

Finding middle ground? Multi-objective Natural Language Generation from time-series data

Dimitra Gkatzia, Helen Hastie, and Oliver Lemon

School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh

{dg106, h.hastie, o.lemon}@hw.ac.uk

Abstract

A Natural Language Generation (NLG) system is able to generate text from non-linguistic data, ideally personalising the content to a user's specific needs. In some cases, however, there are multiple stakeholders with their own individual goals, needs and preferences. In this paper, we explore the feasibility of combining the preferences of two different user groups, lecturers and students, when generating summaries in the context of student feedback generation. The preferences of each user group are modelled as a multivariate optimisation function, therefore the task of generation is seen as a multi-objective (MO) optimisation task, where the two functions are combined into one. This initial study shows that treating the preferences of each user group equally smooths the weights of the MO function, in a way that preferred content of the user groups is not presented in the generated summary.

1 Introduction

Summarisation of time-series data refers to the task of automatically generating summaries from attributes whose values change over time. Content selection is the task of choosing what to say, i.e. what information to be included in a report (Reiter and Dale, 2000). Here, we consider the task of automatically generating feedback summaries for students describing their performance during the lab of a computer science module over the semester. This work is motivated by the fact that different user groups have different preferences of the content that should be conveyed in a summary, as shown by Gkatzia et al. (2013).

Various factors can influence students' learning, such as difficulty of the material (Person et al., 1995), workload (Craig et al., 2004), attendance

in lectures (Ames, 1992) etc. These factors change over time and can be interdependent. The different stakeholders (i.e. lecturers and students) have different perceptions regarding what constitutes good feedback. Therefore, when generating feedback, we should take into account all preferences in order to be able to produce feedback summaries that are acceptable by both user groups.

Stakeholders often have conflicting goals, needs and preferences, for example managers with employees or doctors with patients and relatives. In our data, for instance, lecturers tend to comment on the hours that a student studied, whereas the students disprefer this content. Generating the same summary for both groups allows for meaningful further discussion with common ground.

Previous work on NLG systems that address more than one user group use different versions of a system for each different user group (Gatt et al., 2009) or make use of User Models (Janarthanam and Lemon, 2010; Thompson et al., 2004; Zukerman and Litman, 2001). Here, we explore a method that adapts to both expert preferences and users simultaneously (i.e. lecturer and students preferences), by applying Multi-Objective optimisation (MOO). MOO can be applied to situations where optimal decisions are sought in the presence of trade-offs between conflicting objectives (Chankong and Haimes, 1983). We explore whether balancing the preferences of two user groups can result in an adaptive system that is acceptable by all users. At the same time, the programming effort is reduced as only one system needs to be developed. Moreover, by pooling all available data together, there is less need for an extensive data collection.

In the next section, we present three systems: one tuned for lecturers, one for students, and one that attempts to find middle ground. In Section 3, we describe an evaluation of these three systems and in Section 4 we discuss the results. Finally, in

Section 5, directions for future work are discussed.

2 Methodology

Reinforcement Learning (RL) is a machine learning technique that defines how an agent learns to take optimal sequences of actions so as to maximize a cumulative reward (Sutton and Barto, 1998). Here we extend the framework proposed by Gkatzia et al. (2013) whereby the content selection is seen as a Markov Decision problem and the goal of the agent is to learn to take the sequence of actions that leads to optimal content selection. A Temporal Difference learning method (Sutton and Barto, 1998) was used to train an agent for content selection. Firstly, we will describe the data in general. Secondly, we refer to the RL system that adapts to lecturers' preferences as described by Gkatzia et al. (2013). Thirdly, we will describe how we collected data and developed a methodology that adapts to students' preferences and finally how we combined the knowledge of both steps to develop an MO system. The three systems (Lecturer-adapted, Student-adapted, MO) share the same architecture but the difference lies in the reward functions used for training.

2.1 The Data

For this study, the dataset described by Gkatzia et al. (2013) was used. Table 1 shows an example of this dataset that describes a student's learning habits and a corresponding feedback summary provided by a lecturer. The dataset is composed of 37 similar instances. Each instance consists of time-series information about the student's learning routine and the selected templates that lecturers used to provide feedback to this student. A template is a quadruple consisting of an *id*, a *factor* (Table 1), a *reference type* (trend, weeks, average, other) and surface text. For instance, a template can be (1, marks, trend, 'Your marks were <trend>over the semester'). The lexical choice for <trend>(i.e. increasing or decreasing) depends on the values of time-series data. There is a direct mapping between the values of factor and reference type and the surface text. The time-series attributes are listed in Table 1 (bottom left).

2.2 Time-series summarisation systems

Actions and states: The state consists of the time-series data and the selected templates. In order to explore the state space the agent selects a time-series attribute (e.g. marks, deadlines etc.) and

then decides whether to talk about it or not. The states and actions are similar for all systems.

Lecturer-adapted reward function

The reward function is derived from analysis with linear regression of the provided dataset and is the following cumulative multivariate function:

$$Reward_{LECT} = a + \sum_{i=1}^n b_i * x_i + c * length$$

where $X = \{x_1, x_2, \dots, x_n\}$ is the vector of combinations of the data trends observed in the time-series data and a particular reference type of the factor. The value of x_i is given by the function:

$$x_i = \begin{cases} 1, & \text{if the combination of a factor trend} \\ & \text{and a particular reference type is} \\ & \text{included in the feedback} \\ 0, & \text{if not.} \end{cases}$$

The coefficients represent the preference level of a factor to be selected and how to be conveyed in the summary. Important factors are associated with high positive coefficients and the unimportant ones with negative coefficients. In the training phase, the agent selects a factor and then decides whether to talk about it or not. If it decides to refer to a factor, the selection of the template is performed deterministically, i.e. it selects the template that results in higher reward. Length represents the number of factors selected for generation.

Student-adapted reward function

The Student-adapted system uses the same RL algorithm as the Lecturer-adapted one. The difference lies in the reward function. The reward function used for training is of a similar style as the Lecturer-adapted reward function. This function was derived by manipulating the student ratings in a previous experiment and estimating the weights using linear regression in a similar way as Walker et al. (1997) and Rieser et al. (2010).

Multi-objective function

The function used for the multi-objective method is derived by weighting the sum of the individual reward functions.

$$R_{MO} = 0.5 * R_{LECT} + 0.5 * R_{STUDENT}$$

To reduce the confounding variables, we kept the ordering of content in all systems the same.

3 Evaluation

The output of the above-mentioned three systems were evaluated both in simulation and with real

factors	week 2	week 3	...	week 10
marks	5	4	...	5
hours_studied	1	2	...	3
...

factors	factor trend
(1) marks	trend_other
(2) hours_studied	trend_increasing
(3) understandability	trend_decreasing
(4) difficulty	trend_decreasing
(5) deadlines	trend_increasing
(6) health_issues	trend_other
(7) personal_issues	trend_decreasing
(8) lectures_attended	trend_other
(9) revision	trend_decreasing

Summary

Your overall performance **was excellent** during the semester. Keep up the good work and maybe try some more challenging exercises. Your attendance was **varying** over the semester. Have a think about how to use time in lectures to improve your understanding of the material. You spent **2 hours studying the lecture material on average**. You should dedicate more time to study. You seem to find the material **easier to understand compared to the beginning of the semester**. Keep up the good work! You revised **part of** the learning material. Have a think about whether revising has improved your performance.

Table 1: Top left: example of the time-series raw data for feedback generation. Bottom left: example of described trends. Right box: a target summary generated by an expert (bold signifies the chosen content).

users. Example summaries of all systems are presented in Table 2.

3.1 Evaluation in Simulation

26 summaries were produced by each system. The output of each system was evaluated with the three reward functions. Table 3 shows the results.

As expected, all systems score highly when evaluated with the reward function for which they were trained, with the second highest reward scored from the MO function. Table 2 illustrates this with the MO Policy clearly between the other two policies. Moreover, the MO function reduces the variability between summaries as is also reflected in the standard deviation given in Table 3.

We used BLEU (4-grams) (Papineni et al., 2002) to measure the similarities between the feedback summaries generated by the three systems. BLEU score is between 0-1 with values closer to 1 indicating texts are more similar. Our results demonstrate that the summaries generated by the three systems are quite different (BLEU score between 0.33 and 0.36). This shows that the framework presented here is capable of producing quite different summaries based on the various reward functions.

3.2 Evaluation with real users

The goal of the evaluation is to determine whether the end-user can pick up on the above-mentioned

differences in the feedback and rank them according to their preferences. The output of the three systems was ranked by 19 lecturers and 48 first-year Computer Science students. Time-series data of three students were presented on graphs to each participant. They were also shown 3 feedback summaries and they were asked to rank them in terms of preference.

As we can see from Table 4, the two user groups significantly preferred the output of the system which was trained for their preferences (Mann-Whitney U test, $p < 0.05$). Interestingly, lecturers found both the outputs produced by the Lecturer-adapted system and the Student-adapted system significantly preferable ($p < 0.05$) to the output produced by the MO system. In contrast, students significantly preferred the output generated by the Student-adapted system over the other two. Finally, both user groups rated the MO system 3rd, but there is not a significant difference between the student ratings for the MO system and the Lecturer-adapted system.

4 Discussion

It is interesting to examine the weights derived from the multiple-linear regression to determine the preferences of the different user groups. For instance, lecturers' most preferred content is hours_studied, therefore the reward function gives high scores to summaries that mention the hours

Lecturer-adapted	Student-adapted	Multi-objective
Make sure you revise the learning material and try to do the lab exercises again. You dedicated more time studying the lecture material in the beginning of the semester compared to the end of the semester. Have a think about what is preventing you from studying. Your understanding of the material could be improved. Try going over the teaching material again. You have had other deadlines during weeks 5, 6, 8, 9 and 10. You may want to plan your studying and work ahead. You did not face any health problems during the semester.	You found the lab exercises very challenging . Make sure that you have understood the taught material and don't hesitate to ask for clarification. You dedicated more time studying the lecture material in the beginning of the semester compared to the end of the semester. Have a think about what is preventing you from studying. Your understanding of the material could be improved. Try going over the teaching material again. Revising material during the semester will improve your performance in the lab.	Your attendance was varying over the semester. Have a think about how to use time in lectures to improve your understanding of the material. You found the lab exercises very challenging. Make sure that you have understood the taught material and don't hesitate to ask for clarification. You dedicated more time studying the lecture material in the beginning of the semester compared to the end of the semester. Have a think about what is preventing you from studying. You did not face any health problems during the semester. You revised part of the learning material. Have a think whether revising has improved your performance.

Table 2: Example outputs from the three different systems (bold signifies the chosen content).

Time-Series Summarisation Systems	Lecturer Function	Student Function	MO Function
Lecturer-adapted system	243.82 (70.35)	51.99 (89.87)	114.12 (49.58)
Student-adapted system	72.54 (106.97)	213.75 (59.45)	127.76 (52.09)
MO system	123.67 (72.66)	153.79 (56.61)	164.84 (83.89)

Table 3: Average rewards (and standard deviation) assigned to summaries produced by the 3 systems. Bold signifies higher reward.

Summarisation Systems	Lecturer's Rating	Student's Rating
Lecturer-adapted	1st (2.15)*	3rd (1.97)
Student-adapted	1st (2.01)*	1st* (2.22)
MO	2nd, 3rd (1.81)	3rd (1.79)

Table 4: Mode of the ratings for each user group (*Mann-Whitney U test, $p < 0.05$, when comparing each system to the MO system).

that a student studied in all cases (i.e. when the hours_studied increased, decreased, or remained stable). This, however, does not factor heavily into the student's reward function.

Secondly, lecturers find it useful to give some advice to students who faced personal issues during the semester, such as advising them to talk to their mentor. Students, on the other hand, like reading about personal issues only when the number of issues they faced was increasing over the semester, perhaps as this is the only trend that may affect their performance. Students seem to mostly prefer a feedback summary that mentions the understandability of the material when it increases which is positive feedback. Finally, the only factor that both groups agree on is that health issues is

negatively weighted and therefore not mentioned.

The MO reward function attempts to balance the preferences of the two user groups. Therefore, for this function, the coefficient for mentioning health issues is also negative, however the other coefficients are smoothed providing neither strong negative or positive coefficients. This means that there is less variability (see Table 3) but that perhaps this function meets neither group's criteria.

5 Conclusion and Future Work

In conclusion, we presented a framework for developing and evaluating various reward functions for time-series summarisation of feedback. This framework has been validated in that both simulation and subjective studies show that each group does indeed prefer feedback generated using a highly tuned reward function, with lecturers being slightly more open to variation. Further investigation is required as to whether it is indeed possible to find middle ground between these two groups. Choices for one group may be negatively rated by the other and it might not be possible to find middle ground but it is worth investigating further other methods of reward function derivation using stronger feature selection methods, such as Principal Component Analysis.

References

- Carole Ames. 1992. Classrooms: Goals, structures, and student motivation. *Journal of Educational Psychology*, 84(3):p261–71.
- Chankong and Haimes. 1983. Multiobjective decision making theory and methodology. In *New York: Elsevier Science Publishing*.
- Scotty D. Craig, Arthur C. Graesser, Jeremiah Sullins, and Barry Gholson. 2004. Affect and learning: an exploratory look into the role of affect in learning with autotutor. In *Journal of Educational Media*, 29:241-250.
- Albert Gatt, Francois Portet, Ehud Reiter, James 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. In *Journal of AI Communications*, 22:153-186.
- Dimitra Gkatzia, Helen Hastie, Srinivasan Janarthanam, and Oliver Lemon. 2013. Generating student feedback from time-series data using Reinforcement Learning. In *14th European Workshop in Natural Language Generation*.
- Srinivasan Janarthanam and Oliver Lemon. 2010. Adaptive referring expression generation in spoken dialogue systems: Evaluation with real users. In *11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*.
- K Papineni, S Roukos, T. Ward, and W. J. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *40th Annual meeting of the Association for Computational Linguistics*.
- Natalie K. Person, Roger J. Kreuz, Rolf A. Zwaan, and Arthur C. Graesser. 1995. Pragmatics and pedagogy: Conversational rules and politeness strategies may inhibit effective tutoring. In *Journal of Cognition and Instruction*, 13(2):161-188.
- Ehud Reiter and Robert Dale. 2000. Building natural language generation systems. In *Cambridge University Press*.
- Verena Rieser, Oliver Lemon, and Xingkun Liu. 2010. Optimising information presentation for spoken dialogue systems. In *48th Annual Meeting of the Association for Computational Linguistics*.
- Richard Sutton and Andrew Barto. 1998. Reinforcement learning. In *MIT Press*.
- Cynthia A. Thompson, Mehmet H. Goker, and Pat Langley. 2004. A personalised system for conversational recommendations. In *Journal of Artificial Intelligence Research* 21, 333-428.
- Marilyn Walker, Diane Litman, Candace Kamm, and Alicia Abella. 1997. PARADISE: A framework for evaluating spoken dialogue agents. In *35th Annual meeting of the Association for Computational Linguistics*.
- Ingrid Zukerman and Diane Litman. 2001. Natural language processing and user modeling: Synergies and limitations. In *User Modeling and User-Adapted Interaction*, 11(1-2), 129-158.