Neural Network-Based Generation of Sport Summaries: a Preliminary Study

David Stéphane Belemkoabga[*], Aurélien Bossard[**], Abdallah Essa[*], Christophe Rodrigues[*] and Kévin Sylla[*]

[*]Léonard de Vinci Pôle Universitaire, Research Center
92 916 Paris, La Défense, France,
`firstname.name@devinci.fr`
[**]Laboratoire d'Informatique Avancée de Saint-Denis, Université Paris 8
2 rue de la Liberté, 93526 Saint-Denis, France
`firstname.name@iut.univ-paris8.fr`

Abstract

This paper presents a global summarization method for live sport commentaries for which we have a human-written summary available. This method is based on a neural generative summarizer. The amount of data available for training is limited compared to corpora commonly used by neural summarizers. We propose to help the summarizer to learn from a limited amount of data by limiting the entropy of the input texts. This step is performed by a classification into categories derived by a detailed analysis of the human-written summaries. We show that the filtering helps the summarization system to overcome the lack of resources. However, several improving points have emerged from this preliminary study, that we discuss and plan to implement in future work.

1 Introduction

In this paper, we propose a new approach for sport commentary summarization. This approach is still an ongoing work, and the results presented here are preliminary. Sport commentaries represent an interesting resource, as the live commentaries we work on are associated with a summary written by an expert : the commentator himself. Indeed, the commentator writes a summary at the end of every game. Automatically generating a game summary would release the commentator from a part of his heavy workload and thus would free up his time for more complex and rewarding tasks, such as game in-depth analysis.

Summarizing live sport events commentaries is a challenging task. First of all, they are live written; new commentaries can conflict with or complete former ones. For example, if a soccer player scores two goals, the live commentaries about each goal are relevant information. However, extracting the live commentaries about each goal will not be suffi-cient in order to generate a good summary. It would indeed lead to producing redundant summaries.

Moreover, if you consider a game as an event, it is composed of several subevents. Some of them are deemed important enough to be commented. However, most of those commented subevents are not important enough to appear in a summary. So, live commentaries are mostly made up of noise : in a soccer game, there will be more shots than goals, even more substitutions than goals, which are the most important information of a game. This noise has to be filtered. Therefore, studies have to be carried out about the relevance of an information in the game summarization context.

The style of human-written summaries differs from the one of live commentaries. For all these reasons, statistical extractive summarization models are not relevant for this kind of data. Extractive models consist in fact in extracting relevant sentences from source corpora and put them together in order to build a summary. The difference in style between summaries and commentaries and the noise in commentaries are a substantial hindrance to building live commentaries summaries with extractive models. As for neuronal abstractive models, fast-growing these past few years, they need huge training corpora to be efficient : several hundreds of thousands of documents associated to their summary. However, we can only assemble a corpus that covers five years of a national championship – approximately 1700 game commentaries associated with their summary.

Moreover, neural abstractive summarizers are mostly designed for news summarization. News summarization fits well neural summarizers as neural models can only take a limited number of words for input. Journalists use an inverted pyramid structure, so the most important information is packed in the first paragraph. Taking the first n words as input ensures that a neural model will work only with

147

important information. Live commentaries do not have such a hierarchical structure, and commonly used statistical indicators do not seem to be useful. Therefore, selecting the input of a neural model is a challenge in our application.

In order to overcome these obstacles, we need to train our own abstractive summarization model. We have to reduce the data noise in order to allow the abstractive model to converge quickly with a limited amount of data. We hypothesize that the human summaries variability is low enough to make it possible for the model to learn despite having a small corpus as input.

We propose a global summarization method that aims to lower the input data entropy in order to enhance the automatic summaries quality. This method relies on a an information selection prior to learning in order to shorten the input texts and thus get rid of data that are useless to the summary generation.

The paper is structured as follows: in a first part, we present the related work. In a second part, we introduce our corpus and its features. Then we describe our method, followed by the experiments and the results. We end by discussing the results and exposing our perspectives.

2 Related Work

The automatic generation of sports commentary summaries is, to our knowledge, a subject that is very little discussed in the literature.

We can mainly cite the work of (Zhang et al., 2016) which, from game commentaries available online, generates an extractive summary. The method consists of three main steps: a first step of modeling sentences according to surface clues defined empirically as the sum of the *tf-idf* sentence words, the presence or absence of important words in the sentence such as "red card", "goal"... Then, from a set of reference sentences, a learning step predicts the sentences ROUGE score according to these surface clues. Finally, a last ranking step allows the sentences with the best scores to be incorporated into the summary. According to these authors, however, the approach suffers from several limitations. Since the process is sentence-centered, the summaries generated have a lot of redundancy between sentences. Furthermore, learning tends to penalize short sentences that are sometimes wrongly considered less informative because their direct contribution to the ROUGE score is lower.

On this same task, we can also cite the work of (Bouayad-Agha et al., 2012) whose particularity is to propose a system of generative summary based on the definition of a specialized ontology for soccer games. Thus, from the data extracted from commentaries in an ontology, handwritten rules are triggered in order to rephrase the information and generate a summary. The main limitations of the approach are the need for an exhaustive ontology population (players, teams,...) as well as a generation of stereotypical summaries because they are built from the same rules.

On the same issue but from a very different angle, the work of (Corney et al., 2014) starts from the comments of twitter users during games and produces subjective summaries. For each official commentary related to an event during the game, the supporters' comments on twitter are analyzed on a 4 minutes window, the goal being to extract the most representative tweet of this event from the subjective point of view of the supporters of each team. For that purpose, the tweets are first distributed between the two teams. A user is defined as a supporter of a team if in his comments the team is overrepresented compared to other teams. In a second step, for each team, the most important topics are detected using a variant of the *tf-idf*. Finally, for each team, the most representative tweet of the subjects found is selected, without further processing.

More recently, the work of (Li et al., 2019) presents a model able to produce NBA games summaries. Based only on game statistics, it can generate a summary composed of two parts: a game overall summary and a player centered summary. This model uses latest deep learning techniques with a Wasserstein generative adversarial networks (WGAN) proposed by (Arjovsky et al., 2017). However, despite the model used, the method only generates stereotyped summaries filling a fixed template. For the overall summary, the template is defined as follows:

> On [Date], [Team(A)] made a [Score(A)-Score[B] [learned phrase] [Team(B)].

Except for the learned phrase, all elements in [] are directly assigned from game statistics. The learned model will only affect the learned phrase used to characterize with words to which degree team A wins or loses against team B.

To avoid the pitfall of generating stereotypical summaries, given the difference in style between the summaries and the live commentaries, the high noise level in the commentaries and the low volume of data, we choose to stand out from previous methods by taking advantage of recent advances in neuronal generation. To do so, we approach the problem from the perspective of neuronal generation preceded by a noise reduction step in the source texts to allow the generative model to converge quickly.

3 Live Commentaries Corpus

We extracted all the available Ligue 1 soccer games commentaries and their associated human-written summary from L'Equipe website (`http://www.lequipe.fr`). The archived games with live commentaries cover a five seasons period from 2014 to 2019. Game commentaries are 8000 words long on average and 80 interventions of the commentator. Live commentaries about the game are interspersed with facts about the game players: information about a recent trade, an ongoing goals streak... Manual summaries are 55 words long on average (cf. Figure 1). Therefore, live commentaries are significantly longer than 400 words which is the length commonly used by most neuronal summarization models in order to reduce their complexity (Rush et al., 2015; See et al., 2017).

The complexity of automatic summarization based on such live commentaries is thus far too important, especially given the small amount of documents we can use to learn.

Figure 1 shows the three last minutes of Paris vs Lille commentaries. It displays the noise in these documents. One can especially notice a poll posted by the commentator between 90+1' and 90+2' minutes, and three commentaries that one can consider as noise for the purpose of generating a short summary: extra time announcement and three missed actions. One can also notice the difference in style between the summary and the commentaries.

The specificities of these documents force us to rethink the automatic summarization process by first filtering only relevant information, then generating a summary in order to mimic manual summaries style.

4 Our Model

Automatically summarizing soccer game commentaries presents a major difficulty: standard

90+3 Final whistle
Four days before their away game in Madrid for Champions League, the PSG with Neymar back in (out since October 5th) has played seriously against a diminished LOSC. Very precise, the Argentinians Icardi and Di Maria goals gave Paris the advantage during the first period. The PSG then handled the game.
90+2 The last corner for the LOSC shot by Yusuf Yazici for nothing. Jose Miguel Fonte's header goes way over the goal.
Who inspired you most during this 2-0 PSG's victory at home ? (Mauro Icardi / Angel Di Maria)
90+1 Thiago Silva heads the ball first on Yusuf Yazici's cross.
90 Cavani misses the target
Edinson Cavani misses the third goal. During a counter-attack on the right wing led by Angel Di Maria, Kylian Mbappe lures Jose Miguel Fonte with his off the ball movement. In his back, Edinson Cavani shots a right foot volley. Above the goal.
90 Extra time: 3 minutes

Figure 1: Example of a live game text broadcast and its summary (top commentary, 90+3') from Lequipe.fr website. Our translation in italics.

frequency-based techniques for evaluating the importance of a word or sentence are ineffective due to the source documents specificity: the most important information about a game is often the rarest one – result, expulsions, goals. This, combined with the style of the summary which is radically different from the commentaries, led us to abandon the extractive methods for an abstract method. However, the small number of documents that can be used for learning is very restrictive[1].

---

[1]Even if we had summaries over twenty years, we would have 500 times fewer documents than the CNN/Dailymail Corpora used in most abstractive neural summarization research works.

Information type	%		Information type	%
Result	80		End of w/l streak	14
Championship rank	55		Missed penalty	7
Goal scorer	45		Balanced game	5
Team domination	24		Converted penalty	4
Win/loss streak	22		Injury	3
Efficiency	19		1st game since	3
1st/2nd half quality	18		Player missing	3
Game quality	18		Decisive coaching	3
Ejection	16			

Table 1: Information categories and the percentage of summaries in which they are represented (based on an entire League 1 season)

We hypothesize that, given the small average size of summaries and their low linguistic variability, the decoder part of an encoder/decoder model can learn how to generate a summary using style elements of manual summaries. On the other hand, the relatively large size of the commentaries makes the encoding task more complicated, if not impossible given the small size of the learning corpus.

Therefore, we choose to reduce the entropy of the source texts in order to allow an encoder/decoder to learn how to generate summaries with a restricted corpus size. The simplest idea is to keep in the input commentaries only those that are deemed relevant for the development of a summary, before learning an automatic summary model based on these filtered commentaries. We detail both steps here.

4.1 Sentence Filtering

In order to filter the input sentences, we need to characterize the sentences that carry important information, and those that do not. To this end, we decided to rely on the manual summaries written by the game commentators. We assume that these summaries contain only relevant information.

4.1.1 Manual Corpus Annotation

We analyzed an entire year of League 1 (so 380 pairs of live commentaries / summary) and typed the information found in the summaries. The presence of a particular information within a summary is a sign of its relevance. We therefore identified the types of information, then counted the occurrences of the different types of information and kept the most frequent ones.

This led us to the following list of information categories, summarized in Table 1.

Then, we systematically searched for this information in the live commentaries and annotated

them according to the type of information they conveyed. For example, the figure 1 does not contain a relevant commentary, but the game summary contains important information : Neymar's return, PSG's efficiency ("surgical, the Argentinians..."), Icardi's goal, Di Maria's goal, PSG's victory (resulting from the half-time win and subsequent management). In this summary as in many others, information is implied and derived from other information, which has made the task of defining types of information particularly complex. Thus, we have an annotated corpus to learn to categorize commentaries according to the information they carry.

4.1.2 Categorizing Commentaries

We kept only the 17 most frequent classes, considering that below a certain threshold – empirically set – the frequency of a type of information within the summaries was too low for it to be considered important.

Before proceeding with the sentences classification, we trained a language model (Bengio et al., 2003) (Sundermeyer et al., 2015) on the commentaries corpus in order to take into account the specificities of this particular corpus (specific vocabulary, different style from the general language). This model, represented in figure 2 learns word embeddings thanks to a neural network of bi-LSTM (Graves and Schmidhuber, 2005) units which aims at improving the next-word probability prediction.
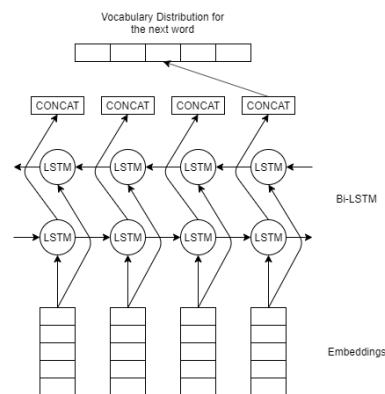

Figure 2: Language model architecture

We then proceeded to train a binary classification model of commentaries on a one-year sample of annotated commentaries. The model used represented in figure 3 is a Bi-LSTM (two successive layers of LSTM, one proceeding from the beginning to the end of the sentence, the other from the end to the beginning of the sentence). This bi-directional architecture allows better results in

language processing tasks. The input layer takes a game commentary and the output layer a binary value. The commentary is classified as relevant if it covers one of the 17 selected categories, otherwise the game is classified as irrelevant.
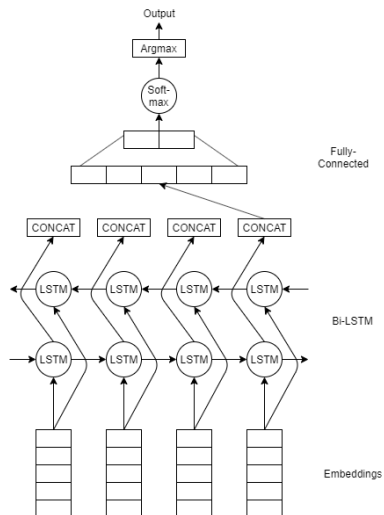

Figure 3: Classification model architecture

We applied this model to all commentaries outside the learning corpus. As a result, we can filter and present to a neural summary model only the commentaries deemed relevant, and thus improve the model response by reducing the entropy of the input data. The results of the binary classification are presented in Table 2.

Method	Recall	Precision	F-measure
Binary model	0.87	0.89	0.87

Table 2: Accuracy of the classification model

4.2 Generative Summary Model

We used a *pointer-generator network* (See et al., 2017). We trained it on two different datasets: a Raw corpus and a filtered corpus (binary classes). *Pointer-generator* is a supervised learning method derived from sequence-to-sequence translation models (Bahdanau et al., 2014) with an attention mechanism (Nallapati et al., 2016).

5 Experiments

To test our approach, we compare two automatic summary models: an extractive method, TextRank (Mihalcea and Tarau, 2004) as well as a generative method, *pointer-generator* on the sample of the last

167 League 1 games that were not used during any learning step.

Experimental Setup : Both methods are tested with and without prior filtering of the sentences judged relevant, according to the method presented in §4.1.2, in order to validate the hypothesis that reducing entropy in source texts has a positive effect on model convergence and on the quality of the summaries produced. The workflows for generative methods with and without pre-filtering are shown in Figures 4 and 5.
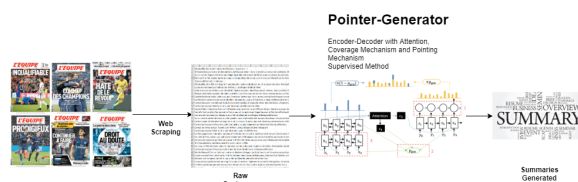

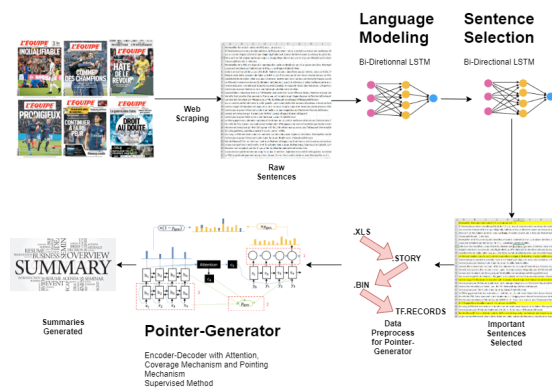Figure 4: Pipeline architecture without classification

Figure 5: Pipeline architecture with classification

The language model uses a word embedding layer of dimension 64. The recurrent cells of the encoder and decoder are of dimension 64. The size of the output layer is the size of the vocabulary, which is 4480.

The classification model uses the same word embedding layer that it retrieves from the language model after training and two layers of LSTM (bi-LSTM) each of size 16, the output layer is of size 2 (0 or 1 for important and unimportant).

The *pointer-generator* model uses an encoder and a decoder with bi-LSTMs of dimension 128. The vocabulary size of the model is 50000. During training, the model takes texts truncated to 400 words and produces summaries of no more than 100 words, which is much more than the number

of words used in human summaries. In order to reduce the size of the problem and speed up training, we have a batch size of 4. Error backpropagation is done with the Adam optimizer with a learning rate of 0.15. The models were learned over 30000 iterations (80 epochs). It takes fewer iterations than the See et al. (2017) model to get results because our training set is much smaller.

Evaluation Metric: Summaries are evaluated with the commonly used ROUGE package (Lin, 2004). The ROUGE-N score is a metric that compares the N-grams in common between the reference summary and the summaries to be evaluated. We took as a reference summary the summaries written by the commentator at the end of the game.

We use the specific configuration that showed the best correlation with human evaluations in Graham (2015) (a ROUGE-2 precision score).

Baselines: We compare the generative method to two extractive methods: one without, and one with filtering commentaries. We use TextRank (Mihalcea and Tarau, 2004), a method comparable to (Radev, 2004) but designed for mono-document summarization. It is a graph-based method that considers summarization as the extraction of the most central sentences in a graph. The implementation used is the one of Nyzam and Bossard (2019), freely available online[2]. Even if TextRank method was designed in 2004, it was shown in (Zheng and Lapata, 2019) that it still compares to more recent methods when there is no correlation between sentence position and centrality.

6 Results

The results are presented in the table 3. We observe a consequent improvement in the ROUGE scores of the extractive and generative models when run on filtered commentaries.

We also find that extractive models are better in recall but less accurate than generative models. The extractive model used here indeed maximizes the number of words in the abstracts, unlike the generative model. As a result, extractive summaries are composed of 69 words on average compared to 44 words for generative summaries, so they can carry more information.

It is also noted that while the generative summaries present relevant information, it is often

Method	Recall	Precision	F-measure
Extractive	3.5	1.7	2.3
Extractive + filtering	3.7	2.4	2.7
Generative	2.5	3.4	2.9
Generative + filtering	2.9	**4.1**	3.3

Table 3: ROUGE-2 Scores of the different summary systems. The best score obtained with Graham (2015)'s configuration is in bold.

> Dans un match nul à domicile pour Nantes qui s'enfonce au score logiquement, Lille se quittent pour la zone de relégation. Le FC Nantes s ' incline pour des Lillois courageux , mais se retrouve le doublé de tableau.
>
> *In a home draw for Nantes that sinks unsurprinsigly in score, Lille are splitting for the relegation zone. FC Nantes loses for brave LOSC players, but finds itself the brace of rankings.*

Figure 6: Example of a summary generated by our generative system (we try to retranscript syntactic errors in our translation in italics)

poorly expressed. Figure 6 shows an example of a summary generated by our system. The following information is common between this summary and the manual summary presented in Figure 7 :

- Nantes and Lille tie up ;
- Lille leaves relegation zone ;
- A brace was scored.

Extractive summaries contain a great deal of redundancy and irrelevant information. Using surface clues other than purely frequency clues, such as (Zhang et al., 2016), would surely partly solve this problem. Figure 8 presents an extractive summary for the same game as the summaries of Figures 6 and 7.

7 Discussion

Our model offers relevant information, but with an approximate linguistic quality. We assume that

> Le FC Nantes et Lille se séparent sur un match nul après une fin de match haletante. Malgré un Sala décisif (doublé), les Canaris pourront nourrir des regrets après avoir mené deux fois au score. Nantes reprend sa 5e place devant Montpellier tandis que Lille sort de la zone rouge.
>
> *FC Nantes and LOSC split in a draw after a thrilling end of game. Despite a decisive Sala (brace), the Canaris will have regrets after having led two times. Nantes takes back its 5th place ahead of Montpellier, while Lille moves out of the relagation zone.*

Figure 7: Example of a manual summary written by L'Equipe's commentator (our manual translation in italics)

---

[2]MOTS : https://github.com/ToolAutomaticSum/MOTS/

Sala signe un doublé ! Thomasson insiste à gauche dans la surface mais se heurte à un Lillois. Lucas Lima hérite du ballon et propose un centre du pied gauche. Totalement libre aux 6 mètres, Sala place sa tête et trompe Maignan sur la droite. Sala attaquant © L'Equipe Grâce à son buteur Sala (9e but en L1 cette saison), le FC Nantes mène très logiquement à la pause tant les Canaris ont été solides et efficaces. But de Sala ! Il centre pour Thomasson qui prolonge vers Sala. Sala attaquant

Sala scores a brace! Thomasson keeps pushing on the left of the box but gets blocked by a LOSC player. Lucas Lima gets the ball and crosses with his left foot. Free at the 6-yard line, Sala heads the ball and beats Maignan on his left side. © L'Equipe Thanks to his scorer Sala (9th goal in L1 this season), the FC Nantes leads unsurprisingly after the first half as they were solid and efficient. Sala scores! He crosses for Thomasson who passes towards Sala. Sala forward

Figure 8: Example of an extractive summary (manual translation in italics)

this lack of linguistic consistency is mainly due to the lack of training data; it prevents the model from capturing language features despite the low linguistic variability displayed in human-written summaries due to document specialization.

As we assumed, the extractive model results in very noisy juxtapositions of commentaries that contain a lot of irrelevant information. The results are therefore far away from human summaries. This is shown on the one hand by reading the summaries produced and on the other hand by the evaluation of the ROUGE score.

Commentary filtering prior to learning does improve the quality of the generated summaries. Learning is simplified by reducing noise and the size of the input data.

We did not perform a manual evaluation of the summaries, because the work is still in its early stages and the time needed for a manual evaluation, e.g. pyramid (Nenkova and Passonneau, 2004) is better spent in late stages. However, we analyzed the automatic summaries produced by our method, and we made the following observations:

- they are close to be grammatically correct;
- even if their ROUGE-2 scores are twice as good as TextRank model, they lack some major information;
- linguistic quality of the results with pre-filtering is far better than without pre-filtering (observation that needs to be confirmed by an accurate evaluation).

This can be due to several causes: first, even if we filtered the commentaries, they are still longer on average than the 400 words commonly used as input of neural summarizers (and used by our summarizer as well). Given the small amount of data

available for training and the fact that information is cut from the input texts, it can explain that major information is missing.

Second, the filtered commentaries are still noisy. Instead of using filtering techniques, information extraction techniques could be used to fill predefined templates for the 17 important information categories we defined in §4.1.1. This would lead to more concise input texts, focusing on the core of each relevant information only.

Third, some information cannot be found in the commentaries. We think of championship ranks, ongoing streaks, which are rarely raised in pregame commentaries, or the overall technical quality of a game, which can be derived from game statistics (percentage of completed passes). However, these statistics can be extracted and given as input of the pointer-generator decoder. This way, the pointer-generator would have access to the information needed to generate sentences conveying what our analysis of the data considered as major information.

8 Conclusion

In this paper we presented a model that allows the generation of abstractive summaries of specialized documents with limited data in French language. Our goal was to show that for summary generation and in specific contexts, abstract models could converge more quickly by reducing the entropy of the input data. Our preliminary results show that after having filtered the input texts and even with a small amount of data, the neural summarizer reaches a much higher precision, and also a better linguistic quality.

We found that much of the information needed to manually generate summaries is not present in the live commentaries. Indeed, many important facts: absence of a player, efficiency, domination of a team, balanced game, are often only deductible from non textual data. Systematically providing this input data to a generative system can help it to improve summary generation. In this way, we plan to add to the text sequences the relevant statistics for the generation of summaries. We also plan to provide more focused and concise texts as input to a neural generative summarizer in order to improve its summaries even with a limited amount of data. We could also improve a language model by using extra texts about soccer games, and thus improve the linguistic quality of the generated summaries.

Acknowledgement

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia. PMLR.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. Cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155.

Nadjet Bouayad-Agha, Gerard Casamayor, Simon Mille, and Leo Wanner. 2012. Perspective-oriented generation of football match summaries: Old tasks, new challenges. *ACM Trans. Speech Lang. Process.*, 9(2):3:1–3:31.

David Corney, Carlos Martín Dancausa, and Ayse Goker. 2014. Two sides to every story: Subjective event summarization of sports events using twitter. In *CEUR Workshop Proceedings*, volume 1198.

Yvette Graham. 2015. Re-evaluating automatic summarization with bleu and 192 shades of rouge. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 128–137.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *NEURAL NETWORKS*, pages 5–6.

Changliang Li, Yixin Su, Ji Qi, and Min Xiao. 2019. Using gan to generate sport news from live game stats. In *Cognitive Computing – ICCC 2019*, pages 102–116, Cham. Springer International Publishing.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Ramesh Nallapati, Bing Xiang, and Bowen Zhou. 2016. Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*.

Valentin Nyzam and Aurélien Bossard. 2019. A modular tool for automatic summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations*, pages 189–194.

Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:2004.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. *CoRR*, abs/1704.04368.

Martin Sundermeyer, Hermann Ney, and Ralf Schlüter. 2015. From feedforward to recurrent lstm neural networks for language modeling. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 23(3):517–529.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. 2016. Towards constructing sports news from live text commentary. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1361–1371, Berlin, Germany. Association for Computational Linguistics.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6236–6247. Association for Computational Linguistics.