# PASS: A Dutch data-to-text system for soccer, targeted towards specific audiences

**Chris van der Lee** and **Emiel Krahmer** and **Sander Wubben**
Tilburg center for Cognition and Communication (TiCC)
Tilburg University
The Netherlands
`{c.vdrlee, e.j.krahmer, s.wubben}@tilburguniversity.edu`

## Abstract

We present PASS, a data-to-text system that generates Dutch soccer reports from match statistics. One of the novel elements of PASS is the fact that the system produces corpus-based texts tailored towards fans of one club or the other, which can most prominently be observed in the tone of voice used in the reports. Furthermore, the system is open source and uses a modular design, which makes it relatively easy for people to add extensions. Human-based evaluation shows that people are generally positive towards PASS in regards to its clarity and fluency, and that the tailoring is accurately recognized in most cases.

## 1 Introduction

For the past few years, news organizations worldwide have begun to show interest in automating various types of news reports. One of the domains that is especially viable for automation is the domain of sports, since the outcomes of most sports matches can be extracted from the data. Additionally, sports statistics (who played, who scored, etcetera) are stored for many games that are neither visited, nor reported on by sports reporters. Automated text generation systems can generate reports for these games.

However, most of the current text generation systems used for journalistic purposes (e.g. Wordsmith[1], Quill[2]) are closed systems that are inaccessible for the general public and for interested researchers. As a result, it is not fully transparent how

---

[1] `https://www.automatedinsights.com/`
[2] `https://www.narrativescience.com/`

these systems work. At the same time, early NLG systems on sports-reporting (André et al., 1988; Robin, 1994; Theune et al., 2001, among others) are also inaccessible because the code for these systems has become obsolete or abandoned. The goal of this paper, therefore, is to present a new data-to-text system, which we call Personalized Automated Soccer texts System (hereafter: PASS). PASS is inspired by earlier NLG research and capable of generating soccer reports from data. The system is open-source and freely available, and set-up in modular way, so that interested researchers can use the system as a testbed for their own, possibly specialized NLG algorithms.

As we argue below, this project is inspired by a previous system and fits with the increased emphasis on replication in science, but this project is more than a straightforward reimplementation. In particular, we show and evaluate how the core system can be used to generate tailored reports for specific audiences.

One of the strengths of data-to-text generation is that texts can easily be tailored towards specific audiences (Gatt and Krahmer, 2017). In order to showcase this strength, PASS produces two texts for fans of each of the teams participating in a soccer match. The difference between these two texts is the tone of voice in the reports. One of the goals in the development of PASS was to generate emotional language that would be expected when people report on an event they are emotionally invested in. In the context of soccer, this means that if the club of the targeted audience loses, the tone of a PASS report would be more disappointed or frustrated and if the

95

club of the targeted audience wins, the tone would be more upbeat. The language of these reports was made to look similar to the reports written by professional journalists by using a corpus-driven approach in the development of the PASS system.

## 2 Related work

Data-to-text systems, systems that "generate texts from non-linguistic data, such as sensor data and event logs" (Reiter, 2007, p. 97), have been around for a long time and still remain a popular topic for Natural Language Generation. Some of the data-to-text language generation tasks that have been investigated recently include weather forecast generation (Belz and Kow, 2010; Angeli et al., 2010; Gkatzia et al., 2016a, among others), medical reports (Gatt et al., 2009; Gkatzia et al., 2016b; Schneider et al., 2013, among others), and financial reports (Nesterenko, 2016, among others).

The domain of sports is also a domain that is investigated quite frequently. This domain is appealing because the content organization could be (partly) fixed for many sports. At the same time, the sports domain is complex enough that it gives rise to many challenges at almost every stage of the data-to-text pipeline (Barzilay and Lapata, 2005). Data-to-text systems in the sports domain can be roughly divided into two categories. The first category is the **commentary** category. Systems in this category produce texts in a style that is similar to the live commentary that can be heard when watching a live sports event. This means that content selection and organization is relatively simple: most, if not all, observable events are covered and this is done in a chronological order. Examples of data-to-text systems that fall into this category are Tanaka-Ishii et al. (1998), Chen and Mooney (2008), and Konstas and Lapata (2012), which all produce soccer reports.

The second, **summary** category could provide a bigger challenge for content selection and organization. These texts are more similar to texts that can be read in newspapers or websites after the sports event and should provide a report on the most interesting elements of the game. This means that content selection is more important and a chronological order is not necessarily used. Examples of systems in this category are Robin (1994), and McKeown et al.

(1995), which produced basketball reports, and Theune et al. (2001), and Barzilay and Lapata (2005), which produced reports on soccer matches.

The current system falls in the latter category. PASS is a data-to-text system that produces Dutch summaries of soccer matches and that uses a template-based approach. Template-based systems can generally be characterized by their slot-filler structure: texts with gaps that can be filled with information. while this approach is sometimes contrasted with "real" NLG, research has shown that template-based approaches generally result in texts of relatively high quality (van Deemter et al., 2005), that are generated relatively quickly (Sanby et al., 2016).

The current project is in line with the ongoing concerns about replication in science in general. However, it is important to stress that PASS is a reimplementation of GoalGetter, not a replication. We aimed to make PASS generate soccer reports somewhat similar to those generated by GoalGetter, but we did not use any of the source code of GoalGetter. Instead, we have built a new system from the ground up, using the description and results of GoalGetter as inspiration, while simultaneously adding new techniques to emphasize the variety of the system's output.

In the last few years, people have become increasingly interested in replicating published research (Ioannidis, 2005; Nosek et al., 2015; Mieskes, 2017, among others). However, in order to replicate previous studies in this field, reimplementation of previous systems is often necessary. Many older systems such as GoalGetter have become abandonware. They are not (any longer) publicly available, their code is obsolete, and sometimes have never been properly evaluated. Reimplementation is required for these older systems before they can be replicated. Therefore, our goal was to develop a system according to modern standards that produces output that is similar to GoalGetter. Furthermore, we have made our implementation publicly available[3], and have performed a human-based evaluation of the system. This makes it possible for others to attempt replication of the current study.

---

[3] https://github.com/TallChris91/PASS

| Type | Match information |
|---|---|
| General | League, date, time, stadium, city, referee, attendees, final score, teams, goal scorers |
| Match events | Assists, regular goals, own goals, penalty goals, penalty misses, yellow cards, red cards (2x yellow), red cards (direct) |
| Last game | League, date, opponent, final score, played home/away, won/tied/lost, changes in lineup |
| Players in lineup, substitutes and managers | Name, full name, nickname, birth date, birth place, height, weight, position, kit number, name in *Goal.com*, *Goal.com* player page, youth clubs, senior clubs, national teams represented, current team |
| Last five games | Opponent, final score, played home/away, won/tied/lost |
| Relative strength | Wins per team for previous meetings, draws in previous meetings, percentage of people predicting win for the home team/win for the away team/tie, date of previous meetings, which team played home/away in previous meetings, final score previous meetings, most predicted results |
| Match statistics | Total shots, shots on target, completed passes, passing accuracy, possession, corners, offsides, fouls, total passes, short passes, long passes, forward/left/right/back passes, percentage of forward passes, blocked shots, shots on the left/right/centre of the goal, percentage of shots outside the 18-yard box, total crosses, successful crosses, crosses accuracy, crosses inside/outside 18-yard box, left crosses, right crosses, total attempted take-ons, successful take-ons, successful left/right/centre/total take-ons in the final third of the match, blocks, interceptions, clearances, recoveries, total tackles, successful tackles, tackle accuracy |

**Table 1:** Information stored from *Goal.com*.

## 3 Data collection

### 3.1 Gathering the data

GoalGetter scraped data from *Teletext*: a system that broadcasts textual data to television and Internet. While this system still exists, the amount of data available on *Teletext* is limited, is not stored, and many sources nowadays offer more data. Therefore, an application was built to automatically scrape soccer match data from *Goal.com*[4], and store this data in XML-format. Similarly to *Teletext*, *Goal.com* contains information about teams that played, final score, goal scorers, referee, attendees and players that were given a yellow or red card. However, *Goal.com* keeps track of a sizable amount of data in addition to this, such as the players that participated in the game, score predictions, the results of previous match-ups between the teams and a sizable amount of detailed statistical information; cf. Table 1. While most of this *Goal.com*-specific data has not been used in the current version of PASS, the availability of this information makes it relatively easy to use this data in future versions.

### 3.2 Designing the templates

With PASS, an attempt was made to produce reports where the tone of voice is emotional, while

the report still appears to be relatively professional. The language in the templates therefore needed to be close to what could be encountered in human-written soccer reports. To achieve this, the templates were derived from sentences in the MeMo FC corpus (Braun et al., 2016). The MeMo FC corpus contains match reports copied directly from the websites of the soccer clubs that participated in the match. These reports are intended for the supporters of their respective club and often contain an emotional tone. These characteristics made the corpus particularly suitable for PASS.

Three steps were undertaken to convert reports in the MeMo FC corpus to templates. The first step was to manually label a sample of reports in the corpus: for each sentence in the sample, we examined what event it described. This first step was done to cluster sentences that described similar events and to get a general idea which categories could be distinguished. Separate databases were made for reports that described a win, a tie or a loss for the team of the website it originates from. The template categories were the same for all these databases, but the templates were different. After the first step followed a reduction step: for every extracted category and sentence we judged if there was *Goal.com*-data available for the information it conveyed and if the information would have been present in GoalGetter.

---

[4] http://www.goal.com

97

| Section | Category | Variants |
|---|---|---|
| Title | | All-purpose, deciding goal, 2+ goal difference, 6+ total goals, no goals, red card targeted team, final goal targeted team, targeted team played away, late equalizer for targeted team, late equalizer other team |
| Introduction | Won/tied/lost | All-purpose, red card targeted team, red card other team |
| | Final score | All-purpose, final goal targeted team, late equalizer other team, focus team tie after being 2+ goals down |
| Game course | Regular goal | All-purpose, goal with assist, deciding goal, deciding goal with assist, 2 goal difference, 2+ goal difference, two successive goals for one team, early goal, goal giving the lead, equalizer, anschlusstreffer, ergebniskosmetik, 2+ goal by a player, final goal, only goal |
| | Own goal | All-purpose, deciding goal, 2 goal difference, 2+ goal difference, early goal, goal giving the lead, equalizer, final goal, only goal |
| | Penalty goal | All-purpose, deciding goal, 2 goal difference, 2+ goal difference, goal giving the lead, equalizer, anschlusstreffer, ergebniskosmetik, 2+ goal by a player, final goal |
| | Penalty miss | All-purpose |
| Debriefing | Yellow card | Multiple in match, one in match, none in match |
| | Red card (2x yellow) | Multiple in match, one in match, second yellow for targeted team |
| | Red card (direct) | All-purpose, early red, red for targeted team |

**Table 2:** Template categories and their variants.

This led to a reduction of roughly half of the categories and sentences; cf. Table 2. This means that a sizable portion of the content found in most human-written reports is not conveyed in the reports generated by PASS. However, the most crucial information about a match was still present after the second step. In the last step, the sentences were converted to templates. This means that the parts in the sentence containing specific information on a match were replaced by empty gaps and information about which type of data should be used to fill in the gap. Sentences were rephrased if this was necessary to make the template applicable to multiple soccer matches. However, these changes were kept to a minimum in order to stay as close to the source material as possible.

The templates used for PASS are somewhat different from the templates used in GoalGetter. Goal-Getter contains less categories and templates, but Theune et al. (2001) ensured variation in the text by using 'syntactic templates'. They made a syntactic structure for each template, so that small changes could automatically be made to the original template if the circumstances required these changes. For instance, the template changed from (1) to (2) if the second goal of a player had to be described.

(1)　　*"<goal scorer> scored **a** goal"*

(2)　　*"<goal scorer> scored **his second goal**"*

We did not add a syntactic system to the templates, but stored templates such as (1) and (2) as separate categories. PASS, contains a larger amount of categories and templates per category, compared to GoalGetter. This makes PASS produce a similar, if not greater amount of variation in the generated reports.

### 3.3 Content selection and document structure

We used a sample of articles from the MeMo FC corpus to get a feeling for the document structure used in human-written soccer reports. We found that a roughly similar document structure like the one in GoalGetter reports is often used for human-written soccer reports. This means that, a four-part division of a soccer report can often be found in human-written reports. These four parts are:

**Title** Usually the result (win/tie/loss) and the final score of the match.

**Introduction** A match preview and the most important results of the match. For example, information about the opponent, expectations about the match difficulty, previous results and current ranking, did the team win/tie/lose, and the final score.

98
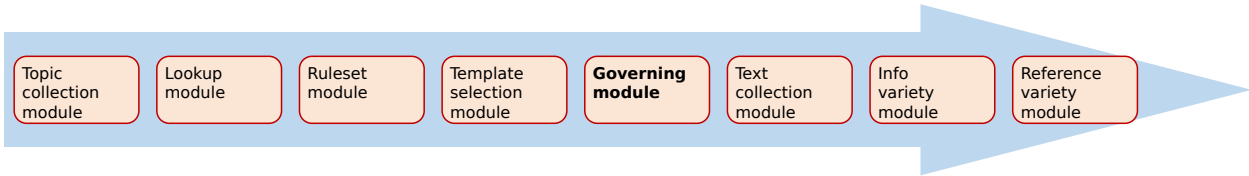
**Figure 1:** Modules used in PASS.

**Game course** A chronological report on the most important events of a match, usually linked together with the subjective evaluations of the writer. For example, a report on the goals, biggest scoring chances, most noteworthy fouls, and which team plays better.

**Debriefing** The consequences of the match and general information about future matches. For example, information on bookings and suspensions, rankings after the match, date of the next match.

Not all commonly found types of information in the MeMo FC corpus were used in PASS reports, since not all information was adequately represented in the *Goal.com*-data and to make the output of PASS more similar to GoalGetter. This means that the introduction-part only expresses win/tie/loss information and the final score. The game course-part focuses on goals and missed penalties, and the debriefing-part merely displays information on bookings. Every part was represented in a separate paragraph.

## 4 PASS system

In this section, we will describe the process PASS takes to go from data to text. The system uses handwritten rules and templates to achieve this goal and produces short reports on a soccer match personalized for each team that played, like the ones in Table 3.

### 4.1 Algorithm

While PASS is similar to GoalGetter in terms of output, the method to achieve this output is different (Theune et al., 2001, for a description of GoalGetter's architecture). The biggest difference is that a modular approach was used in the design of PASS.

By using a modular design, it is easy to make adjustments, improvements and extensions. This means that the modules shown in Figure 1 can easily by replaced by other modules.

PASS starts with the module that governs the generation of the title and introduction. The order in which the topics for this part are reported on is fixed for the current version of the system: title, win/tie/loss information and the final score. The **governing module** will walk through every topic in a stepwise order and interact with all the other modules necessary to generate the text for the introduction-part. We will give an overview of these other modules that it uses for each step.

First, when the governing module starts with a new step, a *unused_topic* will become a *current_topic*. Then, the **lookup module** is activated that opens the template database and retrieves all the template categories and corresponding templates that could be used for the *current_topic*. Part of these template categories can only be used if certain conditions are met, while there is also a general-purpose category containing templates that can be used in every situation.

After a collection has been found of all the template categories corresponding to *current_topic*, the **ruleset module** is activated. This module checks for each template category if the conditions to use said category have been matched. If this is the case, the ruleset module will return `True` to the governing module. If not, it will return `False`. If the governing module receives `True` for a template category, it will add the templates from the category to a list of the *possible_templates*.

If every category has been checked by the ruleset module, the **template selection module** will select a template from the *possible_templates* list in a weighted random fashion. We observed in the MeMo FC corpus that if the right conditions are met, human writers tend to prefer language describing

| | Report for fans of Achilles '29 | Report for fans of Dordrecht |
|---|---|---|
| Dutch | **Thoone velt Dordrecht: 2-1**<br>Jop van Steen en Freek Thoone hebben ervoor gezorgd dat de uitploeg zonder punten achterbleef. In Groesbeek werd voor 1022 toeschouwers met 2-1 gewonnen van Dordrecht.<br><br>De uitploeg kwam na 10 minuten uit het niets op een 0-1 voorsprong door een prachtige treffer van Janga. Jop van Steen schoot in de 48e minuut de dik verdiende gelijkmaker tegen de touwen. Thoone bracht na 88 minuten de winnende treffer op het scorebord: 2-1.<br><br>Scheidsrechter Van den Kerkhof was genoodzaakt 3 gele kaarten te geven, aan Arnaud De Greef, Boy van de Beek en Josimar Lima. | **Het zit Dordrecht niet mee tegen Achilles '29: 2-1**<br>De uitploeg leed een zure nederlaag uit tegen de ploeg van manager Eric Meijers. Dordrecht verloor na een hoopvol begin met 2-1 van Achilles '29.<br><br>Aanvaller Janga zette de ploeg van manager Gérard de Nooijer op een 0-1. Achilles '29 kwam door twee gelukkige treffers van Van Steen en Freek Thoone op een 2-1 voorsprong.<br><br>Er werden 3 gele kaarten uitgedeeld: aan de zijde van Dordrecht voor Arnaud de Greef en Josimar Lima en aan de zijde van de thuisploeg voor Boy van de Beek. |
| English | **Thoone slays Dordrecht: 2-1**<br>Jop van Steen and Freek Thoone have ensured that the away team did not get any points. In Groesbeek, a 2-1 victory against Dordrecht was achieved before 1022 attendees.<br><br>Out of nowhere, the away team got a 0-1 lead because Janga made a beautiful goal after 10 minutes. Jop van Steen shot the well deserved equalizer against the ropes in the 48th minute. Thoone put the winning goal on the score board after 88 minutes: 2-1.<br><br>Referee Van den Kerkhof was forced to give 3 yellow cards, to Arnaud De Greef, Boy van de Beek and Josimar Lima. | **Dordrecht does not have much luck against Achilles '29: 2-1**<br>The away team conceded a sour defeat away against the team of manager Eric Meijers. Dordrecht lost after a promising start with 2-1 against Achilles '29.<br><br>Attacker Jaga gave the team of manager Gérard de Nooijer the 0-1. Achilles '29 got a 2-1 lead by two lucky goals of Van Steen and Freek Thoone.<br><br>3 yellow cards were issued: on the side of Dordrecht to Arnaud de Greef and Josimar Lima and on the side of the home team to Boy van de Beek. |

**Table 3:** Two variants of a match report generated by PASS.

details that apply specifically to the situation, as is shown in (3), rather than language that can be used in every situation, as is shown in (4).

(3)     *"Joachim Andersen made the equalizer directly after the opening goal"*

(4)     *"Joachim Andersen scored the 1-1"*

Therefore, the more conditions were required to be true, the higher the weight we assigned to the template when selecting a template. This increased the chance that a template was selected that was more tailored to the situation at hand, although general-purpose templates still had a decent chance to be selected.

When one template has been selected to convey the *current_topic*, the empty slots in the template need to be filled with the right kind of information. This is done by the **template filler module**.

Every empty slot had been given a tag in the template database (e.g. *<stadium>*, *<referee>*, *<attendees>*). The template filler module uses these tags to find the corresponding piece of information in the match data, then fills the empty slot with this data and returns the filled-in template to the governing module.

The game course and the debriefing were both generated in a largely similar way. With one exception: unlike the introduction, these parts had no fixed order. The topics for the game course and debriefing depended on the match events. For example, a 1-0 result with a missed penalty requires two topics to be reported on in the game course, while a 6-4 result and no missed penalties means ten topics (every goal) in the game course. This meant that an extra module was added, the **topic collection module**. This module extracted the topics from the match data and gave them the right order. Af-

ter the topics were collected and ordered, the exact same modules were used as for the introduction.

After every governing module has produced text for their respective parts, they activate the **text collection module**. This module simply had the task of taking the text for every part and combine them in the right order.

While the system produced reasonable output with the described modules, three more modules were added to increase the variety within and between reports. The **information variety module** ensured that certain types of information in the report would not be repeated. Before the information variety module, certain constructions such as the following could exist:

(5)    *"Ajax obtained the victory before the eyes of **16,673 attendees**. **16,673 attendees** saw the match against AZ end with a 0-3 score."*

Reporting on the attendee information a second time would be redundant in this context. The information variety checks the finished report to see if templates are used with redundant information. If this is the case, the module interacts with the template selection and template filler modules to get an alternative template for the template with redundant information. The information variety module keeps going through the finished report until it cannot find any more redundant information.

Like repetition of information, repetition of references can also have a negative impact on the text quality of the report. This can be observed in the following example:

(6)    *"**Ajax** obtained the victory before the eyes of 16,673 attendees. **Ajax** beat AZ with 0-3."*

The **reference variety module** crawls through the text to spot the same referent in two subsequent sentences. If the module is able to find this, it will use a different form to address the referent in the second sentence (e.g. *Ajax* becomes *the club of manager Peter Bosz*). General-purpose templates have been designed to refer to a person or a soccer team. These templates are picked randomly and the empty slots are then filled in with the template selection and template filler modules, respectively. While this module

works for the current version of PASS, it is possible that the module is too simple for longer, more complicated reports. Therefore, this module will probably be replaced by a probabilistic module as is seen in Ferreira et al. (2016) in future versions of PASS.

Finally, we wanted to demonstrate the variety in outcomes PASS can generate. Therefore, the **between-text variety module** was implemented. This module keeps track of the templates that were used when generating a soccer report. When generating a new report, this module interacts with the template selection module, deleting all templates from the *possible_templates* list if they had been used in the previous soccer report. This ensures that every generated report is completely different from the previous one, thus increasing overall variety.

## 5   Evaluation

We conducted a human-based evaluation to measure the text quality of PASS. For the purpose of the evaluation, a sample was taken of 10 soccer matches played in the Dutch second league in the 2015/2016 season. This means that a total of 20 reports (2 per soccer match) were evaluated by participants. Each participant got to see all 20 reports.

20 Dutch students (13 male, average age 20.6 years) participated in the evaluation. For every match, these participants were asked to answer five questions. The first question was a multiple choice question and served as a manipulation check: 'For fans of which team was the report written: the intended team/the other team'. This question was asked since one of the main functions of PASS is the generation of reports targeted towards fans of each team. After the manipulation check, participants were asked to rate the clarity and fluency of the reports. Clarity refers to how clear and understandable the report is, and was measured using two seven-point Likert-scale questions ('The message of this text is completely clear to me', 'While reading, I immediately understood the text'). Fluency refers to how fluent and easy to read the report is and was also measured using two seven-point Likert-scale questions ('This text is written in proper Dutch', 'This text is easily readable').

An analysis of the manipulation check results showed that people were able to correctly tell to-

wards fans of which team the text was tailored in 91% of all cases. A chi-square test also showed a significant correlation between the intended and perceived tailoring towards fans of the clubs ($\chi^2(1) = 233.33$, $p < .001$). Furthermore, the results showed that participants were overall positive in regards to the clarity and fluency of the reports. The average scores of clarity ($M = 5.64$, $SD = 0.88$) and fluency ($M = 5.36$, $SD = 0.79$) were well above the neutral score of 4.

## 6 Discussion

We have presented a data-to-text system, PASS, that converts data of a soccer match to a textual soccer report. This system was a partial reimplementation of GoalGetter (Theune et al., 2001). Like GoalGetter, a template and rule-based approach was used to design PASS, but there were also several differences between GoalGetter and PASS. For instance, the data source was changed from *Teletext* to *Goal.com*, which provided us with more data. The templates were also constructed in a different fashion. Theune et al. (2001) used syntactically enriched templates, which made a template applicable for several conditions so that more variety in the reports was achieved. PASS uses regular templates, but a corpus-driven approach made it possible to produce a sizable amount of templates and categories, which positively impacted the variety of the PASS reports. However, the biggest change was the implementation of text personalization. GoalGetter generated one 'neutral' report, while PASS generated two reports: one for fans of each club that participated in the match. Personalization was achieved through the use of more 'biased' emotional language as was found in the MeMo FC corpus (Braun et al., 2016). Human-based evaluation showed that this manipulation of the bias in the text was successful. In 91% of all cases, people were able to perceive the tailoring in the intended way. Furthermore, the human-based evaluation showed a positive perception of the text quality in regards to clarity, as well as fluency.

### 6.1 Future work

While GoalGetter was the end result of the research project, the current version of PASS is a first version that will be expanded upon in future research. A simple way of expansion would be to use more template categories and templates and to include more of the available *Goal.com* information in the reports. This is a feasible way to potentially increase the text quality. Additionally, the current version of PASS produces language that could be seen as evaluative (e.g. 'the well deserved equalizer', 'lucky goals'). This evaluative content is currently not backed up by objective data, but can be seen as the subjective view in favor of one side. An interesting future topic would be to explore the usage of these evaluative remarks in connection with statistical data.

Another way of expansion that we are currently investigating is to convert the rule-based content selection and surface realization to a trainable approach. Like most of the template-based data-to-text systems, a sizable amount of manual work was necessary to build PASS. All the templates and rules were written by hand with the specific goal to produce reports for the domain of soccer. This means that the current PASS system cannot easily be adapted to produce reports for other domains. One way of solving this problem would be to build a module that could produce and apply templates with a minimal amount of supervision. Trainable approaches to content selection have been tried previously (Gkatzia, 2016, for an overview). However, most of these approaches only attempt to extract sentences that are aligned with the data. We would also want these sentences to automatically be converted to usable templates, and that these templates could subsequently be applied to produce reports with a minimal amount of rules. To our knowledge, this is a relatively unexplored area of research. The rare study (Kondadadi et al., 2013) that does try to execute all these steps, attempts to produce reports where the topics are always fixed. However, for many domains such as soccer this approach would be problematic, since the topics for these domains could differ greatly as many different events could have taken place. These, and other ideas, are easily explorable with the base that is the current PASS system. The modular design of the system makes all kinds of expansions easily achievable.

# References

Elisabeth André, Gerd Herzog, and Thomas Rist. 1988. On the simultaneous interpretation of real world image sequences and their natural language description: the system soccer. In Yves Kodratoff, editor, *Proceedings of the 8th European Conference on Artificial Intelligence*, page 449–454, London, August. Pitman.

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512, Stroudsburg, PA, October. Association for Computational Linguistics.

Regina Barzilay and Mirella Lapata. 2005. Collective content selection for concept-to-text generation. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 331–338, Stroudsburg, PA, October. Association for Computational Linguistics.

Anja Belz and Eric Kow. 2010. Extracting parallel fragments from comparable corpora for data-to-text generation. In John Kelleher, Brian MacNamee, and Ielka van der Sluis, editors, *Proceedings of the 6th International Natural Language Generation Conference*, pages 167–171, Stroudsburg, PA, July. Association for Computational Linguistics.

Nadine Braun, Martijn Goudbeek, and Emiel Krahmer. 2016. The multilingual affective soccer corpus (masc): compiling a biased parallel corpus on soccer reportage in english, german and dutch. In Amy Isard, Verena Rieser, and Dimitra Gkatzia, editors, *Proceedings of the 9th International Natural Language Generation Conference*, pages 74–78, Stroudsburg, PA, September. Association for Computational Linguistics.

David Chen and Raymond Mooney. 2008. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th International Conference on Machine Learning*, pages 128–135, Cambridge, MA, July. ACM.

Thiago Castro Ferreira, Emiel Krahmer, and Sander Wubben. 2016. Towards more variation in text generation: developing and evaluating variation models for choice of referential form. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 568–577, Stroudsburg, PA, August. Association for Computational Linguistics.

Albert Gatt and Emiel Krahmer. 2017. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *arXiv preprint arXiv:1703.09902*, pages 1–111.

Albert Gatt, Francois Portet, Ehud Reiter, Jim Hunter, Saad Mahamood, Wendy Moncur, and Somayajulu Sripada. 2009. From data to text in the neonatal intensive care unit: using nlg technology for decision support and information management. *AI Communications*, 22(3):153–186.

Dimitra Gkatzia, Oliver Lemon, and Verena Rieser. 2016a. Natural language generation enhances human decision-making with uncertain information. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 264–268, Stroudsburg, PA, August. Association for Computational Linguistics.

Dimitra Gkatzia, Verena Rieser, and Oliver Lemon. 2016b. How to talk to strangers: generating medical reports for first-time users. In *Proceedings of the 2016 IEEE International Conference on Fuzzy Systems*, pages 579–586, Redhook, NY, July. Curran Associates, Inc.

Dimitra Gkatzia. 2016. Content selection in data-to-text systems: A survey. *arXiv preprint arXiv:1610.08375*.

John Ioannidis. 2005. Why most published research findings are false. *PLoS Medicine*, 2(8):696–701.

Ravi Kondadadi, Blake Howald, and Frank Schilder. 2013. A statistical nlg framework for aggregated planning and realization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1406–1415, Stroudsburg, PA, August. Association for Computational Linguistics.

Ioannis Konstas and Mirella Lapata. 2012. Unsupervised concept-to-text generation with hypergraphs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, page 752–761, Stroudsburg, PA, June. Association for Computational Linguistics.

Kathleen McKeown, Jacques Robin, and Karen Kukich. 1995. Generating concise natural language summaries. *Information Processing & Management*, 31(5):703–733.

Margot Mieskes. 2017. A quantitative study of data in the nlp community. In *Proceedings of the First Workshop on Ethics in Natural Language Processing*, pages 1–7, Stroudsburg, PA, April. Association for Computational Linguistics.

Liubov Nesterenko. 2016. Building a system for stock news generation in russian. In Amy Isard, Verena Rieser, and Dimitra Gkatzia, editors, *Proceedings of the 9th International Natural Language Generation Conference*, Stroudsburg, PA, September. Association for Computational Linguistics.

Brian Nosek, Alexander Aarts, Christopher Anderson, Joanna Anderson, and Heather Barry Kappes. 2015. Estimating the reproducibility of psychological science. *Science*, 349(6251).

Ehud Reiter. 2007. An architecture for data-to-text systems. In Stephan Busemann, editor, *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 97–104, Stroudsburg, PA, June. Association for Computational Linguistics.

Jacques Robin. 1994. *Revision-based Generation of Natural Language Summaries Providing Historical Background*. Ph.D. thesis, Columbia University.

Lauren Sanby, Ion Todd, and Maria Keet. 2016. Comparing the template-based approach to gf: the case of afrikaans. In Amy Isard, Verena Rieser, and Dimitra Gkatzia, editors, *Proceedings of the 9th International Natural Language Generation Conference*, Stroudsburg, PA, September. Association for Computational Linguistics.

Anne Schneider, Alasdair Mort, Chris Mellish, Ehud Reiter, Phil Wilson, and Pierre-Luc Vaudry. 2013. Mime-nlg in pre-hospital care. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 152–156, Stroudsburg, PA, August. Association for Computational Linguistics.

Kumiko Tanaka-Ishii, Kôiti Hasida, and Itsuki Noda. 1998. Reactive content selection in the generation of real-time soccer commentary. In *Proceedings of the 17th International Conference on Computational linguistics*, pages 1282–1288, Stroudsburg, PA, August. Association for Computational Linguistics.

Mariët Theune, Esther Klabbers, Jan-Roelof de Pijper, Emiel Krahmer, and Jan Odijk. 2001. From data to speech: a general approach. *Natural Language Engineering*, 7(1):47–86.

Kees van Deemter, Emiel Krahmer, and Mariët Theune. 2005. Real vs. template-based natural language generation. *Computational Linguistics*, 31(1):15–23.