower body part gully passes Deceived twice Inside reach high ground Hardcapped valiant try stumps defended Replay Time Out Wide sweeper opens doors Close range Big watch comes alive 18+ overs Slowpoke tries defend ©

```
2019 World Baseball Classic Stadium Mumbai Indians pick Pakistan 9th place
overall Find reasons why things couldn´t stand out better They’ll beat Bangladesh
5 times More importantly, think Twice You woníu expect something special
going on Your gut reaction suggests different story Couldntees punched halfway
thru second base McClatchy manages to free Rayudana Copyright 2019 Cricket
Ground Bengaluru Menace walks way below level amid loaded crowd Just five
minutes into play McCulloch replays come close Listen closely now, folks
wondering aloud should someone recognise those isolated strikes From underneath
covers looking inward And although none could quite pinpoint exactly how
far Up next front bat Down sharply angled left handed corner Buttling duckles
bobbing upstream Demanding big evening follows carefully drawn flight Pathogen
slams waste of a doubt Overlookable spin lets fly High pitched dive thumping
Bottomful noselift clears bank clean catch Deep cuts leading to sixth spot
Slider soaring overhead porter caught diving mistake Ahead of light switch
Broadcaster appeals for covering weak end Pullout delivers crisp punch Workover
falls flat Great deal Looks amazing!, umpired - aware that lifting six men
easier might arrive If people knew nothing else, imagine yourself confronted
with massive crowdsourcing Huge influx of visitors Every day 1,000 takeovers
Comebacks pass us expecting bigger figures Nobody knows anything besides
ourselves Classics win 3rd XI 7% OFT UNDERWISE 2nd XVIX OVERWHILE WE GET
ASKS WHEN ALL THE PUTS WILL LOOK LIKE JOHNSON BALL ZESMA Hits 400 Million
People Are Waiting Outside FOR BUZZFLASH!!  Massive amount of empty seats
clog open 99 per cent Test worker uncomfortably waits patiently waiting for
real action 0/1
```

This is significantly worse than the earlier samples, although the hyper-parameters used were the same. We encountered these issues because we’ve included a repetition penalty and a generous max length. So either the output has an end early on, or it ends up rambling. What’s worse, the pre-trained GPT2 model has characters and inputs from the general web, and these seem to be appearing in the output.
In fact, if we include a large minimum length and repetition penalty, the model spits out characters that don’t appear in the training data at all. Some of this can already be seen in the above model, but in one case the model had a long list of japanese characters in the output, much to our confusion.

Still, the shorter passages generated by the model are fairly close in language to what one could expect to hear in a cricket match. It’s possible that with little more sophistication and even more fine-tuning, the outputs could definitely be an interesting, albeit short, read for the average cricket enthusiast.

5. Conclusions and Way Forward

This whole project has been a huge learning experience in the field of NLG, and we have learnt so much about what goes into these models and what we can do to affect their quality, output and functioning.
To recap our results, we saw decent sections of output with the football summary generator. But we realised quickly that the input data we had for this task was not well suited to Causal Language Modelling. The output was often unfocused, confused by the variance in the language of our input reports.
To get around this, we tried a more uniform cricket commentary dataset, and saw that the uniformity in language caused a marked improvement in the quality of the output. The cricket commentary generated some engaging and interesting lines that were engaging to a casual reader. From what we've learnt, we can safely say the work we have done so far is just the beginning. Here is a couple of things we could do to extend what we've done:

1. Implementing other Attention-based CLM, like XLNet.

2. Trying to improve the quality of the input data, especially for the football report generation. In the context of the Sky Sports reports, finding ways to identify subsections of the report that deal with just the match summary is something that can be explored.

3. Another approach could be to process and categorize the passages in the report into various types, and run CLM models on each component separately.

4. Exploring similar approaches for data from other sports (eg. for basketball, the ***harvardnlp/boxscore-data*** [4] could be interesting.)

5. Generating entire match summaries using NMT instead of CLM. Instead of long paragraphs with an input prompt, NMT would convert event data vectors into natural language by 'translating' them. [2].

Bibliography

[1] Nikhil Agrawal. *Understanding Attention Mechanism: Natural Language Processing*. URL: https://medium.com/analytics-vidhya/https-medium-com-understanding-attention-mechanism-natural-language-processing-9744ab6aed6a.

[2] Miguel Ayala. *Generating Football Match Summaries with NMT*. URL: https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/custom/15723716.pdf.

[3] *distil-GPT2*. URL: https://huggingface.co/distilgpt2.

[4] *harvardnlp/boxscore-data*. URL: https://github.com/harvardnlp/boxscore-data.

[5] *HuggingFace Transformers*. URL: https://huggingface.co/docs/transformers/index.

[6] *IPL 2019 Commentary Dataset*. URL: https://www.kaggle.com/datasets/saivamshi/ipl-2019-commentary-data?select=ipl2019_matches_final.csv.

[7] *Kaggle Football Events*. URL: https://www.kaggle.com/datasets/secareanualin/football-events.

[8] Prakhar Mishra. *https://towardsdatascience.com/understanding-masked-language-models-mlm-and-causal-language-models-clm-in-nlp-194c15f56a5*. URL: https://towardsdatascience.com/understanding-masked-language-models-mlm-and-causal-language-models-clm-in-nlp-194c15f56a5.

[9] Samyukt Sriram Nishant Mahesh. *Distil-GPT2 for Match Summary / Commentary*. URL: https://github.com/nishant-mahesh/football_summary_generator/blob/main/Models/football_final_model_gpt2.ipynb.

[10] Samyukt Sriram Nishant Mahesh. *Implementation of Shakespeare bot on Football Data*. URL: https://github.com/nishant-mahesh/football_summary_generator/blob/main/Models/LSTM_Shakespeare.ipynb.

[11] Patrick von Platen. *How to generate text: using different decoding methods for language generation with Transformers*. URL: https://huggingface.co/blog/how-to-generate.

[12] Nishant Mahesh Samyukt Sriram. *Custom Skysport Scraper*. URL: https://github.com/nishant-mahesh/football_summary_generator/blob/main/skysports.py.

[13] Nishant Mahesh Samyukt Sriram. *Scraped Sky Sports Data*. URL: https://github.com/nishant-mahesh/football_summary_generator/tree/main/data.

[14] *Scrapy*. URL: https://scrapy.org/.

[15] Alex Sherstinsky. *Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network*. URL: https://arxiv.org/abs/1808.03314.

[16] *Sky Sports Football Results*. URL: https://www.skysports.com/football-results.

[17] *Statsbomb Open Data*. URL: https://github.com/statsbomb/open-data.

[18] Orhan G. Yalçın. *Create Your Own Artificial Shakespeare in 10 Minutes with Natural Language Processing*. URL: https://towardsdatascience.com/create-your-own-artificial-shakespeare-in-10-minutes-with-natural-language-processing-1fde5edc8f28.