IntelLanG 2020

Proceedings of the Workshop on Intelligent Information Processing and Natural Language Generation

7 September, 2020
Santiago de Compopstela
Spain

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 978-1-952148-26-2

Introduction

We are pleased to present the proceedings of the Intelligent Information Processing and Natural Language Generation workshop (IntelLang 2020), which was held as part of the 24th European Conference on Artificial Intelligence (ECAI 2020) organized in Santiago de Compostela as a fully digital conference. The workshop was organized with the cooperation of the Spanish Network of Excellence on Intelligent Data Processing and Natural Language Generation and endorsed by SIGGEN, the ACL Special Interest Group in Natural Language Generation.

Natural Language Generation (NLG) studies systems for the automatic creation of text from non-linguistic information. The transformation of input data into the final output text involves a number of steps, each involving non-trivial choices. This is arguably the case also in neural NLG systems, where such choices are made implicitly in the case of end-to-end models, or can be handled by dedicated modules.

The use of intelligent data and information processing techniques can help in many relevant aspects of the NLG problem, for example in the contribution of formalisms for knowledge modeling and management, KDD, Data Mining and Machine Learning techniques and tools for the analysis of data, or in the development of models for the evaluation of the quality of the proposals, among many others. Artificial Intelligence information processing techniques can also gain a lot from their interaction with the particular area of NLG, as is the case with the explainable artificial intelligence research field.

The aim of this first edition of the IntelLang workshop was to identify challenges and to value current results that arise from the interaction of intelligent information processing techniques and research in natural language generation, both at the level of models and applications. The workshop provided a forum for discussion of these new research directions, with the stage set by invited talk by Professor Kees van Deemter, followed by the presentation of four long papers and four short papers which are collected in these proceedings.

Kees van Deemter's invited talk took as its starting point the notion of "theoretical NLG", that is, the study of NLG as a window into language generation procedures and models which can shed light on broader questions related to language and communication. Drawing on research on referring expression generation and research on generating and on the generation of quantified phrases, the talk highlight multiple areas where collaboration between NLG researchers, theoretical linguists, logicians and cognitive scientists can lead to models whose goal is not only to perform adequately in a given domain, but to generate predictions and further our theoretical understanding of the phenomena under consideration.

In "SportSett:Basketball - A robust and maintainable dataset for Natural Language Generation", by Thomson et al., the authors investigate the data requirements for the difficult real-world problem of generating statistic-focused summaries of basketball games. For this, they introduce the Sport-Sett:Basketball database, an easy-to-use resource that allows for researchers to easily query data, from many different dimensions, for output in a variety of formats for different architectures.

In "Automatic Follow-up Question Generation for Asynchronous Interviews", Pooja Rao S B et al propose a follow-up question generation model capable of generating relevant and diverse follow-up questions. This system is based on a 3D virtual interviewing system, Maya, a virtual agent-based interviewing system equipped with verbal interactivity from follow-up question generation.

In "How are you? Introducing stress-based text tailoring", by Balloccu et al, the authors study the impact of stress on reading and interpretation of text and propose a method for tailoring a document by exploiting complexity reduction and affect enforcement. Their research is framed in the context of project

NeuroFAST, which focuses on the socio-psychological forces that could influence eating behaviour.

In "Neural Language Generation for a Turkish Task-Oriented Dialogue System", by Artun Burak Mecik, Volkan Ozer, Batuhan Bilgin, Tuna Cakar, and Seniz Demiry, the goal is to develop a Turkish task-oriented dialogue system that enables users to navigate over a map to obtain information about dinner venues according to their preferences, and make reservations based on received recommendations. This work is the first that proposes the use of a neural generation model in a Turkish conversational system.

In "Analyzing daily behaviours from wearable trackers using linguistic protoforms and fuzzy clustering", by Martinez-Cruz et al, a methodology is proposed for analyzing common activity patterns on the basis of data provided by wearable devices, based on the use of linguistic protoforms inspired by Zadeh's Computing with Words and Perceptions paradigm, and fuzzy clustering. The methodology has been illustrated by means of a case study conducted for 200 days using the Fitbit device, recording HKIs related to duration of sleep stages and heart rate, in order to analyze restlessness patterns during sleep.

In "FitChat: Conversational AI for active aging", Wiratunga et al introduce the FitChat conversational bot, intended to encourage users to improve their physical activities. The approach uses a co-creation methodology to identify effective conversational skills by means of an iterative refinement process. The system was evaluated with seven users.

The paper "Fuzzy Logic for Vagueness Management in Referring Expression Generation" by Marín et al, provides an overview of some of the contributions regarding the use of Fuzzy Logic to referring expression generation. While fuzzy logic can capture the semantics of linguistic terms and expressions, due to the graduality associated with the fulfilment of such terms and expressions by objects, the fulfilment of referring expressions and consequently the referential success with respect to particular objects or sets becomes a matter of degree. A review of proposals for developing measures of referential success is provided, with special emphasis on those based on specificity measures.

The contribution "Iterative Neural Scoring of Validated Insight Candidates" by Susaiyah et al deals with the problem of providing comparative insights taking the form of comparative statements about the value of a certain measure in different contexts. This problem is relevant in many applications, for instance in health self-management services based on wearable devices, where statements like "On Weekdays you walk less than on Weekends" must be assessed in terms of their statistical significance, interestingness for the user, and validity for achieving the desired goals, among others. The proposal in this work is to use neural networks and transfer learning algorithms in order to assess statistical significance and, at the same time, to learn user preferences and its changes on time using an online-learning scheme.

In summary, we believe that the workshop attracted a broad spectrum of contributions, emphasising either or both of the workshop's main themes - NLG and Information Processing. Our hope is that these contributions will serve to enhance the sharing of ideas among the two communities.

Finally, we would like to thank everyone who contributed to the success of this workshop, especially the authors, the program committee members, the organizers of the ECAI 2020 conference and the ECAI 2020 workshop chairs.

Daniel Sánchez, Raquel Hervás and Albert Gatt

Granada, Madrid & Malta

September 2020

Organisers:

Daniel Sánchez, Raquel Hervás, Albert Gatt

Program Committee:

Jose M. Alonso, Anya Belz, Alberto Bugarín, Josep Carmona, Luka Eciolaza, Claire Gardent, Albert Gatt, Dimitra Gkatzia, Raquel Hervás, Henrik Leopold, Elena Lloret, Nicolás Marín, Simon Mille, Jose Angel Olivas, Ehud Reiter, Mariano Rico, Daniel Sánchez, Leo Wanner, Slawomir Zadrozny, Sina Zarrieß

Invited Speakers:

Kees van Deemter, Utrecht University, The Netherlands

Invited Talk

Kees van Deemter: Restoring the link between linguistics and computation: the case of quantified expressions

Various people have observed that larges areas of Natural Language Processing have grown further and further apart from the concerns of researchers whose main interest is in language and communication. In this talk I will explore the question of what an optimal collaboration between linguists and computer scientists might look like, in light of the research questions, methods, and tools that both sets of researchers can now offer. The focus of my exploration will be the notion of quantification, which has long been studied by linguists and logicians, but which has not often been the focus of work in modern NLP.

Table of Contents

Analyzing daily behaviours from wearable trackers using linguistic protoforms and fuzzy clustering

Carmen Martinez-Cruz,
Javier Medina Quero,
Macarena Espinilla Estevez
University of Jan
Departament of Computer Science
Campus Las Lagunillas, 23071 Jan
cmcruz, jmquero, mestevez@ujaen.es

Sergio Gramajo
National Technological University
Resistencia,Argentina
sergio@frre.utn.edu.ar

Abstract

The proliferation of low-cost wearable trackers are allowing users to collect daily data from human activity in a non-invasive way and outside of laboratory environments. Exploiting these data properly enable the supervision and counseling from experts remotely; however, extracting key indicators from the long datastreams is hard, often based on statistical metrics or clustering from raw data which lack interpretability. To solve it, we propose an interpretable definition of key indicators by means of linguistic protoforms which include fuzzy temporal processing and fuzzy semantic quantification. Moreover, we use the protoforms defined by experts to evaluate the source datastream in order to provide a straightforward description of the daily activity of users. Finally, the degrees of truth of each protoform are analyzed using a fuzzy clustering method to provide an interpretable description of the long-term user activity. This work includes a case study where data from a user activity (heart beats per minute and sleep stages) have been collected by a Fitbit wearable device and evaluated by the proposed methodology.

1 Introduction

The increase of wearable activity trackers has led to a massive growth in their use in the population (Shih et al., 2015). The use of these devices has proved to increase physical activity between young (Heale et al., 2018) and older (Mercer et al., 2016a) adults and promote a health behavior change (Mercer et al., 2016b). Consequently, wearable activity trackers are the key in new interventions to avoid physical inactivity, which contributes to an estimated 3.2 million deaths each year (Lim et al., 2012).

Among the most relevant data recorded by these devices are heartbeats per second, sleep stages duration time and any sleep disturbance, such as the reduction of the overall sleeping hours or the excessive sleep, that would result in an increase in the warning signs to experts. Poor sleep quality is associated with chronic diseases, weight increase and cognitive dysfunction. The National Sleep Foundation emphasizes following the sleep level targets and guideline and study how these technologies can be useful in this sense, such as the smartwatches. In the near future, this kind of devices may tell us the same as a sleep laboratory. Home monitoring trough smartwatch solutions offers the possibility of sleep coaching interventions or performing analysis to detect any other healthy problems designed by experts (Foundation, 2019).

According to this, smart tracking health devices, such as smartwatches and smartphone APPs, have become increasingly popular. These devices claim to monitor several human activities, and one of the most analyzed is the sleep duration of their users. Most of these devices utilize data generated from in-built sensors to determine sleep parameters. There are many studies that evaluate and compare the accuracy of these sleep tracking devices against more conventional methods used to measure sleep duration and quality (Kang et al., 2017; Kolla et al., 2016). In this way, different commercial smartwatches monitor several aspects of human activity and provide statistical information to users (Bai et al., 2018). The usefulness of this kind of devices were evaluated and validated for monitoring participant sleep levels outside the laboratory environment (Dickinson et al., 2016). Thus, the use of low-cost sleep monitoring devices like Fitbit can help to assess sleep trends where clinical accuracy of laboratory is not necessary (Dickinson et al., 2016) nowadays. Notwithstanding the results indicate that a reasonable degree of sleep staging accuracy can be achieved using a wearable device, which may be of utility in longitudinal studies of sleep habits (Beattie et al., 2017).

Besides, the relative efficacy of different approaches to improve physical activity and sleep using technology-based methods have been examined, although their relatively efficacy to improve these behaviors has not been directly compared (Duncan et al., 2016). In addition to this, some tools have been designed to help the understanding of the sleep quality through contextual information obtained by data from wearables devices (Liang et al., 2016).

In this paper, we present a methodology to analyze the daily activity of users which have been collected by a wearable tracker. A linguistic approach allows to describe the resulting health key indicators (HKIs) and a fuzzy clustering process has been applied to detect some user behaviour patterns. The key points of the proposed methodology are the following:

- A reliable wearable device Fitbit is used to obtain activity tracking data through a bluetooth/wireless connection (Diaz et al., 2015). It provides a HTTPS Web API for accessing data from Fitbit, i.e., automatic activity logs and manually entered records. An application to access and analyze the Fitbit user's data, specially, those related with heart-beats per minute rate and the duration of the sleep activity status (wake, restless, light sleep, REM phase, deep sleep) has been developed here.

- Collected data has been used to define the user most relevant HKIs using protoforms. These protoforms has been designed by the expert knowledge of the supervisor of the user activity. Protoforms summarize the collected information and select the time interval of the day which better suits with the expert criteria using linguistic temporal terms and linguistic quantifiers that provides expressiveness and semantic to the result.

- A fuzzy clustering process is applied to the aggregated truth degree of each day protoform, in order to analyze the common activity patterns for a given user. The suitable relation among the proposed clusters and protoforms enables an interpretable representation of daily activity of users which is meaningful to the supervisor of the user activity.

A review of previous researches related to our work have been included in Section 2. The rest of the paper is organized as follows: in Section 3, we present a methodology to analyze data and represent this knowledge through linguistic protoforms. Some experimental results performed on a dataset from a Fitbit data stream are shown in Section 4. Finally, the conclusion and future work is provided in Section 5.

2 Related works

On the first hand, knowledge-driven methods have been proposed to describe daily human activity by means of sensors (Chen et al., 2011; Medina-Quero et al., 2016). The difficulty of these approaches lies in translating the expert knowledge into a computational method in a flexible, interpretable and rigorous way. Among the wide range of approaches, fuzzy logic (Zadeh, 2006) has been described as an high-interpretable knowledge model for reasoning (Zadeh, 2002) and aggregating (Kacprzyk and Yager, 2001) data under uncertainty. Moreover, the use of protoforms and fuzzy logic (Zadeh, 2002) have provided encouraging results integrating expert knowledge to process sensor data streams in multiple areas, such as, weather forecast (Ramos-Soto et al., 2014), prediction of the urgency demand within smart cities (Medina Quero et al., 2018), providing linguistic summaries from heart rate streams (Peláez-Aguilera et al., 2019) or monitoring of patients with preeclampsia in wearable devices(Espinilla et al., 2017).

On the other hand, data-driven methods have proliferated relating features from sensor streams (Okeyo et al., 2014) to human activity by means of Machine Learning approaches (Minor et al., 2015; Choi et al., 2017). A large majority of these works have focused on supervised learning, where an extensive labelled dataset is required to classify targeted human behaviours (De-La-Hoz-Franco et al., 2018). This requirement faces up with the individual learning of key interest indicators since it is not agile collect and label huge datasets for each person and indicator. In this way, egocentric daily activity recognition (Yan et al., 2015) is mainly supported by non supervised methods, where clustering algorithms (Xu and Tian, 2015) provide the discovering of daily patterns and tasks from users. We highlight the integration of fuzzy approaches with clustering methods, specifically the fuzzy C-means algorithm (Bezdek et al., 1984), which have provided suitable methods to extract meaningful patterns from sensors (Moreno-Cano et al., 2015).

Here, we propose an agile linguistic description of HKIs through the use of linguistic protoforms to pool the advantage of knowledge-drive and automatic learning. These protoforms are computed over the sensor data stream collected by a Fitbit wearable tracker. The daily aggregation of protoforms is subsequently evaluated by fuzzy clustering, which provides the most relevant user behaviour patterns.

3 Methodology

In this section we present a methodology to analyze synchronized source streams from wearable trackers under a semi supervised approach. Firstly, in Section 3.1 we describe the source streams and platform tools to collect activity data. Secondly, in Section 3.2, a linguistic approach based on protoforms is presented to define the HKIs from the source streams. Third, in Section 3.3, the degree of truth of each daily instanced protoform is aggregated and evaluated by a fuzzy clustering process. The resulting clusters expose a linguistic and visual framework to identify the behaviour patterns of an user.

3.1 Collecting source streams from wearable trackers

Fitbit IONIC smartwatch has monitorized and collected sleep stages duration and heart rate activities for an user in a period of time for this study. This device has been chosen because of its many advantages over others models related with the features of: multi-day battery life, accuracy, all sleep stages, accessibility to use datasets in cloud to preprocessing of data, compatibility, data storage and integrated GPS antenna (Bai et al., 2018).

In a formal way, each source stream s^l is represented by a 2-tuple value $\bar{s_i^l} = \{s_i^l, t_i\}$, where s_i^l defines a given value collected by the wearable tracker s^l and t_i its time-stamp. Hence, the long-term information from a given user is composed of a data stream $S^l = \{\bar{s_0^l}, \ldots, \bar{s_i^l}, \ldots, \bar{s_n^l}\}$, which is collected by the wearable device.

For the aim of this work, the two target source streams collected by the wearable tracker Fitbit are:

- s^{as} defines the activity status (AS) by means 5 discrete values: wake (WK), restless (RS), light sleep (LS), REM phase (RP), deep sleep (DS); so $s_i^{as} \in \{WK, RS, LS, RP, DS\}$.
- s^{hr} defines the heart rate (HR) by a natural number which represents the beats per minute (bpm) in the human range; so $s_i^{hr} \in [40, 220]$.

3.2 Protoform for evaluating source streams from wearable trackers

The proposed methodology aims to define the HKIs from Fibit source streams using a set of a protoform instances, which are straightforwardly defined by the supervisor of user activity who has the expertise knowledge in this context.

First, we introduce an ad-hoc *protoform*, to integrate an interpretable and rich-expressive approach that models the expert knowledge in a linguistic way, P_o in the shape of:

$P_o(\bar{s_i^l})$: V_r T_j Q_k

Where:

- V_r defines a fuzzy linguistic term to evaluate the data-stream.
- T_j defines a Fuzzy Temporal Window (FTW) where the term V_r is aggregated. The FTWs are described straightforwardly according to the distance from the current time t^* to a given timestamp t_i as $\Delta t_i = t^* - t_i$ using the membership function $\mu_{T_j(\Delta t_i)}$. The aggregation functions of V_r over T_j for a given $\bar{s_i^l}$ are defined by the following t-norm and t-conorm:

$$V_r \cap T_j(\bar{s_i^l}) = V_r(s_i^l) \cap T_j(\Delta t_i) \in [0, 1]$$

$$V_r \cup T_j(\bar{s_i^l}) = \bigcup_{\bar{s_i^l} \in S^l} V_r \cap T_j(\bar{s_i^l}) \in [0, 1]$$

  where a fuzzy weighted average (Dong and Wong, 1987) (FWA), which is defined in appendix Abbreviations, is proposed to model these functions (Peláez-Aguilera et al., 2019).
- Q_k defines a fuzzy quantifier to evaluate the *intensity* of the linguistic term V_r within the temporal window T_j (Medina-Quero et al., 2016). The quantifier applies a transformation $\mu_{Q_K} : [0, 1] \rightarrow [0, 1]$ to the aggregated degree of $\mu_{Q_K}(V_r \cup T_k(\bar{s_i^l}))$.

This shape of protoform has been successfully developed in summarizing the stream data from health devices (Peláez-Aguilera et al., 2019; Medina-Quero et al., 2016) using a linguistic approach.

In Figure 1, an example of the instantiated protoform *Activity status is deep sleep around 2 and 4*

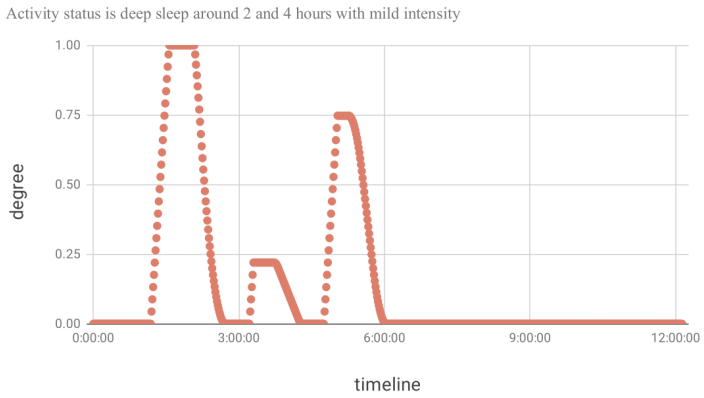


Figure 1: Degree of truth of the protoform *Activity status is deep sleep around 2 and 4 hours WITH mild intensity* that summarizes 12 hours of source stream.

hours with mild intensity is shown, which is computed over a fragment of the source stream from Fitbit.

Moreover, protoform $P_o(\bar{s_i})$ can be combined using fuzzy logical operators to increase the linguistic expressiveness of the model, i.e, negation, union or intersection operators (Peláez-Aguilera et al., 2019) can be included in the final description of the stream. In addition to this, the set of protoform instances can be replaced by a shorter linguistic expressions, much closer to natural language, e.g. *deep restful sleep* that describes the previously analyzed protoform instance *Activity status is deep sleep around 2 and 4 hours with mild intensity*.

3.3 Fuzzy clustering to detect behaviour patterns

In this section, we aim to identify patterns from the users daily activity collected by the wearable tracker. To do that, the membership degree described by protoform instances is analyzed to provide a linguistic description of the most relevant user activity thought of the discovered patterns.

First, we compute the truth degree (P_o^T) of each protoform for each day. To compute it, the aggregation function $\bigcup$ (implemented by the max function) is applied to get the degree of truth of the protoform $P_o(\bar{s_i})$ for each fragment of day T:

$$P_o^T : \bigcup P_o(\bar{s_i}), \bar{s_i^l} \in T$$

Second, the degrees of truth of the protoform instances, that represents the value of an user HKIs in a given day, are evaluated to extract the common patterns in a long term evaluation. To do that, we applied fuzzy clustering using Fuzzy C-means algorithm (Bezdek et al., 1984) over the maximal daily degree of the protoform instances. It is important to notice that the aim of combining these representation of HKIs with fuzzy clustering relies in obtaining a pattern which represent the degree of relevance for each protoform in each cluster. For example, the values of three fuzzy clusters ($K = 3$ in the clustering algorithm) computed from a source stream computed for five protoform instances $P_{1...5}$ defined by the expert criteria, are illustrated in Table 1. The values of these fuzzy clusters correspond to the maximal aggregated degree of each protoform in a day. A bar chart of these results is shown in Figure 2).

Table 1: Degree to which daily protoform instances $P_{1...5}$ belong to each cluster.

Cluster	P_1	P_2	P_3	P_4	P_5
Short description	high HR	low-intensity HR	deep sleep	light sleep	rest nap
C1	0.05	0.02	0.02	0.03	0.15
C2	0.86	0.50	0.54	0.85	0.91
C3	0.63	0.53	0.56	0.90	0.03

4 Case study

In this section we analyze with real data how to describe linguistically the daily behaviour of an user monitored by the wearable tracker Fitbit. The architecture of the system and the process of identifying the patterns are defined as well.

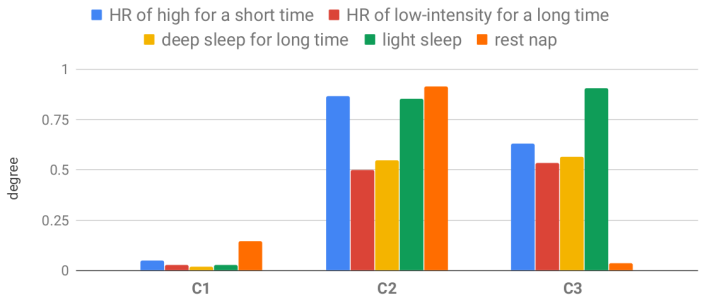


Figure 2: Visual representation of the three clusters obtained from five protoform instances that shows the daily behaviour pattern of an user.

4.1 Data acquisition and processing

This case study has been tested with 200 days data of real activity from an user that wear a Fitbit Ionic device all day. This device has generated a source stream with 700.526 samples with AS, sleep stages and heart rate from the user and sent all these collected activity records to the Cloud.

Thus, to obtain this information, we have developed a web application called *Monwatch* that allows to synchronize and view the information collected by the wearable (and sent by Fitbit Smartphone APP) from the Fitbit Cloud. Monwatch has been developed using the Django Framework version 2 and Python 3.6. All data is synchronized via HTTPS requests according to Fitbit API requirements and the returned data is stored locally in a MySQL Database. To do that Fitbit API requires the use of OAuth 2.0 authorization framework that enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application (our application) to obtain access. OAuth 2.0 is a authentication standard defined in RFC 6749 (D. Hardt, 2019).

4.2 System architecture

The complete scheme of the proposal from the smartwatch real-time monitoring to the gathering and processing of that data is shown in Figure 3. First, the smartwatch collects the raw data and sends it via Bluetooth to the Smartphone using the Fitbit APP which shows several activities in statistical way. Next, data are sent to the Fitbit cloud from where, through a API Service, we are able to extract raw data to Monwatch. So, Monwatch : i) processes data, ii) stores the records in an internal database, iii) generates the instances of protoforms and their truth degree, iv) classifies the protoform instances according with the resulting clusters and v) exports and displays the results by linguistic expressions.

4.3 Protoform definition and their linguistic representation

A supervisor has defined the five HKIs that accurately describes the source streams of Fitbit using the protoform $V_r\ T_j\ Q_k$ described in Section 3.2 :

1. HR is high around 15 and 30 minutes with normal intensity
2. HR is low around 2 and 4 hours with normal intensity
3. AS is deep around 2 and 4 hours with moderate intensity
4. AS is (rem AND light) around 2 and 4 hours with moderate intensity
5. AS is (restless AND asleep) around 30 minutes and 2 hours with moderate intensity

These HKIs or protoform instances contains several terms, FTW and quantifiers which have been defined by fuzzy sets. The trapezoidal membership function for each fuzzy set defined in this proposal is described in Table 2 [1].

In addition to this, each protoform instance is represented by a set of linguistic expressions closer to natural language. These expressions that improve the semantic expressiveness of the protoform instances and shorten its length in most of the cases,

[1] TS, TR and TL are described in the appendix Abbreviations A

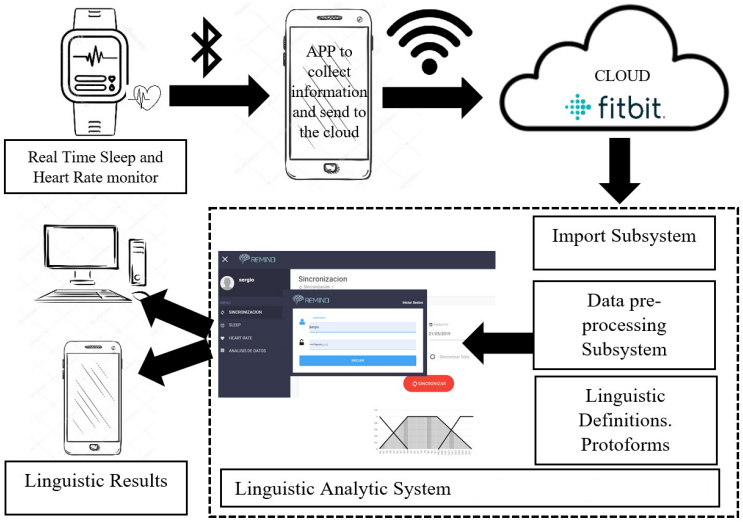


Figure 3: System architecture.

Table 2: Trapezoidal membership functions for terms, FTWs and quantifiers defined for the protoform $V_r\ T_j\ Q_k$

.

Textual description in natural language	**Type**	μ_T
hr is low	V_r	$TL(s_i^l)[60bpm, 70bpm]$
hr is high	V_r	$TR(s_i^l)[80bpm, 90bpm]$
around 15 and 30 minutes	T_j	$TL(\Delta t_i)[15m, 30m]$
around 2 and 4 hours	T_j	$TL(\Delta t_i)[240m, 480m]$
around 30 minutes and 2 hours	T_j	$TL(\Delta t_i)[30m, 240m]$
normal intensity	Q_k	$TR(x)[0.25, 0.75]$
moderate intensity	Q_k	$TR(x)[0, 0.5]$

are defined in Table 3. According to this, our system stores the data knowledge using a set of protoform instances in a first place and, in the second one, a linguistic summary, more suitable to the final user, is provided by the system (Marín and Sánchez, 2016).

Table 3: Short linguistic description of each HKIs.

Linguistic description	**Protoform instance** ($V_r\ T_j\ Q_k$)
HR of high-intensity for a short time	*HR is high around 15 and 30 minutes with normal intensity*
HR of low-intensity for a long time	*HR is low around 2 and 4 hours with normal intensity*
deep sleep for long time	*AS is deep around 2 and 4 hours with moderate intensity*
light sleep	*AS is (rem AND light) around 2 and 4 hours with moderate intensity*
rest nap	*AS is (restless AND asleep) around 30 minutes and 2 hours with moderate intensity*

4.4 Identification of patterns

The degree ofB truth of each protoform instance is aggregated for a day and results have been evaluated by the fuzzy C-means clustering algorithm to extract some pattern behaviours of HKIs. This algorithm has been executed with different number of clusters ($N = 4, 5, 6, 7$) to analyze these protoform instances behaviour. The clusters obtained are shown in Figure 4.

In Figure 4, we can observe the different patterns

of HKIs that can be found for an user in a day in each cluster. There are four scenarios according with the analyzed number of clusters. For example, in the first scenario of the figure A) (with $N = 4$) the first cluster represents the days where the user mainly developed a high-hr session and a long rest nap at once, this is characterized as a pattern and its corresponding linguistic description is: *Day with HR of high-intensity for a short time and a rest nap*. Following the same example, there are also another pattern that shows those days where the user has not a deep sleep for long time but the remaining HKIs has been activated.

Regarding the analysis of the number of clusters in this real example, a solution with $N = 7$ disintegrates the values too much, giving us too specific patterns with isolated HKIs. Solutions with $N = 5$ or $N = 6$ are more inclusive with all the HKIs and both give us a good solution.

Finally, in addition to the extraction and visualization of daily pattern behaviours from clusters, the trend of the membership degree for each day and clusters of a given time period provides a very rich and representative information for the supervisor of the activities. For example, in Figure 5, we show the membership degree between clusters and 30 consecutive days which provide a visual relation of the trend of behaviour.

5 Conclusions and ongoing works

In this work a combination of semi supervised analysis of user behaviour is proposed. For that, a wearable tracker provides the source streams about sleep stages duration and heart rate from users in a non invasive way. The aim of the methodology has been focused on integrating expert criteria by means of protoforms, which evaluate the source streams computing the degree of the protoform instances in full time line. The second contributions lies in extract behaviours patterns for each day using fuzzy clustering.

The results from the case study, which was developed for more than 200 days, shows a promising capability to aggregate data, extract patterns and provide a linguistic and visual representation due to interpretability of the protoforms.

In on going works, we will focus on evaluating a wide range of users, which could provide several behaviours patterns. The analysis between the clusters from several users will provide a suitable comparative between user profiles.

Acknowledgments

Funding for this research is provided by EU Horizon 2020 Pharaon Project *Pilots for Healthy and Active Ageing*, Grant agreement no. 85718 and the Spanish Ministry of Science, Innovation and Universities and the European Regional Development Fund - ERDF (Fondo Europeo de Desarrollo Regional - FEDER) under project PGC2018-096156-B-I00 *Recuperación y Descripción de Imágenes mediante Lenguaje Natural usando Técnicas de Aprendizaje Profundo y Computación Flexible*.

Moreover, this contribution has been supported by the Andalusian Health Service by means of the research project PI-0387-2018 and the *Action 1 (2019-2020)* no. EI_TIC01 of the University of Jaén.

References

Yang Bai, Paul Hibbing, Constantine Mantis, and Gregory J. Welk. 2018. Comparative evaluation of heart rate-based monitors: Apple watch vs fitbit charge hr. *Journal of Sports Sciences*, 36(15):1734–1741. PMID: 29210326.

Z Beattie, Y Oyang, A Statan, A Ghoreyshi, A Pantelopoulos, A Russell, and C Heneghan. 2017. Estimation of sleep stages in a healthy adult population from optical plethysmography and accelerometer signals. *Physiological Measurement*, 38(11):1968–1979.

James C Bezdek, Robert Ehrlich, and William Full. 1984. Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203.

Liming Chen, Chris D Nugent, and Hui Wang. 2011. A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering*, 24(6):961–974.

R. Choi, W. Kang, and C. Son. 2017. Explainable sleep quality evaluation model using machine learning approach. In *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 542–546.

Ed. D. Hardt. 2019. The oauth 2.0 authorization framework. RFC 6749.

Emiro De-La-Hoz-Franco, Paola Ariza-Colpas, Javier Medina Quero, and Macarena Espinilla. 2018. Sensor-based datasets for human activity recognition–a systematic review of literature. *IEEE Access*, 6:59192–59210.

Keith M Diaz, David J Krupka, Melinda J Chang, James Peacock, Yao Ma, Jeff Goldsmith, Joseph E Schwartz, and Karina W Davidson. 2015. Fitbit®:

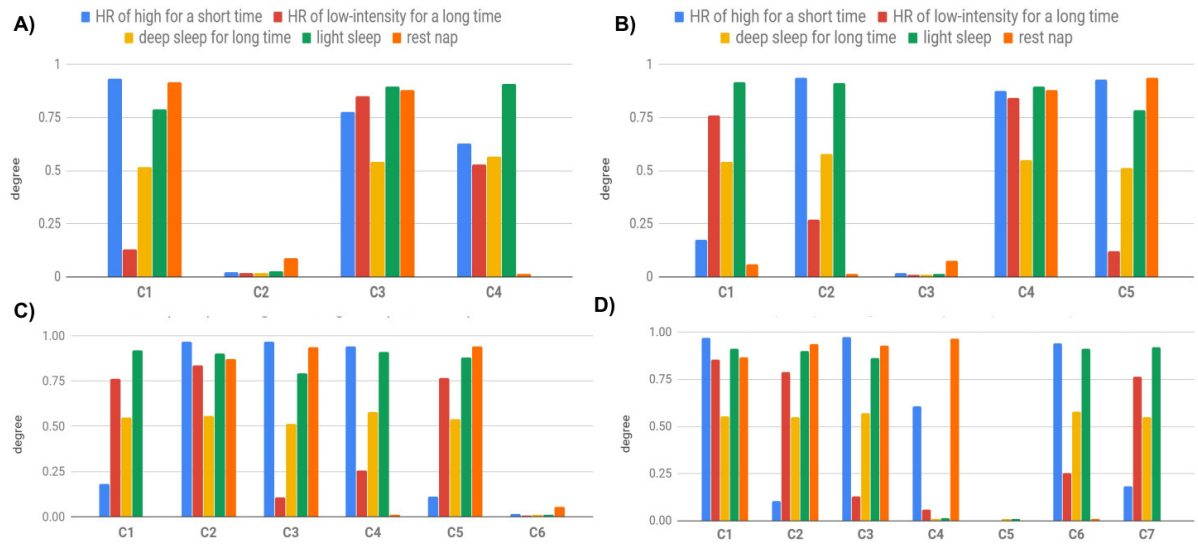


Figure 4: Clusters obtained by fuzzy C-means algorithm of the main HKIs for different values of N: A) $N = 4$, B) $N = 5$, C) $N = 6$ and D) $N = 7$

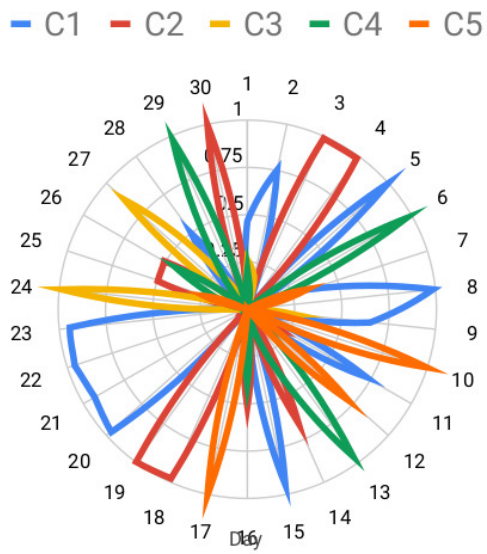


Figure 5: Evolution of the membership degree between clusters and 30 consecutive days for fuzzy C-means with $N = 5$.

An accurate and reliable device for wireless physical activity tracking. *International journal of cardiology*, 185:138–140.

David Dickinson, Joseph Cazier, and Thomas Cech. 2016. A practical validation study of a commercial accelerometer using good and poor sleepers. *Health Psychology Open*, 3(2):2055102916679012.

WM Dong and FS Wong. 1987. Fuzzy weighted averages and implementation of the extension principle. *Fuzzy sets and systems*, 21(2):183–199.

M. J. Duncan, C. Vandelanotte, S. G. Trost, A. L. Rebar, N. Rogers, N. W. Burton, and W. J. Brown. 2016. Balanced: a randomised trial examining the efficacy of two self-monitoring methods for an app-based multi-behaviour intervention to improve physical activity, sitting and sleep in adults. *BMC public health*, 16(670).

Macarena Espinilla, Javier Medina, Ángel-Luis García-Fernández, Sixto Campaña, and Jorge Londoño. 2017. Fuzzy intelligent system for patients with preeclampsia in wearable devices. *Mobile Information Systems*, 2017.

National Sleep Foundation. 2019. Which sleep tracker is best for you?

Liane D Heale, Saunya Dover, Y Ingrid Goh, Victoria A Maksymiuk, Greg D Wells, and Brian M Feldman. 2018. A wearable activity tracker intervention for promoting physical activity in adolescents with juvenile idiopathic arthritis: a pilot study. *Pediatric Rheumatology*, 16(1):66.

Janusz Kacprzyk and Ronald R Yager. 2001. Linguistic summaries of data using fuzzy logic. *International Journal of General System*, 30(2):133–154.

Seung-Gul Kang, Jae Myeong Kang, Kwang-Pil Ko, Seon-Cheol Park, Sara Mariani, and Jia Weng. 2017. Validity of a commercial wearable sleep tracker in adult insomnia disorder patients and good sleepers. *Journal of Psychosomatic Research*, 97:38 – 44.

Bhanu Prakash Kolla, Subir Mansukhani, and Meghna P. Mansukhani. 2016. Consumer sleep tracking devices: a review of mechanisms, validity and utility. *Expert Review of Medical Devices*, 13(5):497–506. PMID: 27043070.

Zilu Liang, Bernd Ploderer, Wanyu Liu, Yukiko Nagata, James Bailey, Lars Kulik, and Yuxuan Li. 2016.

Sleepexplorer: a visualization tool to make sense of correlations between personal sleep data and contextual factors. *Personal and Ubiquitous Computing*, 20(6):985–1000.

Stephen S Lim, Theo Vos, Abraham D Flaxman, Goodarz Danaei, Kenji Shibuya, Heather Adair-Rohani, Mohammad A AlMazroa, Markus Amann, H Ross Anderson, Kathryn G Andrews, et al. 2012. A comparative risk assessment of burden of disease and injury attributable to 67 risk factors and risk factor clusters in 21 regions, 1990–2010: a systematic analysis for the global burden of disease study 2010. *The lancet*, 380(9859):2224–2260.

Nicolás Marín and Daniel Sánchez. 2016. On generating linguistic descriptions of time series. *Fuzzy Sets Syst.*, 285(C):6–30.

Javier Medina-Quero, Macarena Espinilla, and Christopher Nugent. 2016. Real-time fuzzy linguistic analysis of anomalies from medical monitoring devices on data streams. In *Proceedings of the 10th EAI international conference on pervasive computing technologies for healthcare*, pages 300–303. ICST (Institute for Computer Sciences, Social-Informatics and .

Javier Medina Quero, Miguel Ángel López Medina, Alberto Salguero Hidalgo, and Macarena Espinilla. 2018. Predicting the urgency demand of copd patients from environmental sensors within smart cities with high-environmental sensitivity. *IEEE Access*, 6:25081–25089.

Kathryn Mercer, Lora Giangregorio, Eric Schneider, Parmit Chilana, Melissa Li, and Kelly Grindrod. 2016a. Acceptance of commercially available wearable activity trackers among adults aged over 50 and with chronic illness: a mixed-methods evaluation. *JMIR mHealth and uHealth*, 4(1):e7.

Kathryn Mercer, Melissa Li, Lora Giangregorio, Catherine Burns, and Kelly Grindrod. 2016b. Behavior change techniques present in wearable activity trackers: a critical analysis. *JMIR mHealth and uHealth*, 4(2):e40.

Bryan Minor, Janardhan Rao Doppa, and Diane J Cook. 2015. Data-driven activity prediction: Algorithms, evaluation methodology, and applications. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 805–814. ACM.

Victoria Moreno-Cano, Fernando Terroso-Saenz, and Antonio F Skarmeta-Gomez. 2015. Big data for iot services in smart cities. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 418–423. IEEE.

George Okeyo, Liming Chen, Hui Wang, and Roy Sterritt. 2014. Dynamic sensor data segmentation for real-time knowledge-driven activity recognition. *Pervasive and Mobile Computing*, 10:155–172.

María Dolores Peláez-Aguilera, Macarena Espinilla, María Rosa Fernández Olmo, and Javier Medina. 2019. Fuzzy linguistic protoforms to summarize heart rate streams of patients with ischemic heart disease. *Complexity*, 2019.

Alejandro Ramos-Soto, Alberto Jose Bugarin, Senén Barro, and Juan Taboada. 2014. Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data. *IEEE Transactions on Fuzzy Systems*, 23(1):44–57.

Patrick C Shih, Kyungsik Han, Erika Shehan Poole, Mary Beth Rosson, and John M Carroll. 2015. Use and adoption challenges of wearable activity trackers. *IConference 2015 Proceedings*.

Dongkuan Xu and Yingjie Tian. 2015. A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2):165–193.

Yan Yan, Elisa Ricci, Gaowen Liu, and Nicu Sebe. 2015. Egocentric daily activity recognition via multitask clustering. *IEEE Transactions on Image Processing*, 24(10):2984–2995.

Lotfi A Zadeh. 2002. A prototype-centered approach to adding deduction capability to search engines-the concept of protoform. In *2002 Annual Meeting of the North American Fuzzy Information Processing Society Proceedings. NAFIPS-FLINT 2002 (Cat. No. 02TH8622)*, pages 523–525. IEEE.

Lotfi A. Zadeh. 2006. *Generalized Theory of Uncertainty (GTU) – Principal Concepts and Ideas*, pages 3–4. Springer Berlin Heidelberg, Berlin, Heidelberg.

A Appendices

HR Heart Rate

HKI Health Key Indicator

FTW Fuzzy Temporal Window

AS Activity Status

FWA $$V_{r\cup T_k}(s^j)=\frac{1}{\sum T_k(\Delta t_i^j)}\sum_{m_i^j\in s^j} V_r(v_i^j)\times T_k(\Delta t_i^j)$$

TS $$TS(x)[l_1,l_2,l_3,l_4]=\begin{cases}0 & x\leq 0\\ (x-l_1)/(l_2-l_1) & l_1\leq x\leq l_2\\ 1 & l_2\leq x\leq l_3\\ (l_4-x)/(l_4-l_3) & l_3\leq x\leq l_4\\ 0 & l_4\leq x\end{cases}$$

TR $$TR(x)[l_1,l_2]=\begin{cases}1 & x\leq l_1\\ (l_2-x)/(l_2-l_1) & l_1\leq x\leq l_2\\ 0 & l_2\leq x\end{cases}$$

TL $$TL(x)[l_1,l_2]=\begin{cases}0 & x\leq l_1\\ (x-l_1)/(l_2-l_1) & l_1\leq x\leq l_2\\ 1 & l_2\leq x\end{cases}$$

Automatic Follow-up Question Generation for Asynchronous Interviews

Pooja Rao S B and **Manish Agnihotri** and **Dinesh Babu Jayagopi**
International Institute of Information Technology Bangalore
Karnataka, India

Abstract

The user experience of an asynchronous video interview system is often deemed non-interactive and one-sided. Interview candidates anticipate them to be natural and coherent like a traditional face-to-face interview. One aspect of improving the interaction is by asking relevant follow-up questions based on the previously asked questions, and its answers. We propose a follow-up question generation model capable of generating relevant and diverse follow-up questions. We develop a 3D virtual interviewing system, *Maya*, equipped with follow-up question generator. Many existing asynchronous interviewing systems pose questions that are fixed and scripted. *Maya*, on the contrary, reacts with relevant follow-up questions, a relatively unexplored dimension in virtual interviewing systems. We leverage the implicit knowledge from deep pre-trained language models along with a small corpus of interview questions to generate rich and diverse follow-up questions in natural language. The generated questions achieve 77% relevance with human evaluation. We compare our follow-up question generation model with strong baselines of neural network and rule-based systems and show that it produces better quality questions.

1 Introduction

The conventional hiring process is laden with challenges like prolonged hiring, lack of interviewers, expensive labour, scheduling conflicts etc. Traditional face-to-face interviews lack the ability to scale. Recent advances in machine learning has enabled automation in the field of recruitment. Recruiters are heeding to innovative choices like Asynchronous Video Interviews (AVI). Asynchronous interviews have a time-lapse between the communicating parties. These are usually conducted via online video interviews using internet-enabled digital devices. The feasibility and ease of

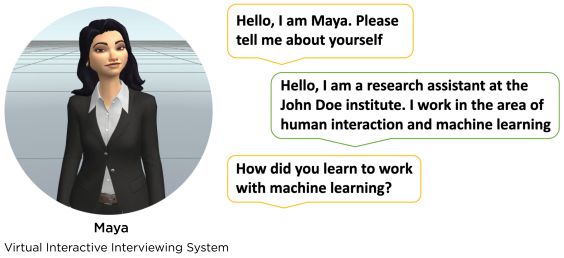


Figure 1: *Maya* - Interactive Interviewing System

automatic assessment of the AVIs when compared to in-person interviews (Rasipuram et al., 2016) is persuading the wide spread use of the system.

Limited prompting and follow-up, and no elaboration on questions is one of the components of structured interviews (Levashina et al., 2014). The current generation of asynchronous interview systems adopt structure and pose predefined questions selected from a relatively large set. However, with large scale adoption of these systems, it may eventually become repetitive and uninteresting for recruiters and candidates alike. The highly structured attribute of AVIs increases predictability, reduces variability, and makes them monotonous (Schmidt et al., 2016). Hence, it might be crucial to find the right balance between structure and probing. The adoption of planned or limited probing might help interviewers collect additional information related to the job, which may lead to increased interview validity (Levashina et al., 2014).

Levashina et al. (Levashina et al., 2014) define follow-up question as the one that is intended to augment an inadequate or incomplete response provided by the applicant, or to seek additional or clarifying information. Asynchronous communication does not enable coordinated turn-taking by interactants (Potosky, 2008). Integrating limited number of follow-up questions during the asynchronous interviews promises to solve the problem. A relevant follow-up question not only improves the interac-

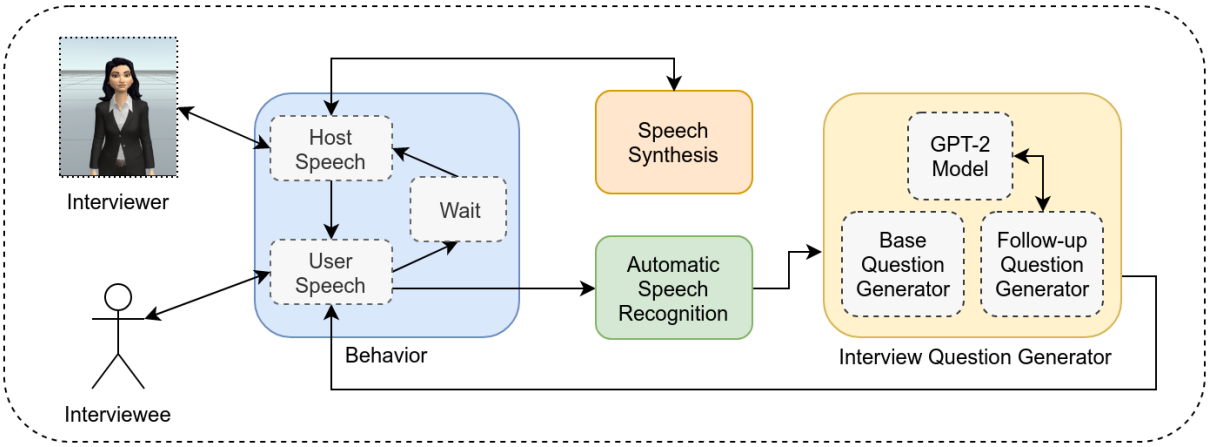


Figure 2: Framework of Interviewing System

tion between the interviewer and the interviewee but also makes it less predictable as the follow-up question is dynamic based on the interviewee's answer.

Based on these factors, we propose *Maya*, a 3D virtual interviewing system for behavioural domain. Specifically, the main contributions of this work are as follows. First, we present *Maya*, an interactive interviewing system equipped with a Follow-up Question Generation. We develop a framework for using large-scale transformer language model to generate relevant and diverse follow-up questions. Second, we perform experiments comparing Follow-up Question Generation (FQG) model with other strong question generation/selection models and show that the proposed model outperforms them by large margins with human evaluation. Finally, we perform experiments to study the robustness of the proposed model to errors in automatic speech recognition (ASR). The results indicate that *Maya* is able to produce high quality follow-up questions and hold an interactive interview with the candidate. We deploy a web-based minimalist virtual interview interface.[1]

2 Related Work

2.1 Natural Language Question Generation

Question Generation (QG), defined as the task to automatically generate questions from some form of text input (Rus and Graesser, 2009), has attracted attention since the First Question Generation Shared Task Evaluation Challenge (Rus et al., 2009). Recently, neural networks have enabled end-to-end training of question generation models influenced by the sequence-to-sequence (Seq2Seq) data-driven learning methods (Sutskever et al., 2014). Serban et al., (Serban et al., 2016) train a neural system to generate simple natural questions from structured triples - subject, relation, object. Du et al., (Du et al., 2017) use encoder-decoder model with attention to generate questions on the machine comprehension dataset SQuAD (Rajpurkar et al., 2016). QG-net (Wang et al., 2018b) is an RNN-based encoder-decoder model, trained on SQuAD, designed to generate questions from educational content.

Follow-up question generation in interviews is a new task and one study explores this (Su et al., 2018). Su et al., adopt a pattern-based Seq2Seq model on a small interview corpus in Chinese. They use a word clustering based method to build a word class table and transform all sentences in the corpus to patterns. Convolutional neural tensor network based (Qiu and Huang, 2015) sentence selection model is used on the answers to select a sentence to generate follow-up question patterns. These patterns are filled with words from the word class table to obtain potential follow-up questions. A statistical language model is used to choose a question by ranking. In contrast, we develop a follow-up question generation model utilizing knowledge from large-scale language model and a small corpus which does not involve pattern matching and template filling.

2.2 Language Model Pretraining

Pre-training on massive amounts of text in an unsupervised form has led to state-of-the-art advancements on diverse natural language processing tasks

[1]The demo of the system can be found at – https://www.youtube.com/watch?v=gdPxdi82nV0

System	Agent	Nonverbal Interaction	Verbal Interaction	Follow-up Q
Rao S B et al., 2017	Text Medium	No interaction	Fixed Script of Questions	No
SPECIES (Nunamaker et al., 2011)	Embodied Agent	Head Movement and Facial Expressions	Template based	Yes
MACH (Hoque et al., 2013)	Embodied Agent	Head Nodding and Smile Sharing	Fixed Script of Questions	No
TARDIS (Anderson et al., 2013)	Embodied Agent	Body Motions, Gestures and Facial Expressions	Fixed Script of Questions	No
ERICA (Kawahara, 2018)	Robotic Agent	Head Movement, Gestures and Eye Gaze	Template based	Yes
Maya (Ours)	Embodied Agent	Gestures, Facial Expressions and Follow-up Question	Dynamic Question Generation	Yes

Table 1: A comparison of asynchronous interview systems. The verbal interaction in *Maya* differs from other works with a follow-up question mechanism as it uses a question generation model rather than using template-based question selection method

(Devlin et al., 2018) (Radford et al., 2018). Currently, these pre-training steps are all variants of language modelling objectives. Howard and Ruder (Howard and Ruder, 2018) train a language model on huge amounts of Wikipedia data and fine-tune this on a target task with a smaller amount of labelled in-domain data. Several works follow this approach of fine-tuning and achieve impressive results. ELMo (Peters et al., 2018) is a bidirectional language model predicting the next and the previous tokens using bi-LSTM networks (Huang et al., 2015). OpenAI's GPT (Radford et al., 2018) train a unidirectional language model on massive text data. BERT (Devlin et al., 2018) is a masked language model trained with an additional objective of next sentence prediction. These models have attained state-of-the-art results on many downstream NLP tasks including the GLUE benchmark (Wang et al., 2018a). Pre-training with GPT model has also been used in generative tasks such as end-to-end dialog systems (Wolf et al., 2019) and automatic knowledge base completion (Bosselut et al., 2019) obtaining remarkable improvements over the models trained only with the in-domain data. Both the works use the transformer language model GPT for initialization. Our work builds on this to develop a Follow-up Question Generation model.

2.3 Agent-based Interviewing Systems

The use of intelligent virtual agents in dialogue systems has notably increased (Swartout et al., 2013) as it allows for a more interactive and immersive experience than traditional voice and text-based systems (López-Cózar et al., 2014). One primary application of virtual agents are in the Asynchronous Video Interviews (AVIs). A job interview is aimed to analyze the hiring feasibility of an interviewee, while a training interview gives accurate feedback about their performance.

While the initial works in AVIs were restricted to the skill assessment (Nguyen et al.), (Rao S B et al., 2017), improving the interview experience has gained momentum. One standard approach is the usage of virtual agents as interviewers instead of textual prompts to conduct interviews (Nunamaker et al., 2011). This approach makes the interview experience more interactive.

SPECIES (Nunamaker et al., 2011) introduced the usage of Embodied Conversational Agents in automated interviews. One of the goals was to study the difference in perceptions with varying attributes of agent. MACH (Hoque et al., 2013) and TARDIS (Anderson et al., 2013) are coaching-based conversational agents. Both of them focus on skill assessment and non-verbal behavior analysis to improve the feedback to interviewees significantly, but the questions are taken from a fixed pool of questions and do not take into account the interviewee's response. ERICA (Kawahara, 2018), consists of a robotic agent who has the capabilities of human-like eye gaze, head movement and gestures, and a statement-response system which is response retrieval method based on pattern and focus token matching. Although the behavior synthesis is a notable improvement, it still lacks robustness in dialogue generation.

While a lot has been done in automatic analysis of interviewee's response (Hemamou et al., 2019) to improve the quality of the interview, not much has been done to make the interview more verbally interactive. All the previous works have either used a fixed script of questions or used a pattern matching based question selection. We aim to improve the question generation system to make it more personal and response-based by generating relevant and grammatically correct follow-up questions.

3 Follow-up Question Generation - FQG

Follow-up Question Generation model is an adaptation framework for generating follow-up questions using language models by training it on an in-domain corpus of question, response and follow-up triplets. These data samples help FQG to learn the question structure and the relation between the triplets, and the knowledge from the language model pre-training produces novel questions.

3.1 Task

The training samples of $\{q, r, f\}$ in natural language, where q is the interviewer question, r is the candidate response and f is the follow-up question, are assumed to be given to the model. The task is to generate f given q and r as inputs.

3.2 Transformer Language Model

In this work, we use the transformer language model architecture, Generative Pre-trained Transformer (GPT-2) introduced in Radford et al. (Radford et al., 2019). This is very similar to the decoder part of the original transformer encoder-decoder model of Vaswani et al. (Vaswani et al., 2017). It uses multiple transformer layers each containing two sub-layers. First is the multi-headed self-attention mechanism over the input context tokens followed by position-wise feed-forward layers to produce an output distribution over target tokens. Our model is based on the recently published PyTorch adaptation of GPT-2.[2]

We initialize the Follow-up Question Generation model with 12-layer decoder-only transformer with 12 self-attention heads containing 768 dimensional states. The parameters are initialized to the smallest version of the GPT-2 model weights open-sourced by Radford et al. 2019 (Radford et al., 2019). The GPT-2 model is pre-trained on the WebText dataset which contains the text of 45 million links from internet (Radford et al., 2019).

3.3 Dataset

In order to train the FQG model, we need the training samples – $\{q, r, f\}$ triplets. We utilize the asynchronous interview dataset from Rao S. B et al. (Rao S B et al., 2017). This dataset consists of behavioural interviews of university students through asynchronous medium of video and written, referred to as the Asynchronous Video Interview dataset - AVI dataset and Asynchronous Written Interview dataset - AWI dataset respectively. We conduct a restricted crowd-sourcing to obtain follow-up questions using interview snippets from AWI dataset. We instruct the volunteers to write a follow-up question based on the presented snippet of interviewer question and the candidate response. Thus, we obtain a follow-up question dataset with more than 1000 samples, each sample containing the triplet of a question, response and a follow-up. The dataset can be found at https://ms-by-research-thesis.s3.amazonaws.com/followMLdata.xlsx

3.4 Fine-tuning

We fine-tune the GPT-2 language model using the dataset described above. 80% of the data is used for training and the rest is used for validation. The input to the model constitutes of tokens from each of the $\{q, r, f\}$ concatenated in a sequence. A set of input embeddings is constructed for this sequence. The word and position embeddings are learnt in the pre-training phase. We use an additional set of embeddings, speaker embeddings to indicate whether the token belongs to question, response or the follow-up. These embeddings are learnt during the fine-tuning phase. The input to the model is the sum of all three types – word, position and speaker embeddings for each token. Figure 3 illustrates how the tokens in $\{q, r, f\}$ are organised to form the speaker embeddings.

Following (Wolf et al., 2019), (Devlin et al., 2018), the fine-tuning is done by optimizing two loss functions – a language modelling loss, and a next-question classification loss. The language modelling loss is the commonly used cross-entropy loss. The last hidden state of the self-attention model is fed into a softmax layer over all the tokens in the vocabulary to obtain next token probabilities. These probabilities are then scored using the cross-entropy loss where the human written follow-up question tokens are used as labels.

A next-question classifier is trained to recognize the correct next question among the distractors of random questions. We append the dataset consisting of correct follow-up questions with randomly sampled questions from a pool of 200 (same as the ones used in Section 5), acting as distractors. This trains the model to learn a sense of sentence ordering. The classifier is a linear layer apply-

[2] https://github.com/huggingface/transformers

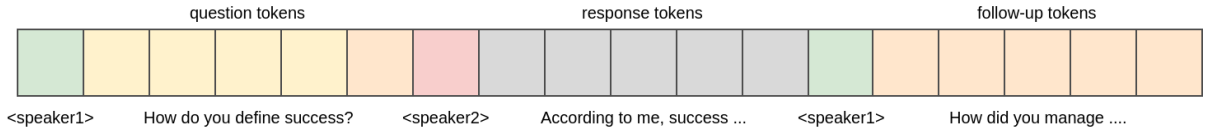


Figure 3: Input representation for training Follow-up Question Generation model

ing a linear transformation to the last hidden state of self-attention model to compute a value. Using the computed values, a softmax layer obtains the classification probabilities. Then we apply a cross-entropy loss to correctly classify the correct follow-up question. We use $n = 2$ as the number of choices for classification making it a binary classification task. The parameters of the transformer language model and the next-question classifier layer are fine-tuned jointly to maximize the log-probability of the correct label.

3.4.1 Decoding details

We use the top-k random sampling strategy for decoding (Fan et al., 2018). At each timestep, the probability of each word in the vocabulary being the next likely word is given. The decoder randomly samples a word from the *k* most likely candidates. Here *k* is a hyperparameter determined to be *k*=10 experimentally.

3.5 Results

We report the results of the follow-up question generation model in terms of perplexity (Bengio et al., 2003). We also report the classification accuracy of next-question classification task. Perplexity is usually used to measure the quality of language models. It indicates how well the model predicts the next word correctly. Our model obtains an average validation perplexity of 20.6 and average validation accuracy of 63.1%. These values can be deemed reasonable considering the small size of the in-domain dataset used for fine-tuning. It may also be due to the fact that the questions generated are novel and relevant leveraging the knowledge from the pre-training step which may not be present in the human written follow-up questions.

4 Experiments

In this section we showcase the efficiency of the FQG model through quantitative and qualitative analysis. First, we compare FQG with strong baselines. Second, we quantitatively confirm the relevance of the follow-up questions through human evaluation. Next, we investigate the robustness of the FQG model to errors in speech. Finally, we qualitatively examine the results of the FQG model.

4.1 Experimental Setup

We compare the FQG with two strong baselines. One is a rule-based system based on similarity measure and other is the reader-generator based QG-Net model (Wang et al., 2018b).

4.1.1 Similarity-based Question Selector

This model is a rule-based pre-defined question selector which selects questions from a pool of 200 behavioural questions (same as the ones used in Section 5) based on cosine similarity measure. We calculate the cosine similarity metric between the original interview question and each of the questions from the pool. We consider the top-10 most similar questions and randomly select one to be the follow-up question. This question selector loosely mimics the different rule-based question selectors in the existing systems.

4.1.2 QG-Net

QG-net is a Seq2Seq model with a context reader and question generator. The context reader is a bi-LSTM network which processes each word in the input context and turns it into a fix-sized representation. The question generator is a uni-directional LSTM which generates the question word-by-word incorporating pointer network (See et al., 2017) on the generator vocabulary. This model design enables the generator to output questions that focus on specific parts of input text. The *focus tokens* are encoded with each input word as an additional feature using one-hot encoding indicating whether the word is a focus token. QG-Net is trained on SQuAD dataset consisting of context, question and span of answer tokens within the context. QG-Net uses these answer tokens as focus tokens. Linguistic features like the POS tags, named entity and word case are also used as additional features. We refer the readers to the original paper for a detailed overview (Wang et al., 2018b). QG-Net effectively adapts a general purpose question generation model trained on SQuAD to generate questions from educational content, addressing the problem of insuf-

ficient training data. Hence we choose this as our neural network baseline model. In our case the candidate response is the context and the follow-up question is the question to be generated.

Since QG-Net model expects a sentence with its focus tokens as input, the interview question-answer pairs have to undergo preparatory techniques like finding focus of the answer and extractive summarization before feeding into the QG-Net model. We use the QG-Net model trained on SQuAD dataset released by (Wang et al., 2018b).

Finding Focus of the Answer QG-net uses a binary valued indicator as an added feature to indicate whether a word in context is important to generate a question, regarded as *focus tokens*. We employ a simple technique similar to Hu et al., (Hu et al., 2018) to automatically find these tokens. There exist overlapping tokens in the question (Q) and answer (A) pairs, seen as topics shared between them, that can be considered as focus tokens.

After removal of the stop words, A and Q are represented as a sequence of tokens $[a_1, .., a_n]$ and $[q_1, .., q_m]$ respectively. We consider all the tokens in A as candidates for focus tokens and all the tokens in Q as voters polling for the candidates. GloVe (Pennington et al., 2014) vectors are used to represent tokens from Q and A. The i^{th} answer token a_i gets a cumulative score S_i from all the tokens in the question calculated as

$$S_i = \sum_{j=1}^{m} p_{ij}.sim(a_i, q_j)$$

$$p_{ij} = \begin{cases} 1, & sim(a_i, q_j) > \lambda \\ 0, & \text{otherwise} \end{cases}$$

where $sim(a_i, q_j)$ is the cosine similarity between a_i and q_i. If the averaged S_i is above a certain threshold, a_i is included in the *focus*. This process is repeated for every answer token.

Extractive Summarisation The input to the QG model should be a representative of the response and give information for a potential follow-up. We employ a simple extractive summarization technique on the sentences of the answer. We use the method described above to find the focus of each sentence. We then compare the focus of each sentence with the focus of other sentences using the cosine similarity measure. R and S are two sentences from the candidate response with their focus tokens represented as $[fr_1, ..., fr_p]$ and $[fs_1, ..., fs_q]$ respectively. The cumulative score for each focus token of R is calculated as

$$W_i = \sum_{j=1}^{q} p_{ij}.sim(fr_i, fs_j) \quad N = \sum_{i=1}^{p} W_i$$

where p_{ij} is the indicative variable same as described above. If N crosses a certain percentage of the mean length of two sentences R and S, they are considered to be similar.

Once we have the pair(s) of similar sentences, we choose the one with more information content (more number of focus tokens) as the summary sentence. If more than one pair of sentences are similar to each other, *S* (pre-determined) number of sentences with the highest frequency of similar sentences is considered. The summary sentence along with the focus words is fed to the trained QG-Net model to generate questions.

4.2 Human Evaluation

To evaluate the quality of the generated follow-up questions and compare it against the baselines, we get human annotations. Human annotators involved in this study are non-native English speakers and graduate students with a background in Computer Science and Digital Society. We randomly sample 100 unseen question-answer pairs from the AWI dataset and generate one follow-up question (FQ) per QA pair from all three models– Similarity-based Question Selector, QG-Net question generation and GPT-2 based Follow-up Question Generation. We present the QA pair along with the follow-up questions generated by each model to the human annotators. They are asked to rank the questions based on their preference in the order of two metrics– relevance of FQ to the given interview QA pair and their grammar.

We consider the statistical mode of the ranking from three annotators for each follow-up question. When the mode is not unique i.e, when all three annotators choose a different rank, we resolve the disagreement by getting an extra set of rankings from an experienced interviewer. This is the case for about 10% of the annotations.

The results are shown in Figure 4. The bar plot indicates the count of mode of the ranks from evaluators for each of the model. FQG model significantly outperforms (well beyond p=0.01 level) the other two models with 54% of questions securing Rank 1, followed by 34% from QG-Net. 50% of the questions from SQS secure Rank 2. It can be observed that grammatically correct selected questions from SQS are preferred second after FQG

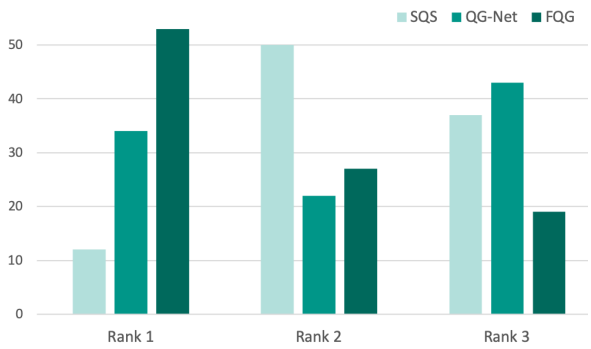


Figure 4: Human ranking of preferred follow-up questions from FQG comparing with two other baseline models based on relevance and grammar. The bar indicates the frequency of rankings, indicating that the FQG model is the most preferred for highest ranking.

model than the gramatically incorrect and somewhat relevant questions from QG-Net model. We conclude that FQG model generates relevant and grammatically correct follow-up questions more often than the existing baselines.

We further strengthen the evaluation of FQG model by obtaining individual human ratings for the follow-up questions. Three human annotators evaluate the quality of the questions on a scale of 1-3, 1 being the lowest. The annotators are instructed to rate the questions based on grammar and relevance of the question to the original interview question and answer. We consider the average ratings from three annotators for evaluation. Figure 5 gives the statistics of the average ratings for the follow-up questions generated. 77% of the questions are scored $\geq$ 2. And 27% are rated $\geq$ 2.5. This shows that the FQG model generates superior quality follow-up questions and are scored well by humans.

4.3 Robustness to Errors in Speech

Investigating the robustness of Follow-up Question Generator has an important motivation. The model is trained on human-written triplets of $\{q, r, f\}$ whereas it will be inferred on the candidates's response obtained from ASR transcript in the virtual interviewing system. Hence, analyzing how follow-up question generation varies for ASR transcripts when compared with human transcripts helps to investigate the robustness of FQG model.

We use the asynchronous interface-based video interview dataset from Rasipuram et al. (Rasipuram et al., 2016) for this purpose as they have manual transcriptions of the interviews. We randomly select 103 question answer pairs. We also

Average Ratings	Avg Rating on written QA pair	Avg Rating on manual transcripts	Avg Rating on automatic transcripts
1	2	0	4
1.3	9	11	15
1.67	12	21	18
2	23	22	22
2.3	27	21	21
2.6	20	17	20
3	7	11	3

Figure 5: Frequency distribution of average human ratings on the quality of generated follow-up questions from the FQG on a scale of 1-3 on the different types of question-answer pair inputs (hand-typed text, manually and automatically transcribed spoken text).

obtain automatic transcriptions for the same pairs of 103 question answers using the Web Speech API (Shires, 2019). We generate a follow-up question for each of this pair. This gives us 206 triplets of question, response, and follow-up questions, 103 each for manual and automatic transcripts.

Three human annotators evaluate the quality of the question on a scale of 1-3, 1 being the lowest. The annotators are displayed with the questions and answers from the manual transcripts and the follow-up questions generated on both manual and automatic transcripts to rate. We consider the average rating of the three annotators for evaluation.

Figure 5 shows count of the average ratings for follow-up questions on manual and automatic transcripts. 69% of the questions generated on manual transcripts and 64% of the questions generated on ASR transcripts get a score of $\geq$ 2. This implies that the FQG model generates relatively good quality follow-up questions on both manual and automatic transcripts.

4.4 Qualitative Analysis

Table 2 compares the follow-up questions generated by the follow-up question generation model with the two strong baselines, similarity-based question selector and QG-Net. It is seen that question generated by FQG is a grammatically correct and relevant follow-up in all cases. In regard with grammar, we can see that the QG-Net model struggles the most. FQG model, despite the grammatical errors in the answer, always adheres to grammar. Grammatical adherence is irrelevant with SQS model as it selects from a pool of grammatically correct questions.

In terms of relevance, we find that the FQG model always generates a follow-up relevant to the question or answer or both in all cases. SQS often produces follow-ups remotely related or unre-

Question: What is the biggest challenge you have faced so far? **Answer**: The biggest challenge that i have faced so far is IIT- JEE examination after completing my class 12. I prepared for it for two years. Unfortunately I didn't cracked it. **SQS**: Where do you see yourself 3 years from now? **QG-Net**: did you think was cracked ? **FQG**: What are the best lessons you have learned from this experience?
Question: Short self-presentation: Introduce yourself **Answer**: Hello, I am X, a passionate software engineer. Till my 12th standard I studied at my native place itself. My strongest plus point according to me is that I dont trust anyone easily and negative point is that I cant resist anything wrong happening to anyone around me.... **SQS**: What has been the greatest disappointment in your life? **QG-Net**: what is you that you can't resist happening? **FQG**: How do you manage your negative point?
Question: What has been the greatest disappointment in your life? **Answer**: I didn't get any high disappoint from my life. I was disappoint from myself when I didn't performed that well in GATE exam and not able to get admission in X, but I got admission in Y and now I feel that it was okay and not feel disappoint. **SQS**: What changes would you make if you came on board? **QG-Net**: what did you do in gate exam ? **FQG**: How would you explain your failure in GATE exam?
Question: How would you be an asset to this organisation? **Answer**: I think I have a hunger to learn and solve a problem. For me, each problem is a puzzle which I need to solve and I will try till I finish it. If I don't know something, I will learn it and get the job done. My hardworking and passionate qualities can be asset to your company. **SQS**: How quickly do you adapt to new technology? **QG-Net**: what do you have a hunger to do ? **FQG**: How do you think you can improve your skills?

Table 2: Comparison of examples generated from three different models for various question-answer pairs. SQS: Similarity-based Question Selector, QG-Net: QG-Net, FQG: Follow-up Question Generation

lated to the answers. Follow-ups by QG-Net model does not always comply to relevance. It generates unrelated or incomplete or grammatically incorrect relevant questions. We can also note that QG-Net model often generates questions which are already answered (eg. Questions 2, 4). We also present some examples generated only by the FQG model on randomly sampled QA pairs in Appendix A.

5 Maya - Interactive Interviewing System

Our interactive interviewing system, *Maya*, consists of two main components – 3D Virtual Interviewer and Interview Question Generator. The first is an Amazon Sumerian (Walker, 2017) based 3D virtual interviewing agent which asks questions and collects the interviewee's responses. We use ASR (Web Speech API (Shires, 2019)) to transcribe the user speech and this text data is fed to the second component, question generator, hosted on a server. Using Amazon Polly text-to-speech toolkit, the virtual agent communicates the generated question to the interviewee. The Interview Question Generator component contains two modules which communicates with the 3D virtual interviewer. *Base question selector* selects a question randomly from 200 questions commonly asked in an HR interview. Next question is a follow-up question generated by the *follow-up question generator*. In our experiments, we limit the number of follow-up question to one. The follow-up question is based on single previous response from the candidate and not the history. We consider one follow-up question as a proxy to planned or controlled probing.

6 Conclusion

We introduce *Maya*, a virtual agent-based interviewing system equipped with verbal interactivity from follow-up question generation. We leverage the implicit knowledge of a large scale transformer language model fine-tuned on follow-up questions dataset to generate relevant, novel and diverse questions based on the candidates' response in an interview. With availability of limited data, this approach scales as it uses external knowledge from a language model trained on a huge corpus. With human evaluation, we show that the questions generated are of good quality. We can also see that the FQG model is often robust to the errors of speech recognition. We restrict the generation of follow-up questions to one as existing research suggests the advantage of limited probing and follow-up. But the model is capable of generating multiple follow-up questions based on the previous response. The FQG model is not limited to behavioural domain but can also be trained on any other domain descriptive questions to generate follow-up questions.

One important future direction of this work can be modelling the problem as generation of follow-up question considering the complete history of the conversation and not just the previous question and response. A user study could be organised to validate the advantages of including the follow-up questions to boost the interaction.

Acknowledgments

This work was partially funded by SERB Young Scientist grant (Grant no: YSS2015001074) of Dr. Jayagopi, Karnataka government's MINRO grant and a grant from Accenture Technology Labs.

We thank all the participants who contributed for data collection. We would also like to thank all the reviewers for their insightful comments and suggestions.

References

Keith Anderson, Elisabeth André, Tobias Baur, Sara Bernardini, Mathieu Chollet, Evi Chryssafidou, Ionut Damian, Cathy Ennis, Arjan Egges, Patrick Gebhard, et al. 2013. The tardis framework: intelligent virtual agents for social coaching in job interviews. In *International Conference on Advances in Computer Entertainment Technology*, pages 476–491. Springer.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Çelikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *ArXiv*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR*.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *ACL*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *ACL*.

Léo Hemamou, Ghazi Felhi, Vincent Vandenbussche, Jean-Claude Martin, and Chloé Clavel. 2019. Hirenet: a hierarchical attention model for the automatic analysis of asynchronous video job interviews.

Mohammed Ehsan Hoque, Matthieu Courgeon, Jean-Claude Martin, Bilge Mutlu, and Rosalind W Picard. 2013. Mach: My automated conversation coach. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 697–706. ACM.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *ACL*.

Wenpeng Hu, Bing Liu, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2018. Aspect-based question generation. In *ICLR Workshop*.

Zhiheng Huang, Wei Liang Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *ArXiv*.

Tatsuya Kawahara. 2018. Spoken dialogue system for a human-like conversational robot erica. In *International Workshop Spoken Dialogue Systems*.

Julia Levashina, Christopher J Hartwell, Frederick P Morgeson, and Michael A Campion. 2014. The structured employment interview: Narrative and quantitative review of the research literature. *Personnel Psychology*, 67(1):241–293.

Ramón López-Cózar, Zoraida Callejas, David Griol, and José F Quesada. 2014. Review of spoken dialogue systems. *Loquens*, 1(2):012.

Laurent Son Nguyen, Denise Frauendorfer, Marianne Schmid Mast, and Daniel Gatica-Perez. Hire me: Computational inference of hirability in employment interviews based on nonverbal behavior. *IEEE transactions on multimedia*.

Jay F Nunamaker, Douglas C Derrick, Aaron C Elkins, Judee K Burgoon, and Mark W Patton. 2011. Embodied conversational agent-based kiosk for automated interviewing. *Journal of Management Information Systems*, 28(1):17–48.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke S. Zettlemoyer. 2018. Deep contextualized word representations. *ArXiv*.

Denise Potosky. 2008. A conceptual framework for the role of the administration medium in the personnel assessment process. *Academy of Management Review*, 33(3):629–648.

Xipeng Qiu and Xuanjing Huang. 2015. Convolutional neural tensor network architecture for community-based question answering. In *IJCAI*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy S. Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*.

Pooja Rao S B, Sowmya Rasipuram, Rahul Das, and Dinesh Babu Jayagopi. 2017. Automatic assessment of communication skill in non-conventional interview settings: a comparative study. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 221–229. ACM.

Sowmya Rasipuram, Pooja Rao S. B., and Dinesh Babu Jayagopi. 2016. Asynchronous video interviews vs. face-to-face interviews for communication skill measurement: A systematic study. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI '16, pages 370–377, New York, NY, USA. ACM.

Vasile Rus, Wyse Brendan, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Cristian Moldovan. 2009. The question generation shared task and evaluation challenge. In *The University of Memphis. National Science Foundation*.

Vasile Rus and Arthur C. Graesser. 2009. The question generation shared task and evaluation challenge. In *The University of Memphis. National Science Foundation*.

Frank L Schmidt, IS Oh, and JA Shaffer. 2016. The validity and utility of selection methods in personnel psychology: Practical and theoretical implications of 100 years... *Fox School of Business Research Paper*.

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

Iulian Serban, Alberto García-Durán, Çaglar Gülçehre, Sungjin Ahn, A. P. Sarath Chandar, Aaron C. Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. *CoRR*.

Glen Shires. 2019. Web speech api: Draft community group report. [Online; posted 17-July-2019].

Ming-Hsiang Su, Chung-Hsien Wu, Kun-Yi Huang, Qian-Bei Hong, and Huai-Hung Huang. 2018. Follow-up question generation using pattern-based seq2seq with a small corpus for interview coaching. In *Proc. Interspeech 2018*, pages 1006–1010.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

William Swartout, Ron Artstein, Eric Forbell, Susan Foutz, H Chad Lane, Belinda Lange, Jacquelyn Morie, Dan Noren, Skip Rizzo, and David Traum. 2013. Virtual humans for learning. *AI Magazine*, 34(4):13–30.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Tara Walker. 2017. Presenting amazon sumerian: An easy way to create vr, ar, and 3d experiences. [Online; posted 27-November-2017].

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018a. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Zichao Wang, Andrew S. Lan, Weili Nie, Andrew E. Waters, Phillip J. Grimaldi, and Richard G. Baraniuk. 2018b. Qg-net: a data-driven question generation model for educational content. In *L@S*.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *ArXiv*.

A Qualitative Examples

We also present some examples generated only by the FQG model on randomly sampled QA pairs in Table 3. As seen in the first part of the table, the questions generated on the validation samples are relevant and grammatically correct but differ from the human written follow-up questions. This shows the richness, diversity and variety in the questions apart from the training data. The second part of the table shows the follow-up questions generated on completely unseen data. The model generalizes well on the unseen data and generates grammatically correct and suitable questions. The third part of the table shows the questions generated on the text from the ASR. As it can be seen, the questions are relevant and appropriate, inspite of the errors in the transcriptions. It is to be noted that the original base question is not exposed to the errors of the ASR and can be a reliable input to the FQG model.

Validation Examples			
Question	Candidate Response	Generated Follow-up	Human Generated Follow-up
How would you be an asset to this organisation?	I can be an asset as I am patient while handling people and situations. I can participate and coordinate well with teams I get to work with. I can take in criticisms and work on to improve myself. I like to learn new things from anybody and will take necessary help whenever needed. I quite friendly with everyone.	How would you handle criticism?	Can criticism always be seen as a means for self-improvement?
Have you ever had to sell an idea to your co-workers? How did you do it?	I had to, when we were brainstorming ideas for a hacking session. I first tried to understand their concerns and requirements. Later I pitched my idea in accordance to that. I ensured that I understood and answered all their queries.	How did you handle the stress?	What were their first reactions?
What obstacles or difficulties have you ever faced in communicating your ideas?	I am not very expressive and can't put thoughts into words, so i have failed in communicating my ideas and explaining it or even getting work done.	How do you cope with a difficult situation?	And what steps have you taken to overcome this hardship?
What do people most often criticize about you?	I am not very social, an introvert and can't get along with people easily. I hvae been criticized about this.	What do you think about people who criticize you?	Have you ever tried to work on these drawbacks?
Unseen Examples			
Question	Candidate Response	Generated Follow-up	
What are your salary expectations?	As of now, I haven't thought much about it. I'm more focused on learning the requirements for this position that I am applying for.	How much do you think you will be able to meet these requirements?	
Are you a team player?	Definitely. While I will deny the fact that I can work independently with minimal supervision, I'm also one companion every leader would ever want to be in his team. Whatever the task is, I make sure it meets and exceeds what is expected of me. I also make it a point to reach out to teammates whenever needed.	Do you have any tips for team members?	
Why do you want to work for this company?	I would be proud to work for a company like yours with such a long history of leadership in the industry. I have carried out web research and believe that the company's products and its future projections are very impressive.	Why do you think this company is successful?	
Unseen Examples with ASR Transcriptions			
Question	Candidate Response from ASR	Generated Follow-up	
Do you consider yourself successful?	I do consider myself successful because I have been able to achieve the things that I wanted for myself so ja in the respect yes I too feel them successful	Do you think you can achieve the things you wanted for yourself?	
Tell me something about yourself.	I am a research assistant at the <name> institute I work in the area of human interaction and machine learning	How did you learn to work with machine learning?	
How do you think this job will help you fulfill your career goals?	when I apply for this job applied for this job and applied for this job many things which I like to do and I will do in the stop my career goals are just to have some work satisfaction and study whatever I want to so I think this job will help me do that and	How would you handle a situation where you have no work satisfaction?	

Table 3: Examples of the follow-up questions generated on the interview snippets from the validation data, unseen data and unseen data from ASR

FitChat: Conversational AI for Active Ageing

Nirmalie Wiratunga, Anjana Wijekoon, Chamath Palihawadana, Kay Cooper, Vanessa Mendham
School of Computing, School of Health Sciences
Robert Gordon University
Aberdeen AB10 7GJ
Scotland, UK
a.wijekoon@rgu.ac.uk

Abstract

Advances in conversational AI are creating novel and engaging experiences for user interaction through AI Chatbots. In this work, we present the voice-based AI chatbot "FitChat" developed to deliver behaviour change interventions that encourage physical activities among older adults. We start by identifying conversation skills or topics necessary to promote physical well-being through co-creation activities with users. For each conversation skill, we further explore the use of Natural Language Understanding (NLU) and Natural Language Generation (NLG) techniques to improve the conversation. We generate personalised conversation, contextualised with the information extracted from user responses. Provisioning educational content from WHO guidelines on physical well-being provided a useful knowledge source for contextualising chatbot responses and a corpus-based approach helped to avoid non-repetitive chatbot responses. We evaluate the prototype using think-aloud sessions where thematic analysis emphasises that voice-based chatbots are a powerful mode of intervention delivery. Analysis of user responses shows the NLU techniques were instrumental in extracting information that is essential to create cohesive and personalised conversations using NLG techniques.

1 Introduction

Presently, the most common method of delivering digital behaviour change interventions for encouraging physical activities is via text-based notifications on mobile phones. Despite the popularity of this approach, there is little evidence to indicate that text notifications are effective at promoting positive behaviour change, particularly long-term impact. The main problem is that text notifications offer only one-way communication, from the device to the user, meaning explicit interaction is not required. Accordingly, text notifications are easily ignored; fewer than 30% of received notifications are typically viewed by users with an average delay close to 3 hours (Morrison et al., 2018). There is clearly a need for an alternative approach.

As a communication medium, conversation appeals to all age groups, but arguably more so towards older adults. This group can have difficulties with new technologies and may be more inclined to appreciate the natural interaction offered by conversational dialogue. It is also noteworthy that older adults in general are not accustomed to text entry with smart phones. With this in mind, we posit that conversation (more specifically, voice-based conversation) presents an opportunity to deliver behaviour change interventions to motivate higher levels of adoption and adherence in older adults when compared with traditional approaches. In addition, the recent popularity of home hubs is considered a positive indication of user-acceptance towards voice based conversational agents for both smartphone and home-hub platforms. This means that our work is well-placed to investigate conversation as an alternative to current text-based intervention methods.

Our aim is to develop an ubiquitous and proactive system that delivers behaviour change interventions in the form of conversation aimed at promoting physical activities in older adults. We start by bringing together end-users from the community through co-creation workshops to help understand what are meaningful conversational interventions and therein develop and design a prototype. In development, we explore the state-of-the-art methods for Natural Language Understanding (NLU) and Natural Language Processing (NLP) to extract information from user responses and integrate these to generate contextually relevant chatbot responses that are non-repetitive. Accordingly we make the following contributions:

- present a co-creation method to identify and design 5 conversational AI skills and associated educational content required to promote physical well-being among older adults;
- develop a personalised response generation strategy that combines information extraction from open ended user responses to contextualise a template-based NLG method;
- create a cloud-based architecture for secure storage and integration with a cross-platform chatbot app; and
- carry out a comprehensive evaluation of the 5 conversation AI skills using a thematic analysis of think aloud sessions.

Rest of the paper is organised as follows; related literature is explored in Section 2 and Methods for identifying conversational skills, extracting information and response generation are explored in Section 3. Next we present the conversation design for each skill in Section 4, followed by the prototype implementation details in Section 5. Qualitative evaluation is presented in Section 6 and concluding remarks are mentioned on Section 7

2 Related Work

Conversational agents have been used as intervention delivery methods in many healthcare application domains including mental health (Morris et al., 2018; Inkster et al., 2018; Suganuma et al., 2018), weight loss and obesity (Stein and Brooks, 2017; Addo et al., 2013), alcoholism treatment (Lisetti et al., 2011, 2013), physical activity and diet (Fadhil et al., 2019; Fadhil and Villafiorita, 2017) and medication adherence (Fadhil, 2018). But this is lacking in applications which target general fitness. Existing smart phone applications restrict user responses to a selection from a number of choices or through free text entry. Initial research has explored the use of web based avatars to integrate voice and emotions into intervention delivery (Lisetti et al., 2013). However voice based conversational interfaces in the form of chat-bots are more naturally intuitive, compared to these web based avatars. Here good conversational coverage is essential to ensure that the learning curve is manageable without requiring the user to memorise key phrases to carry on a dialogue with the tool.

A recent evaluation of Wysa (Inkster et al., 2018), a text/multiple-choice empathetic AI chat-bot for mental well-being, focused on analysing user acceptance of conversational agents. Their findings suggests that a majority of 67% found Wysa to be a "Favourable Experience" compared to 32% who found it to be a "Less Favourable Experience". Users preferred to respond by clicking on options given by the app when compared to entering free text. Lark [1] is another well-known text/choice based Conversational Agent specialised in diabetes management and Stein and Brooks (2017) evaluates Lark for user acceptability and satisfaction where users rated the app at 7.9 (average) on a 0-10 scale. These studies suggest that in general conversational agents are widely accepted by the users, but they are limited to text or choice based responses. This motivates us to exploit advances in conversational AI and explore conversation as a form of delivering interventions in general fitness applications, specifically for older adults.

Recent literature suggests a corpus-based approach for enforcing empathy into text/choice based conversational bots (Morris et al., 2018). A corpus is curated with empathetic responses that will be used by the conversational agent when responding to a user. They measure the acceptability of empathetic responses presented by the bot compared to responses presented by a peer and found that users accept bot responses 79% of the time. Our work is closely related to this approach, where we also create a response bank. We acknowledge that there is a significant burden on knowledge engineering however we overcome this by using several user co-creation activities with a view to creating a conversational agent that is designed by the users for the users.

3 Method

We identify three main steps to delivering personalised conversation to encourage physical activity. Firstly, understanding the interesting topics of conversation, secondly, extracting information from the user to contextualise the conversation and finally, generating non-repetitive responses to encourage long term engagement. In this section we detail our methods applied to realise these steps for FitChat.

3.1 Identifying Conversational Skills

We adapt co-creation methodology to identify the most effective conversational skills expected in a

[1] https://www.lark.com/outcomes

Table 1: Natural Language Understanding skills

Skill	Information	Conversation Type
Personalisation	name, age, gender, height, weight and location	question-answer
Weekly Goal Setting	daily step goal, activity plan for each day of the week	semi-structured/open-ended
Daily Activity Reporting	number of steps, activity and the duration, the reason for not doing a planned activity	semi-structured/open-ended

conversational AI for encouraging PA among older adults. We followed an iterative refinement process (Augusto et al., 2018) and conducted three workshops where the intended stakeholders from the community were invited to participate. Each workshop was held one-month apart allowing the stakeholders to learn the capabilities of the technology and explore and refine requirements iteratively. It was specifically significant given the novelty of the conversational technology within the intended age group.

Workshop 1 introduced participants to the study and the concept of voice based conversational interventions. We sought their views on skills including goal setting and reporting that were proposed by the research group to "break the ice" and start the conversation. In workshop 2 a participatory method was followed (Leask et al., 2019). Role playing (Matthews et al., 2014) activities among workshop participants helped understand expectations where the aim was to observe the forms of natural dialogue that transpired between pairs. Participants designed conversation interactions that would allow a user to record their daily activities or that would allow a user to set goals for the coming week. Workshop 3 reviewed and refined the conversational interactions discussed in the previous workshop. Together with the research group, stakeholders prioritised the list of conversations skills identified during the workshops and short listed the top five skills that form the final prototype of FitChat. Workshop 3 also encouraged the participants to propose a name for the conversational AI where they selected the title "FitChat", inspired by the Doric term "Fit like?" (Hello, How are you?). Following are the shortlist of five skills or *intents* identified during the co-creation activities.

Personalisation: The goal is to provide a personalised experience throughout the application. This skill is likely to be used only once during the initial on-boarding process.

Weekly Goal Setting: The Goal Setting intent is aimed towards making a conscious commitment to specific physical activity goals that are considered to enforce positive behaviour change (Michie et al., 2013). This skill is expected to be used at the beginning of every week.

Daily Reporting: The Reporting intent is aimed at enabling conversation about daily activities. This conversation can be aligned with the goals set for the day and would be encouraging to the user to out-perform themselves next day. This skill is likely to be used at the end of every day.

Weekly Summary: The Summary intent is aimed at providing the user a retrospective look at the of last week's goal achievement. This skill is likely to be used at the end of every week.

Exercise Coach: The purpose of the Exercise Coach intent is to guide users to perform exercises by providing exercise steps through read-aloud instructions in a conversational format. This skill is likely to be invoked multiple times (minimum of twice) a week, according to WHO physical activity guidelines.

3.2 Natural Language Understanding

Once the conversational skills are identified, we examine Natural Language Understanding (NLU) capabilities required in each skill to maintain a cohesive personalised conversation with the user. We identify that personalisation, weekly goal setting and daily reporting are the main three skills that are focused on extracting information from the user. The role-playing co-creation activity further identified the types of information each skill

should extract in order to personalise or contextualise future conversation. Note that conversation here covers question and answer forms that can be are both open or close-ended. We summarise our findings in Table 1.

3.3 Natural Language Generation

Table 2: Information for Natural Language Generation skills

Skill	Information to deliver
Weekly Goal Setting	The WHO recommendations for daily step count and physical activities.
Weekly Summary	Summary of activities and steps recorded during the last seven days
Exercise Coach	exercise steps for three types of exercises, balance, strength and flexibility.

To create contextually relevant responses, firstly we look at what information is required to deliver the response through conversation; and secondly, how we can make that conversation personalised. To address the first concern, we identified three skills essential to deliver information to the user. These are listed with the information they deliver in Table 2. We consider two aspects for personalisation; contextualisation with information extracted at the NLU phase, and generating non-repetitive responses with a corpus or message bank. Two skills stand to benefit from contextualisation as shown in Table 3 and three skills benefit from non-repetitive

Table 3: Personalisation of Natural Language Generation skills

Skill	Contextualisation Information
Daily Activity reporting	The planned daily step goal for the week, planned activities for the day
Weekly Goal Setting	age, height, weight and gender for appropriate WHO recommendations
Weekly Summary	The planned daily step goal for the week, planned activities for each day of the past week

Table 4: Non-repetitive Motivational Responses

Skill	types of motivational messages
Daily reporting	Positive message if activities or steps reported, an encouraging message based on their reason if they fail to perform a planned activity
Weekly Goal Setting	Positive message
Weekly Summary	Positive message
Exercise Coach	Positive message after each exercise set

motivational responses as shown in Table 4.

In order to ensure the non-repetitive behaviour, we adapt a similar methods to corpus-based methods used in literature (Morris et al., 2018) to develop a Motivational Message bank organised under three main categories. Firstly we create a bank of general motivational messages for when the user reports on completed activities; for instance a message such as "Well Done! Regular physical activity is really good for your well being" is uttered by the agent at the end of reporting. Secondly a set of messages to be used when a user does not perform a planned activity due to a specific barrier. Messages in the barriers category are grouped under six barriers that are commonly found in literature (these include Family, Support, Tiredness, Work, Time and Weather). The aim is to deliver a personalised and empathetic response when a user is unable to perform an activity. This message bank is integrated with Reporting, Goal Setting and Summary skills. We started with 20 messages and it is updated regularly. We include few examples from the message bank in Table 5.

4 Conversation Design

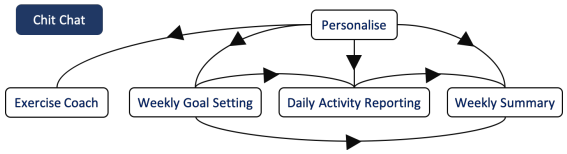


Figure 1: Conversation design pathways

Figure 1 illustrate the 5 skills we identified in Section 3.1. The arrows indicate how information

Table 5: Motivational Message Bank Examples

Achievement	Type	Barrier	Example
Positive	Steps	-	You are doing well with your steps... Keep it up!
Positive	Activity	-	Well done John, You have completed all your planned swimming sessions for this week.
Negative	Steps	-	You are taking less steps compared to last week. Think of more ways to add steps to your daily routine.
Negative	Activity	-	You have completed less exercise sessions than last week. Try to add more sessions to your weekly schedule.
Negative	Activity	Family	Try and exercise with family members. Go for a walk, play tag, put on some music and dance! You'll spend time together and increase your step count
Negative	Activity	Weather	Keep a positive attitude and embrace the weather - walking in the rain can be invigorating and you can go home for a warm shower afterwards!

extracted using NLU will be used to contextualise a conversation skill. For instance, the step goal information extracted during the Weekly Goal Setting skill will be used to contextualise the Daily Activity Reporting skill. In addition to the 5 identified skills we include a chit-chat skill such that the user is able to carry on an informal conversation about generic topics such as weather or news if needed. This will increase the usability of the conversational bot application giving the user more freedom. Next we look at each skill in depth exploring how the conversations are designed and responses are generated.

4.1 Personalisation

Personalising skill is a simple question and answer conversation that is repeated until information required is extracted from the user. The questions are designed such that the goal of each question is to extract a single piece of information. This results in easier extraction of information compared to more open-ended questions and answers. For instance, an open-ended approach would be to ask "Tell me about yourself?" and ask follow-up questions for each missing information. Instead we design a simpler approach; a list of questions such as "What is your name?", "What is your age?", "How tall are you?" to extract information independently. We argue that this approach is suitable here, because the personalising skill is typically used once at the the start.

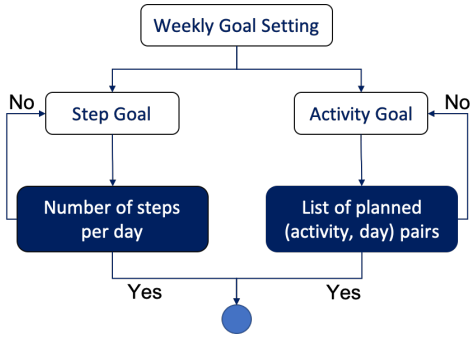


Figure 2: Weekly Goal Setting

4.2 Weekly Goal Setting

During co-creation, participants identified the limitations of fitness apps in recognising physical activities that are beyond ambulatory activities (for instance activities like dancing or tai-chi). Accordingly they proposed two types of goals; steps goal and activity goals. The conversational agent starts the conversation by understanding the type of goal the user wants to set, then guides the user towards providing information required. For a step goal, we extract the number of steps the user plans to complete each day of the week. For an activity goal, we extract one or more activities and the respective day the user plan to perform each activity.

4.3 Daily Activity Reporting

Depending on the type of goal being set for the day the conversational agent initiates a contextually relevant dialogue with the goal of extracting activities the user had undertaken during the day. For this

Table 6: Goal Setting Information Extraction

Goal Type	Agent Utterance	User Response	Information
Step	How many steps do you plan to complete a day?	around 8000 steps	steps per day = 8000
Activity	Tell me about the activities you have planned this week.	I will be swimming on Monday morning. Then some golf on Tuesday and Thursday	Activity List = [(Swimming, Monday), (Golf, Tuesday), (Golf, Thursday)]

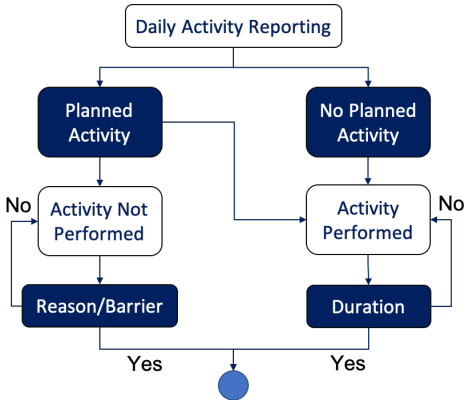


Figure 3: Daily Reporting

purpose the data obtained from the Goal Setting template is retrieved to form the context of the conversation. In Figure we can see that there are two main conversation pathways a user can be directed towards: either the user has performed one or more physical activities and they record them with the Agent, or the user has not performed any physical activities and records a reason (e.g. a barrier). At the end of a conversation pathway, the Agent is designed to respond with an appropriate motivational message from a message bank. These are selected based on the pathway and the reason (i.e. barrier) for when a user has not performed an activity and more details on the message bank. (See examples in Table 5)

4.4 Weekly Summary

Information extracted from the Goal Setting and Reporting templates provide the context of this conversation by highlighting goals achieved and reported physical activities. A motivational message is included at the end of the summary to encourage the user to maintain or improve their performance next week. Here the non-repetitiveness is preserved by diversifying the motivational messages.

"Hello {**name**}! You did {**average number of steps per day**} steps in average per day last week. This is an {**increase/decrease**} from the average from week before, {**motivational message: positive/negative**}. You also did following activities last week; {**activity**} on {**day**} and {**activity**} on {**day**}. {**motivational message: positive**}}!"

4.5 Exercise Coach

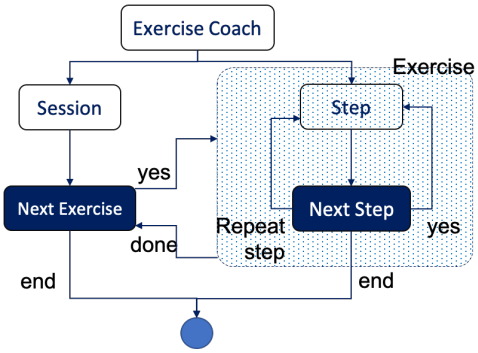


Figure 4: Conversation design pathways

Exercise coach intent can be invoked in two alternatives formats: a single exercise at a time; or a set of exercises curated by a physiotherapist as a single plan to be performed. A detailed view of the conversation flow is illustrated in Figure 4. Parts of this intent can be viewed as an instructional read-aloud function, with the added functionality to enable voice-commands that enable the user to interact in real-time. Example voice commands include: *next* (move to next step or exercise); *repeat* (repeat the current step); and *all steps* (read out the entire exercise).

5 Prototype Implementation

A robust system with minimal maintenance requirements was designed to achieve rapid proto-

Table 7: Daily Activity Reporting Information Extraction

Activity Type	Context	Agent Utterance	User Response	Information
Steps	height, gender	How many steps did you do today?	I did around 2km, not sure about the number of steps	Approximate no of steps:4500, Distance: 2km
Activity	None	Did you do any physical activities today?	Yes. I went dancing with a friend for an hour	Activity: (Dancing, 1 hour)
Activity	planned act.	Did you go swimming today?	Yes I did swim for 2 hours this morning	Activity: (Swimming, 2 hours)
Activity	planned act.	Did you go swimming today?	No, I was not feeling well, may be tomorrow	Barrier: illness

typing. The overall architecture is illustrated in Figure 5. The FitChat mobile application consists of three components: the conversational framework, cloud backend and the smart phone application. DialogFlow implements the conversational intents, while the smart phone application contains the voice based chatbot and the step counting (Traxivity) components. A cloud based micro-services architecture is used to develop the backend enabled by Firebase services. In next sections we will discuss each component in detail.

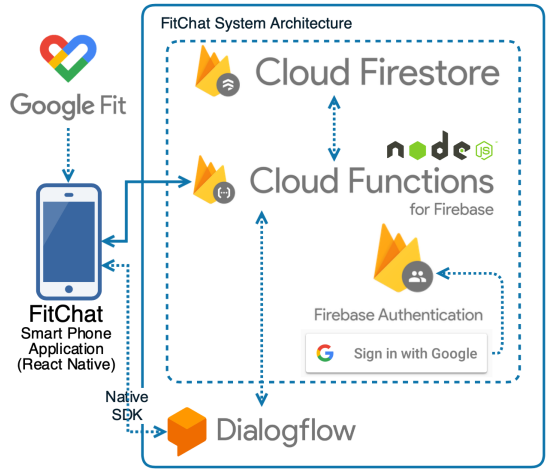


Figure 5: System Architecture

5.1 Intent Implementation

A comparative study of conversational AI frameworks was conducted before choosing one for FitChat intent implementation. We considered three frameworks; Amazon Lex, Google DialogFlow and the Open-source framework Rasa. Both DialogFlow and Lex frameworks offer the flexibility of using additional backend services and mobile integration support that enable building an end to end solution. Rasa, as a open-source platform enables the flexibility to customise conversational techniques and algorithms. We chose DialogFlow as it provided better integration flexibility with mobile applications and was more versatile due to its in-built general conversational intent library. It seamlessly linked with other Google services and most importantly the text to speech functionality was the best among the alternatives. DialogFlow intents are created by adding appropriate training phrases and responses. Each intent calls for custom "Entities" to implement Natural Language Understanding. For instance, goal setting intent defines Entity "ActivitySchedule" that extracts a list of day and an activity pairs from the user response as in Figure 6. Here a combination of regular expressions and response construct structures are used to extract information by matching against the user utterances.

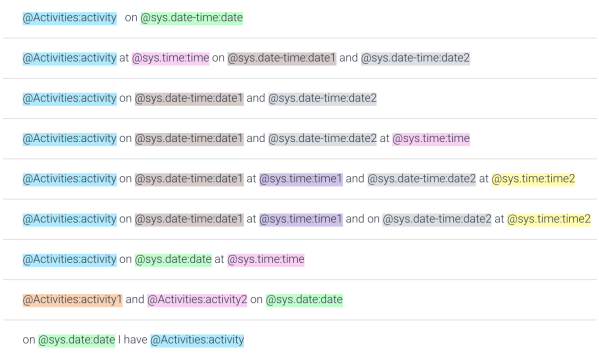


Figure 6: ActivitySchedule Entity for Goal Setting Intent

5.2 Smart Phone Application

FitChat was made available to the end-users via a smart phone application. An analysis of development support for conversational AI concluded that a cross-platform development framework such as React-native to be more suitable compared to native platforms such as Java for Android or Swift for iOS. Low time consumption and minimal development overhead were two deciding factors that emerged in our analysis in favour of React-Native. We made use of the data obtained through Google Fit APIs to record step counts, distance travelled and calorie information for physical activity for Traxivity.

User interface and user experience design of the system is crucial, specifically for the target audience of older adults; accordingly, the application was designed and iteratively refined based on feedback from co-creation workshops (Figure 7). The home screen of the application is set to the FitChat voice bot and the second tab contains the Traxivity component for physical activity tracking. Preferences and manual goal setting can be navigated through the side menu. Additionally, the FitChat bot can be customised for voice speed, pitch and the voice persona according to the user preferences.

5.3 Cloud Backend

We use Firebase which is a Backend-as-a-Service solution by Google for business logic, authentication and data storage implementation. Firebase Cloud Functions (FCF) service is used to run the backend logic which is developed using NodeJS. FCF service also facilitates Firebase Firestore database read/write access which is a flexible NoSQL database where we store user data for contextualising the conversation. The FitChat application uses the Firebase Authentication service along with Google Sign-In. Google Sign-In enables a seamless on-boarding process for the end-user eliminating the need for filling registration forms and increases security by eliminating the need to manage application specific user login details. The cloud backend is developed to be scalable and is server-less with no infrastructure maintenance.

6 Qualitative Evaluation

Think Aloud sessions were planned to evaluate the first prototype of FitChat that allowed generating real-time feedback on the intervention. Think aloud methods are frequently used for usability testing of e-Health applications (Maramba et al., 2019), and involve participants thinking out aloud whilst they perform a task, or immediately afterwards (Eccles and Arsal, 2017).

6.1 Evaluation Protocol

Seven participants took part in five think aloud sessions. Only four participants of the seven had previously participated in co-creation workshops. During the sessions, participants explored the features of the application with minimal input from the researchers. The participants were asked to discover skills and only if needed a keyword is given to initialise the conversations. The think aloud sessions were audio recorded and the feedback was arranged in to themes. Given the data privacy issues, personalisation skill was not included in the evaluation protocol, instead the users were asked to use an existing profile. Exercise coach feature was also excluded from the study given the ethics requirements surrounding study participants performing impromptu exercises without evaluating the physical readiness. Accordingly, we evaluated the efficacy of three conversational skills and observed the general feedback on the FitChat application as a whole.

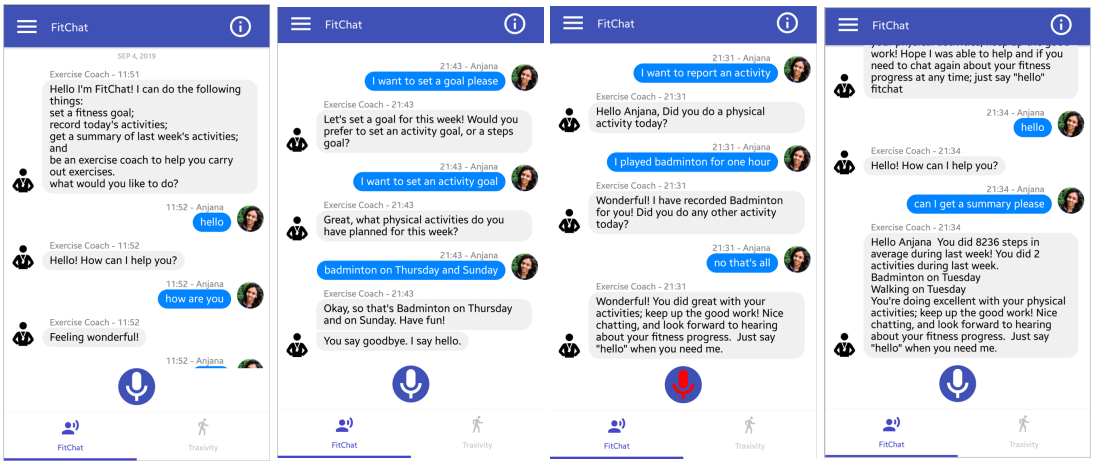
6.2 Thematic Analysis

The thematic analysis arranged the feedback from the users in to five themes; Weekly Goal Setting, Daily Activity Reporting, Weekly Summary, General Feedback and Conversation. These results highlight that all three conversational skills that were included in the this evaluation are identified as core components of the FitChat application. This further validates the selection of these skills over many other that were suggested. The participants of the study had many positive feedback for these features as well as some pointers for improvement as below.

6.2.1 Weekly Goal Setting

Participants generally responded positively to the goal setting feature. However, they commonly said "I want to set an activity/steps goal", rather than "I want to set a goal", and then selecting an activity goal, as instructed in the step-by-step dialogue. Participants commonly suggested that the goal setting feature could be improved if their goals could be stored for longer than one-week:

> *P5:"[be]cause its easier setting up like that and then cancelling"*



(a) FitChat (b) Weekly Goal Setting (c) Daily Activity Reporting (d) Weekly Summary

Figure 7: FitChat Android Application

P6:"yes.. would be useful to set this up for a longer period"

Another suggestion was to include a reminder feature using the goals that are created:

P1:"I've got Pilates on a Tuesday, if it could remember every Tuesday and it would remind me"

6.2.2 Daily Activity Reporting

Participants responded well to the reporting feature and were particularly impressed that the app could link their reported activities with their activity goals, and the motivational feedback:

P1:"I like the 'well done...' I think that's important to say well done, you've achieved that"
P2:"I think the very fact that you have to report what you have or haven't done if you set a goal would kind of make me get out of the sofa"

Most participants suggested that further detail should be added to the activities that are reported to the FitChat application such as calories burned or intensity:

P2:"slightly more detail, in that was it a leisurely swim, I don't know how you're going to phrase it, or was it a power swim?"
P2:"we could walk 10,000 steps but strolling does nothing for us, so does that come into it?"

6.2.3 Weekly Summary

Feedback was overwhelmingly positive for this feature, with several participants outlining its motivational aspect. It was also suggested that it would be useful if this feature included a comparison of summaries in order for participants to reflect on differences in physical activity:

P6:"it might say well done last week you did 40 minutes, the week before you did whatever"
P7:"I think that would motivate me that I see I've done that"
P6:"by doing that you can sort of maybe set a different goal for yourself, oh I'll have to beat that next week sort of thing"

6.2.4 General Feedback

Identification of effective skills to minimise complexity is important usability aspect for conversational bots. A complex solution will introduce a learning overhead to the user which is not desirable specially among the older adults. At times, participants did not intuitively converse with the expected phrases/terms required to interact with the application; however, they quickly learned the terminology during the think aloud sessions. They acknowledged that the app was easy to use once they were familiar with the terminology but expressed that a more complex system would discourage them:

P1:"you would get into this lingo because it's obviously got lingo that you have to tap into"
P3:"I think we would learn very quickly"
P4:"it takes a wee while to know the

tricks like"
P6:"its easy, simple to use, it's just because at the moment its very word specific and restrictive"

6.2.5 Conversation

With regard to the conversational component of this app, the feedback was largely positive:

P1:"I think it's one stage up from a Fitbit definitely because you can interact with it"
P3:"it's quite powerful the speaking bit"
P4:"because you have to speak and listen , aye, you're almost admitted to yourself, it's a bit more, you take it more to heart than just clicking a button"
P6:"talking is a lot simpler I think, certainly with older people"
P7:"conversation is far more motivating"

Some participants suggested that improvements to the conversation could include a greater variety of responses. Some also expressed that they would prefer more informal language:

P4:"[be]cause you'll pay attention and kind of look forward to it's going to be a different form of praise every week"
P5:"just a small criticism, eh when I first read my summary, that feedback, I got all these fancy words"

7 Conclusion

In conclusion, we have identified that conversation has great potential to deliver effective Digital Behaviour Change Interventions to encourage physical activity in older adults. In this work we harness recent chatbot technology advances to build the voice based Conversational AI bot "FitChat". We identify the essential features of such an intervention with older adults from the community through co-creation workshops and implement conversational skills using the state-of-the-art NLP and NLG techniques available for prototyping. We evaluated the first prototype through think aloud sessions. Thematic analysis of the think aloud session outcomes suggests that voice is a powerful mode of delivering motivational content and simplicity is the key to a successful deployment. Generating authentic non-repetitive responses remains a real challenge for chat based digital interventions. Involving uses in the curation of a corpus to address this challenge is an initial solution; however we need methods to evolve and enhance this corpus through system usage; and how to do this efficiently remains an open problem. In future we plan to integrate more customised AI algorithms for NLP and NLG and further improve the quality of the conversation.

Acknowledgments

This project is funded by the GetAMoveOn Network+ (funded by the Engineering and Physical Sciences Research Council, UK (EPSRC) under the grant number: EP/N027299/1) and the SelfBACK Project (funded by the European Union's H2020 research and innovation programme under grant agreement No. 689043). We like to thank everyone involved in the project in many capacities.

References

Ivor D Addo, Sheikh I Ahamed, and William C Chu. 2013. Toward collective intelligence for fighting obesity. In *2013 IEEE 37th Annual Computer Software and Applications Conference*, pages 690–695. IEEE.

J Augusto, Dean Kramer, Unai Alegre, Alexandra Covaci, and Aditya Santokhee. 2018. The user-centred intelligent environments development process as a guide to co-create smart technology for people with special needs. *Universal Access in the Information Society*, 17(1):115–130.

David W Eccles and Güler Arsal. 2017. The think aloud method: what is it and how do i use it? *Qualitative Research in Sport, Exercise and Health*, 9(4):514–531.

Ahmed Fadhil. 2018. A conversational interface to improve medication adherence: Towards ai support in patient's treatment. *arXiv preprint arXiv:1803.09844*.

Ahmed Fadhil and Adolfo Villafiorita. 2017. An adaptive learning with gamification & conversational uis: The rise of cibopolibot. In *Adjunct publication of the 25th conference on user modeling, adaptation and personalization*, pages 408–412. ACM.

Ahmed Fadhil, Yunlong Wang, and Harald Reiterer. 2019. Assistive conversational agent for health coaching: A validation study. *Methods of information in medicine*.

Becky Inkster, Shubhankar Sarda, and Vinod Subramanian. 2018. An empathy-driven, conversational artificial intelligence agent (wysa) for digital mental well-being: real-world data evaluation

mixed-methods study. *JMIR mHealth and uHealth*, 6(11):e12106.

Calum F Leask, Marlene Sandlund, Dawn A Skelton, Teatske M Altenburg, Greet Cardon, Mai JM Chinapaw, Ilse De Bourdeaudhuij, Maite Verloigne, and Sebastien FM Chastin. 2019. Framework, principles and recommendations for utilising participatory methodologies in the co-creation and evaluation of public health interventions. *Research involvement and engagement*, 5(1):2.

Christine Lisetti, Reza Amini, Ugan Yasavur, and Naphtali Rishe. 2013. I can help you change! an empathic virtual agent delivers behavior change health interventions. *ACM Transactions on Management Information Systems (TMIS)*, 4(4):19.

Christine L Lisetti, Ugan Yasavur, Ubbo Visser, and Naphtali Rishe. 2011. Toward conducting motivational interviewing with an on-demand clinician avatar for tailored health behavior change interventions. In *2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, pages 246–249. IEEE.

Inocencio Maramba, Arunangsu Chatterjee, and Craig Newman. 2019. Methods of usability testing in the development of ehealth applications: A scoping review. *International journal of medical informatics*.

Mark Matthews, Geri Gay, and Gavin Doherty. 2014. Taking part: Role-play in the design of therapeutic systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 643–652, New York, NY, USA. ACM.

Susan Michie, Michelle Richardson, Marie Johnston, Charles Abraham, Jill Francis, Wendy Hardeman, Martin P Eccles, James Cane, and Caroline E Wood. 2013. The behavior change technique taxonomy (v1) of 93 hierarchically clustered techniques: building an international consensus for the reporting of behavior change interventions. *Annals of behavioral medicine*, 46(1):81–95.

Robert R Morris, Kareem Kouddous, Rohan Kshirsagar, and Stephen M Schueller. 2018. Towards an artificially empathic conversational agent for mental health applications: system design and user perceptions. *Journal of medical Internet research*, 20(6):e10148.

Leanne G. Morrison, Charlie Hargood, Veljko Pejovic, Adam W. A. Geraghty, Scott Lloyd, Natalie Goodman, Danius T. Michaelides, Anna Weston, Mirco Musolesi, Mark J. Weal, and Lucy Yardley. 2018. The effect of timing and frequency of push notifications on usage of a smartphone-based stress management intervention: An exploratory trial. *PLoS ONE*, 13.

Natalie Stein and Kevin Brooks. 2017. A fully automated conversational artificial intelligence for weight loss: longitudinal observational study among overweight and obese adults. *JMIR diabetes*, 2(2):e28.

Shinichiro Suganuma, Daisuke Sakamoto, and Haruhiko Shimoyama. 2018. An embodied conversational agent for unguided internet-based cognitive behavior therapy in preventative mental health: feasibility and acceptability pilot trial. *JMIR mental health*, 5(3):e10454.

SportSett:Basketball - A robust and maintainable dataset for Natural Language Generation

Craig Thomson, Ehud Reiter, and Somayajulu Sripada

Department of Computing Science, University of Aberdeen:
{c.thomson, e.reiter, yaji.sripada}@abdn.ac.uk

Abstract

Data2Text Natural Language Generation is a complex and varied task. We investigate the data requirements for the difficult real-world problem of generating statistic-focused summaries of basketball games. This has recently been tackled using the Rotowire and Rotowire-FG datasets of paired data and text. It can, however, be difficult to filter, query, and maintain such large volumes of data. In this resource paper, we introduce the SportSett:Basketball database[1]. This easy-to-use resource allows for simple scripts to be written which generate data in suitable formats for a variety of systems. Building upon the existing data, we provide more attributes, across multiple dimensions, increasing the overlap of content between data and text. We also highlight and resolve issues of training, validation and test partition contamination in these previous datasets.

1 Introduction

Natural Language Generation (NLG), particularly at the document planning level, has traditionally been tackled using template (McKeown, 1985) or rules-based solutions (Mann and Thompson, 1988; Reiter and Dale, 2000). Statistical methods have also been explored (Duboue and McKeown, 2003). More recently, it has become popular to frame the NLG problem as one of sequence-to-sequence (Seq2Seq) modelling, where the input of an encoder-decoder architecture is the data, and the target output is human-authored text (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2014; Lebret et al., 2016). A discussion of most techniques can be found in the most recent survey of the NLG field (Gatt and Krahmer, 2018).

Data2Text NLG applications that use classical rules-based methods are currently not comparable with state-of-the-art Seq2Seq applications. For Seq2Seq applications, the input data is either very shallow, or the output texts exhibit a high degree of factual inaccuracy. The task here is very different to others such as chat bots (Adiwardana et al., 2020), where the focus is on appearing human, rather than conveying concise yet relevant information.

A lot of research has been done with datasets created for the WebNLG (Gardent et al., 2017) and E2E (Dušek et al., 2018) challenges. There is also the more recent ToTTo dataset (Parikh et al., 2020). Such datasets can provide interesting sentence-level insights, although the data structures are quite simple (a single table or simple schema) and the human-authored texts are short (usually one or two sentences). This makes them unsuitable for evaluating Data2Text systems which generate summaries based on more complex data analytics. This is not necessarily because the systems are incapable of doing so, but because the data is not available as input. In addition, rules-based systems for problems based on these datasets are neither difficult or time-consuming to implement, meaning it is harder to investigate their limitations under these conditions.

The Rotowire dataset of basketball game summaries (Wiseman et al., 2017), along with the expanded (in terms of game count) Rotowire-FG (Wang, 2019), have become popular resources for investigating the generation of paragraph-sized insightful texts. Several different Seq2Seq models have been proposed, evaluated, and compared using them (Wiseman et al., 2017; Puduppully et al., 2019a,b; Wang, 2019; Gong et al., 2019; Rebuffel et al., 2020). The datasets consist of basketball box scores (see Table 1) and human-authored game summaries (see Figure 2). The example sentence in Figure 1 highlights the level of complexity in these summaries. It includes a set of average statistics for a player over multiple games, as well as the claim that this means the player 'stayed dominant'. This

[1] https://github.com/nlgcat/sport_sett_basketball

is just one sentence of many in the full summary. We found Rotowire to be unsuitable for evaluating this in its current format. However, the domain itself is suitably complex and Rotowire provides a foundation upon which we can build.

Figure 1: Example sentence from Rotowire-FG. Full text in Figure 2

> He's continued to stay dominant over his last four games, averaging 27 points, 11 rebounds and 2 blocks over that stretch.

As part of a PhD project, we are creating a hybrid document planning solution that combines rules-based and machine learning methods. Our hybrid system will use known relationships and simple rules to manipulate a predicate-argument schema based on that of Construction-Integration theory (Kintsch, 1998). It will then learn to perform parts of the NLG process that are overly complex or time consuming to manually define. Like any Data2Text system, we require sufficient data, both in terms of quantity and complexity, to provide a difficult and realistic problem. Our NLG system is not discussed further here, it was mentioned in order to illustrate the problems we encountered on the data side when implementing it. The data issues are the focus of this paper.

In this resource paper, we identify issues with the structure and quality of the existing Rotowire based datasets. We then introduce an alternative, the SportSett:Basketball database. In addition to improving the quality and quantity of data, SportSett:Basketball stores game statistics in a hierarchy which can be queried in multiple dimensions. This allows for a richer entity relationship graph, the exploration of which we hope will enable future research in this challenging area. Serving as a master source, data can be exported in a range of formats, from rich graphs for Rhetorical Structure Theory based systems (Mann and Thompson, 1988), to tables or flat files for machine translation based systems.

2 The SportSett:Basketball database

SportSett:Basketball is a PostgreSQL (The PostgreSQL Global Development Group) relational database (Codd, 1970), with (optional) object-relational mappings (ORM) written in Ruby Sequel (Jeremy Evans). It provides researchers with the ability to query and filter data, in a simple and efficient way. The process of importing data into a normalized relational database also helps to verify the data, clean it, and eliminate redundancy. By writing simple scripts, either in SQL or using the ORM, data can be easily output in the format a researcher requires for their system. We tested this functionality by creating data for a recent OpenNMT based system (Klein et al., 2017; Rebuffel et al., 2020) (see section 3 for details).

There are problems with the structure, quality and partitioning of existing Rotowire based datasets. Our investigation focuses on Rotowire-FG as it contains more games, although the underlying problems are the same in both datasets. Some of these issues are minor, such as the team a player is on being indexed by city rather than name (there are two teams in Los Angeles, the Clippers and the Lakers). Others, like the partition contamination discussed in subsection 2.3 are more serious. The JSON file format also becomes unwieldy as data size and complexity increases, especially when researchers need to perform tasks like checking claims made in generated text relative to input data. For brevity, we do not discuss every minor change we have made here. The repository which will host this data resource will include an in-depth discussion for those who are interested.

The sequential nature of a season is modelled SportSett:Basketball, with each of the 82 regular season games for each team during the 2014 through 2018 seasons. The database does support preseason and playoff games, although the data for them has yet to be imported and verified with the level of scrutiny that regular season games have. Data sources include rotowire.com, basketball-reference.com and wikipedia.com. A UML class diagram of the object-relational mappings can be found in Figure 5 in the appendix.

Other efforts have been made to improve existing datasets. Whilst what we propose here is different to dataset cleaning techniques such as those applied to the E2E dataset (Dušek et al., 2019), we do not view them as mutually exclusive. Such automated techniques could be tried on existing datasets, as well as the new SportSett:Basketball. What we propose is a more fundamental redesign of the data, using traditional data-modelling techniques.

Table 1: Example partial box score for NOP@OKC on February 4th 2015, showing Oklahoma starters. Full box scores show approx 24 players.

Player	MP	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	REB	AST	STL	BLK	TOV	PTS
Serge Ibaka	41:36	6	9	.667	1	2	.500	0	0	N/A	6	0	0	7	0	13
Russell Westbrook	40:28	18	31	.581	2	6	.333	7	9	.778	6	6	1	1	6	45
Dion Waiters	33:06	6	14	.429	0	2	.000	0	2	.000	6	2	2	1	4	12
Steven Adams	25:04	4	7	.571	0	0	N/A	0	0	N/A	10	4	0	0	2	8
Andre Roberson	11:44	0	0	N/A	0	0	N/A	0	0	2	1	0	0	0	1	0

2.1 Dimensions, Sets, Entities and Attributes

Entities in the database represent people (such as players), real objects (such as stadia), events (seasons, games, periods, plays), as well as conceptual objects (like statistics). When using the object-relational mappings, an entity will normally be represented by an object which maps to a tuple in the relational database. Attributes, such as player names or game dates, are mapped to database columns.

A dimension is any axis along which we can group or filter these entities. Dimensions can be simple or complex. A simple dimension occurs when entities can be filtered independently by one of their attributes, such as all games on a given calendar data, or all players with the surname 'Antetokounmpo'. A complex dimension occurs when entities can be arranged in hierarchical sets, such as a person-in-a-game or a team-in-a-game-period. In such cases, the entity would not make sense without both of its parent sets (a person-in-a-game makes no sense without a game, or a person). This is different to previous work on dimensionality on Rotowire which defined three dimensions of time, along with the rows and columns of the box score (Gong et al., 2019). In order to include all data which could be included in the text, we need to include all dimensions, starting from those comprised on atomic entities.

Within the complex dimensions we have identified for the NBA, entities and events at the same level within the hierarchy do not overlap. Players or teams never play in two games simultaneously, similarly, seasons are disjoint sets of games. This removes the need for a complex model of events (Allen, 1983) and is why we can model entities in such a hierarchy.

2.1.1 Sets of People

The atomic entity within this hierarchy is a single person (usually a player although it could be a coach or official). People are grouped together in teams, with teams then grouped within some form of league structure. In the case of the NBA, the league consists of 2 conferences, each containing 3 divisions, of 5 teams. Each team then has a roster of at most 17 players. These groupings are shown in Table 2.

Table 2: Data coverage, dimensions in Rotowire are also in SportSett. Since SportSett models the atomic entities of Person and Play, the entire grid can be extrapolated.

	Events				
People	**All**	**Season**	**Game**	**Period**	**Play***
League					
Conference					
Division					
Team			◇	□	
Person*			◇	△	△

◇ in Rotowire □ partially in Rotowire
△ added in SportSett
* atomic entity

2.1.2 Sets of Events

We define atomic events as plays. Each play covers a span of time where something happened in the game which caused a statistic to be counted. In most cases, either one or two players will be involved in a play, for example one player may attempt a shot which another blocks. There are, however, cases where a play refers to a whole team or to the officials. Plays are grouped into game periods, with there being 4 periods (barring overtime) in a game. Games are, in turn, grouped within seasons. The event hierarchy for the NBA is shown for the in Table 2.

These sets of events differ from any temporal dimension as whilst they are played out as an ordered list, the actual date or time does not matter, provided the order is preserved. Our database allows for querying by date, but this is an attribute of an event (a simple dimension).

2.2 Missing Attributes

Whilst it is highly impractical to align data with everything a human could possibly have said in a text

Figure 2: Human-authored basketball summary for NOP@OKC on February 4th 2015

The Oklahoma City Thunder (25-24) defeated the New Orleans Pelicans (26-23) 102-91 on Wednesday at the Smoothie King Center in New Orleans. The Thunder shot much better than the Pelicans in this game, going 53 percent from the field and 33 percent from the three-point line, while the Pelicans went just 39 percent from the floor and a meager 28 percent from beyond the arc. While the Thunder were down 57-51 at half, they had a huge second half where they out-scored the Pelicans 51-34, allowing them to steal a victory over the Pelicans. It was a big win, as they will be fighting against the Pelicans to secure one of the last spots in the Western Conference playoffs moving forward. With Kevin Durant still sitting out with a toe injury, Russell Westbrook again took it on himself to do the bulk of the work offensively for the Thunder. Westbrook went 18-for-31 from the field and 2-for-6 from the three-point line to score a game-high of 45 points, while also adding six rebounds and six assists. It was his second 40-point outing in his last four games, a stretch where he's averaging 31 points, 7 rebounds and 7 assists per game. Serge Ibaka had a big game defensively, as he posted seven blocks, to go along with 13 points (6-9 FG, 1-2 3Pt) and six rebounds. Over his last two games, he 's combined for 29 points, 14 rebounds and 10 blocks. Dion Waiters was in the starting lineup again with Durant out. He finished with 12 points (6-14 FG, 0-2 3Pt, 0-2 FT), six rebounds and two steals in 33 minutes. The only other Thunder player to reach double figures was Anthony Morrow, who had 14 points (6-11 FG, 2-4 3Pt) and four rebounds off the bench. The Pelicans got most of their production from Anthony Davis, who posted 23 points (9-21 FG, 5-6 FT) and eight rebounds in 39 minutes. He's continued to stay dominant over his last four games, averaging 27 points, 11 rebounds and 2 blocks over that stretch. Giving Davis the most help was Ryan Anderson, who came off the bench to score 19 points (7-17 FG, 3-8 3Pt, 2-2 FT), to go along with five rebounds and two steals in 28 minutes. He 's been the most reliable player off the bench for the Pelicans this season, so it was good to see him have another positive showing Wednesday. Despite shooting quite poorly, Tyreke Evans came close to a triple-double, finishing with 11 points (5-20 FG, 1-5 3Pt), seven rebounds and seven assists. Quincy Pondexter reached double figures as well and posted 10 points (4-8 FG, 2-5 3Pt) and seven rebounds off the bench. These two teams will play each other again on Friday in Oklahoma.

corpus, this does not mean that arbitrarily taking a limited set of data is sufficient. A basic corpus analysis, either manually or with information extraction tools, will show some common phrases and patterns in the text. In the case of Rotowire summaries, the texts frequently mention the day of the week on which a game was played, as well as its location (place and/or stadium). It is also common for a text to state the next opponent for both teams and where those games will be played.

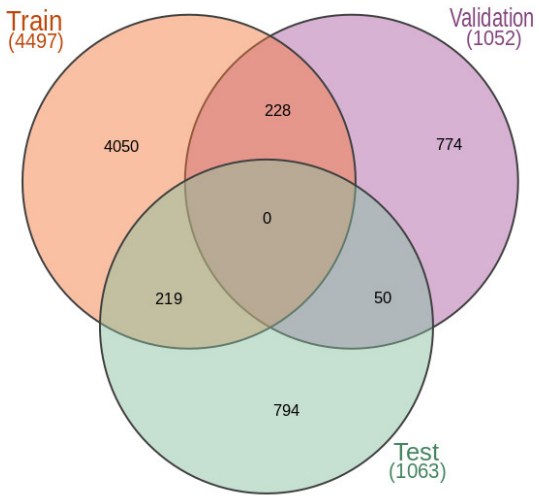
Rotowire does not include the name of the stadium where the game was played. NBA teams do not necessarily play each home game in the same stadium. For NBA International games, a team will play what counts as a home game but in a city outside the U.S. such as London or Paris. A team might also temporarily relocate due to stadium problems or construction. The SportsSett:Basketball database adds attributes for the stadium name and location, as well as more convenient methods of accessing data for previous or subsequent games.

2.3 Partition contamination

It is crucial to ensure that machine learning systems generalise for unseen data. This is usually accomplished by withholding part of the data from the training process, in order to provide for a fair evaluation. Whilst this common train/validate/test partition scheme was adopted in both of the Rotowire based datasets, there are contamination problems. Figure 3 shows the number of instances where multiple human-authored texts are present for the same game data, but are placed in different Rotowire-FG partitions. This could have occurred where summaries written by different sports journalists have been scraped for the same game. Both the surface form of these texts, as well as the opinions they

express, could be very different. As a result of this contamination, systems are evaluated (both in validation and testing) on game data which was previously used to condition the encoder. This could lead to over-fitting of the model.

Figure 3: Venn Diagram showing Partition Contamination in Rotowire-FG. Numbers represent the total games in each set. Numbers in parenthesis indicate the total size of the training, validation and test sets. The level of contamination in the original Rotowire was almost identical.



There are two problems with the existing partition scheme which need to be overcome. Firstly, the partition contamination needs to be removed. This is as simple as only including a game record in one partition, with multiple human-authored texts describing the game being allowed only within this same partition. Secondly, we need to limit contamination as much as possible when data must be used to create aggregate summaries between game events.

This second problem is more complex. Given that human-authored game summaries often include statistics aggregated over several games, it makes sense that a model might take data from more than one game as input (Gong et al., 2019). If this additional data from outwith the game is included in a different data partition then it cannot be used in this way.

We suggest that seasons remain disjoint sets when included in training, validation or test partitions. For example, we use 2014, 2015, and 2016 as training data, 2017 as validation data, then 2018 as withheld test data. This limits contamination as much as is practical and also reflects the scenario in which such a system may be deployed. A sports company such as ESPN might provide data and texts from previous seasons and expect in return an NLG system that generates texts as future seasons play out. We would ideally cross-validate, with different partition setups. However, at about 3 weeks for just one partition setup, we found compute time prohibitive.

3 Initial Experiments

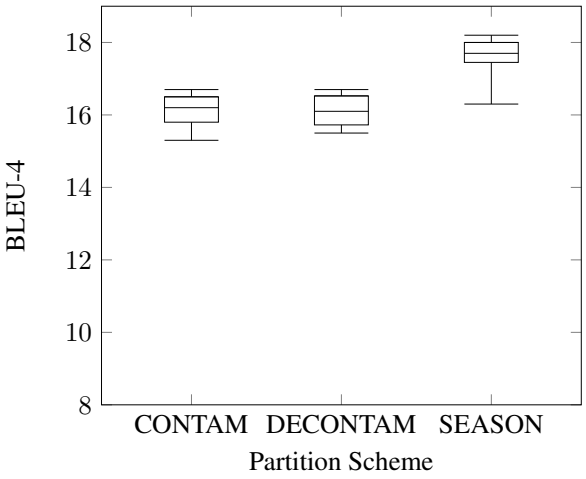
In order to confirm that our data can be easily used by existing systems, we exported it to the format of one of the more recent Seq2Seq architectures (Rebuffel et al., 2020) then attempted to replicate the results for BLEU, one of the more popular automated metrics (Papineni et al., 2002). BLEU scores do not correlate with human evaluation of text (Reiter, 2018) but may be useful in early system development. Our aim was to ensure that our data, particularly with the new partition scheme, could be used in place of the existing Rotowire datasets with minimal effort. We used the same hyper-paramaters as the original work, except for the batch size which we reduced from 32 to 16 due to hardware constraints, training 10 models with different random seeds for each of the below partition schemes:

- CONTAM: The original partition scheme of Rotowire-FG.
- DECONTAM: Starting with CONTAM, we move entires out of sets until contamination is removed.
- SEASON: Season-based partitions with 2014-16 used for training, 2017 as validation, and 2018 as test.

The data entities and attributes were very similar to the original work, with only slight differences due to minor data changes we have made when creating SportSett:Basketball. Training and evaluating these models took three weeks on a workstation containing a pair of 11GB Nvidia 2080Ti RTX GPUs. For each dataset/seed combination we took the model which performed best against the validation set (checking this every 2000 steps up to 30,000). Weights were then averaged over the previous 4 checkpoints of 500 steps each. We then calculated BLEU using the withheld test set

on each final model. Figure 4 shows the BLEU score distribution for each partition scheme.

Figure 4: Box plot showing BLEU scores for each partition scheme.



Using a one-way ANOVA with a post-hoc Tukey test we find no difference between CONTAM AND DECONTAM ($p > 0.5$). We do find the difference between each of these and SEASON to be statistically significant ($p < 0.01$). The results may, however, be sensitive to the choice of partition scheme and therefore no claims are made about the comparative status of these scores. In future, more robust quality measures will be provided.

We also calculated BLEU scores comparing the 2018 test set with a partially shuffled copy of itself (ensuring each game is matched with one other than itself, but the home team is the same). This yielded a score of 8.0 which we use as a baseline, offsetting the y-axis of Figure 4.

It is difficult to determine what effect the contamination of partitions will have had on the results reported in previous work (Wiseman et al., 2017; Puduppully et al., 2019a,b; Wang, 2019; Gong et al., 2019; Rebuffel et al., 2020). Even though the encoder may be conditioned on data which it is then tested with, the target text is different. The system would be learning the style of one author, before being measured against that of a second author. This highlights one of the key failings of n-gram based metrics, there is not only one correct gold-standard text.

The level of factual error we have observed when manually checking a small number of texts ourselves has also been quite high, although further investigation is required in order to ascertain the exact nature and extent of this problem. Metrics based on the overlap of n-grams tell us very little about whether a text has described the relationships between entities across different dimensions correctly.

4 Discussion

The data matters in Data2Text, irrespective of which system architecture is being evaluated. SportSett provides an increased volume of data, as well as an improved structure. The database also allows for researchers to easily query data, from many different dimensions, for output in a variety of formats for different architectures.

Future research will focus on the effect the dimensionality of data outlined in this paper has when generating statistical summaries in the sport domain. This will be investigated both with our hybrid architecture, and Seq2Seq systems. We plan to expand the database to include more sports, since game summaries may differ between them. There are often 75 or more scoring plays in a basketball game, meaning individual plays are usually not mentioned in game summaries unless they occurred on or near the expiration of the game clock. This differs greatly from NFL games (American Football) where even fifteen scoring plays would be considered high. As a result, individual plays feature more heavily in the narrative. We also plan to include human-authored game preview texts which describe upcoming games.

We hope that both the database, along with some of the ideas presented in this paper can be adopted by other researchers looking to solve this complex problem.

Acknowledgments

This work is funded by the Engineering and Physical Sciences Research Council (EPSRC), which funds Craig Thomson under a National Productivity Investment Fund Doctoral Studentship (EP/R512412/1).

We would like to thank our reviewers, as well as the NLG (CLAN) Machine Learning reading groups at the University of Aberdeen for their helpful feedback on this work. We would also like to thank Moray Greig, who was our basketball domain expert.

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot.

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

E. F. Codd. 1970. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387.

Pablo Ariel Duboue and Kathleen R. McKeown. 2003. Statistical acquisition of content selection rules for natural language generation. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 121–128.

Ondřej Dušek, David M. Howcroft, and Verena Rieser. 2019. Semantic noise matters for neural natural language generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 421–426, Tokyo, Japan. Association for Computational Linguistics.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2018. Findings of the E2E NLG challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 322–328, Tilburg University, The Netherlands. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *J. Artif. Int. Res.*, 61(1):65–170.

Heng Gong, Xiaocheng Feng, Bing Qin, and Ting Liu. 2019. Table-to-text generation with effective hierarchical encoder on three dimensions (row, column and time). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3143–3152, Hong Kong, China. Association for Computational Linguistics.

Jeremy Evans. Sequel.

Walter Kintsch. 1998. *Comprehension : a paradigm for cognition*, 1st edition. Cambridge University Press.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. Association for Computational Linguistics.

W.C. Mann and S.A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281. Cited By 820.

Kathleen R. McKeown. 1985. Discourse strategies for generating natural-language text. *Artif. Intell.*, 27(1):1–41.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. Totto: A controlled table-to-text generation dataset. *arXiv preprint arXiv:2004.14373*.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019a. Data-to-text generation with content selection and planning. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019b. Data-to-text generation with entity modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2023–2035, Florence, Italy. Association for Computational Linguistics.

Clément Rebuffel, Laure Soulier, Geoffrey Scoutheeten, and Patrick Gallinari. 2020. A hierarchical model for data-to-text generation. In *Advances in Information Retrieval*, pages 65–80, Cham. Springer International Publishing.

Ehud Reiter. 2018. A structured review of the validity of BLEU. *Computational Linguistics*, 44(3):393–401.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

The PostgreSQL Global Development Group. PostgreSQL.

Hongmin Wang. 2019. Revisiting challenges in data-to-text generation with fact grounding. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 311–322, Tokyo, Japan. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

A Appendices

A.1 UML Diagram

To highlight the number of tables, attributes and relations, we have included a UML Class diagram for the Ruby ORM (see Figure 5. The data structure may change in the future as it is refined and augmented. Please see the repository for the most up to date version.

The Ruby Sequel library consists of Object-Relational Mappings with database table migrations. Researchers can either use SQL directly, which should be very fast, or the ORM, which will be slower but perhaps more intuitive for some. It currently takes about 1 hour on a laptop to export data to the OpenNMT format we used in our experiments.

A.2 Detailed responses to reviewer questions

We had two questions from reviewers which we wanted to address in more detail, but would have disrupted the flow of the paper had we included them directly in any of the sections. Both questions are interesting, and worth addressing, we thank the reviewers for them.

A.2.1 Should cross-fold partitions be used?

The short answer is we believe so, but it would take too much time. Whilst we maintain that partitions should not cross season boundaries in this dataset, systems would ideally be evaluated using different combinations of seasons for training, validation and testing. Our selection of 2014-16 for training, 2017 for validation and 2018 for testing was only one possible setup. If we had evaluated with different partition setups we would have perhaps been able to determine if the per-season partition BLEU score in Figure 4 was an anomaly or a generally seen increase. The problem was that our setup for this paper took about 3 weeks of compute time. Testing additional partition setups was therefore not practical in the time frame we had.

A.2.2 What is the performance difference between this resource and the previous one?

This is a tricky question to answer because it would depend on what the resource was being used for, as well as whether raw SQL or the ORM was used. It also depends on the implementation of code which would read the previous JSON format. When using SQL, some queries will be significantly faster than with JSON. For example, for a separate project, when fact checking texts manually, we encountered a sentence like 'The Wizards came into this game as the worst rebounding team in the NBA this season'. Whilst possible to check with the old JSON format, it would be both slow and difficult. A short SQL query was able to find this information quickly because all keys are indexed. The ORM is inherently slower than raw SQL, being a wrapper layer on top of it. However, given that users can use either SQL, the ORM, or a combination of both (raw SQL within the ORM), we feel we have improved data quality, and ease of use, without sacrificing efficiency. The data is not meant for production systems, it is designed for research. Therefore, provided that data can be generated for experiments within a couple of hours (with the ORM), we feel this is sufficient. By adding more SQL this time would come down, but it would take a little longer to implement the export script. The time taken for data processing is still likely to be much less than the downstream compute time and therefore should not cause unreasonable disruption to any research project which uses it.

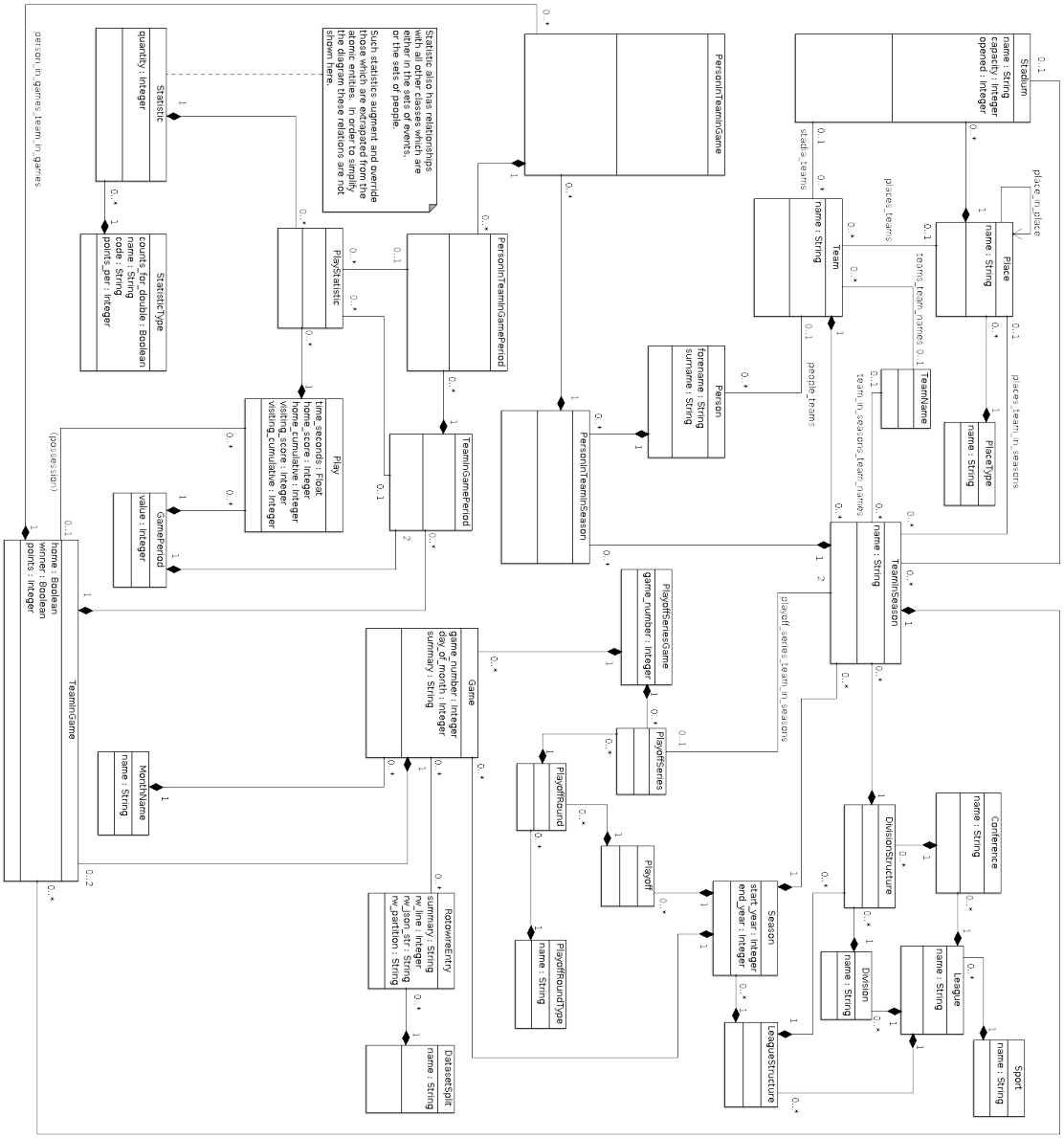


Figure 5: Entities in our database shown as a UML Class Diagram. Classes and named relations are all mapped to individiaul SQL tables.

Iterative Neural Scoring of Validated Insight Candidates

Allmin Susaiyah[1], **Aki Härmä**[2], **Ehud Reiter**[3] **and Milan Petković**[1]
[1] Eindhoven University of Technology, Netherlands
[2] Philips Research, Eindhoven, Netherlands
[3] University of Aberdeen, Scotland

Abstract

Automatic generation of personalised behavioural insight messages is useful in many applications, for example, health self-management services based on a wearable and an app. Insights should be statistically valid, but also interesting and actionable for the user of the service. In this paper, we propose a novel neural network approach for joint modeling of these elements of the relevancy, that is, statistical validity and user preference, using synthetic and real test data sets. We also demonstrate in an online learning scenario that the system can automatically adapt to the changing preferences of the user while preserving the statistical validity of the mined insights.

1 Introduction

Recently, many health and fitness apps have stormed the market claiming to be able to improve user behaviour by playing the role of an artificial health or fitness agent (Hingle and Patrick, 2016; Higgins, 2016). While the customer base for these apps is in billions, it is still a question if they are effective in doing what they claim. One goal of these applications is to help the user understand the own behaviour by giving actionable insights and advises. In this work we focus on comparative insights that can be considered as categorical statements about a measure in two contexts, for example, stating that a measure X is larger in context A than in context B, see Härmä and Helaoui (2016).

For this, the task of determining if two samples are statistically significantly different is frequently performed. While parametric and non-parametric significance tests have been widely used for such tasks, it remains a challenge to include them into a neural learning pipeline that is both scalable and user-centric. A neural network can act as a universal function approximator and can transfer knowledge from one domain to another. In this work, we consider three domains, namely, statistical significance domain, *interestingness* domain and validity domain. The statistical significance domain includes a non-parametric significance test, namely, the Kolmogorov-Smirnov (KS) test. The interestingness domain that incorporates how a user is interested in knowing about a particular comparative insight. The third domain is the validity of the content for the target application. The system should not produce insights or advises that are harmful to the healthcare goals of the service. This can be best guaranteed by a system where all texts are selected from a pre-generated and manually curated collection of *validated insight candidates*, similarly to the PSVI method introduced in Härmä and Helaoui (2016).

In this work, we train a self-supervised neural network that can be a scalable alternative to traditional non-parametric tests (with 92% accuracy at 5% alpha) and we also show how it can be used to learn user preference on top of statistical significance using an online learning strategy. As these characteristics are essential for highly scalable behavior insight mining (BIM) that finds application is fitness coaching, office behaviour (O'Malley et al., 2012), behaviour change support systems (Braun et al., 2018; Sripada and Gao, 2007), and business insight mining systems (Härmä and Helaoui, 2016), the proposed work is highly relevant.

2 Background

2.1 Desirable Characteristics of Insights

Based on recent literature, an insight should have the several, characteristics, namely, statistical significance (Agrawal and Shafer, 1996; Härmä and Helaoui, 2016), interestingness or personal preferences (Freitas, 1999; Fayyad et al., 1996; Su-

Comparison	Example
time-specific	On **Weekdays** you walk less than on **Weekends**
parameter-specific	Your **heart rate** is higher on Mondays than other days
event-specific	**when you bike**, you spend less calories per minute than when you **run**

Table 1: Examples of comparative insights in BIM

darsanam et al., 2019; op den Akker et al., 2015; Härmä and Helaoui, 2016), Causal confidence (Sudarsanam et al., 2019), surprisingness (Freitas, 1999), actionability or usefulness (Freitas, 1999; Fayyad et al., 1996), syntactic constrains (Agrawal and Shafer, 1996), presentatability (op den Akker et al., 2015) timely delivery (op den Akker et al., 2015), and understandability (Fayyad et al., 1996). Among all of these characteristics the most common ones are statistical validity and interestingness.

2.2 Types of Insights

1. Generic insight: These are insights that talk about a rather common or scientific phenomenon. These are not grounded on the user's behaviour. For example: Excessive caffeine consumption can lead to interrupted sleep as can ingesting caffeine too late in the day.

2. Personalised (Manual/Automated) insight (Reiter et al., 2003): These are insights that are tailored to the user either by a human-in-loop or by an algorithm.

    - Absolute insights: These insights talk about user behaviour in one context. We do not focus on such insights in this paper as they are less actionable.
    - Comparative insights: These insights compare the user behaviour between two contexts (Härmä and Helaoui, 2016) as shown in Table 1.

2.3 Insight Generation Mechanisms

Thousands of insights can be generated from even a simple database by slicing and dicing the data into different views. For example, to generate the insight "On Weekdays you sleep less than on Weekends", the database should have logs of user's sleep duration and corresponding dates. The rows of the database corresponding to weekdays are considered as bin A and those corresponding to weekends are considered as bin B. Relevant filters are used to extract these rows. On comparing the average user's sleep duration in each bin, we find that bin A has a lower value than bin B. Subsequently, a statistical significance test is performed to prove its statistical validity. Similarly, many comparisons could be made between two periods such as:

- Mondays and other days
- Workdays and holidays
- February and March

A detailed description of how insights are generated is explained in Härmä and Helaoui (2016).

2.4 Non Parametric Statistical Significance Tests

The data extracted from the two periods mentioned above come from two non-parametric sample distributions. The two most commonly adapted techniques to determine the statistical significance of such distributions are KS test and Mann-Whitney U (MW)test. The former is based on the shape of the distributions and the latter is based on the ranks of the samples. In this paper we choose the KS test arbitrarily. However, the MW test can also be used instead of that.

2.5 Neural Statistics

Neural networks have been used for wide range applications in Machine Learning such as signal de-noising, image classification, stock prediction, and optical character recognition. The ability of the neural network to learn basically any complex function makes a *universal function approximator*. The simplicity in the way by which a neural network generates an inference makes it a suitable choice for many applications. Additionally, the transfer learning capability of the network (Tao and Fang, 2020; Long et al., 2015; Mikolov et al., 2013) allows us to transfer the pre-learned knowledge of the network to solve different and more complex problems. This inspired us to use the neural network to approximate the statistical significance test.

2.6 Online Learning of User Preference

By permuting different contexts one may often find a large number of statistically significant insights but not all of these insights are useful to the user. Hence the user's preference must be considered before presenting the insights to them. The personal preferences of end-users change with time. Filtering the insights based on statistical validity alone is not sufficient to satisfy their interests. A method to learn a user's preference in a convenient and flexible manner will solve this problem. Online learning technology can train models in a flexible manner while still being deployed in product (Settles, 2009, 2011). There is no existing literature on online learning of user preference nor the learning of statistical validity. Such learning will be of great use in BIM applications.

In this work, we present an online learning strategy that learns user preference while simultaneously maintaining the ability to realise the statistical significance. In our technique, we assume that the user is interested only in one type of insight at any point in time. However, in reality, the user might be interested in multiple types of insights simultaneously. We set this limitation for the sake of simplicity and demonstration only, and by no means is it a limitation of our method.

3 Methodology

The entire methodology was performed in two stages, namely, the self-supervised learning stage and the online learning stage. Although each stage has a different data source, model architecture, training, and validation strategy, they share an important connection. The second stage model is transfer learned from the first. In this section, we describe the above-mentioned stages in detail.

3.1 Self-Supervised Learning Stage

As a first stage, we conceptualised and developed a neural network model that learned rich feature representations to determine the statistical validity of comparative insights. We achieved this by training the model with highly diverse synthetic data. The data generation and model training are described below.

3.1.1 Problem Formulation

Let us consider an insight i that compares two distributions d_1 and d_2. The KS significance test can be represented as a function $f(d1, d2)$ that determines the p-value of d_1 and d_2. If the p-value is less than the significance level α, then, d_1 and d_2 are considered significantly different. We formulated a neural network N that approximates f as shown in Equation 1.

$$f \sim N \tag{1}$$

The neural network learns the function f by minimising the mean squared error loss function J_1 as shown in Eq 2.

$$J_1(\theta) = \frac{1}{n}\sum_{i=1}^{n}(f(d_{1i}, d_{2i}) - N_\theta(d_{1i}, d_{2i}))^2 \tag{2}$$

3.1.2 Data Generation for Base Model Selection

A data-set containing 300000 pairs of histograms of uniform distributions was generated using the NumPy-python package. The number of samples, mean and range of each distribution was chosen randomly. The ground truth labels for each pair of distribution were generated using the p-values of the two-sample KS test. The SciPy-python package was used for this. We compared it with our less optimised implementation of of KS test and found it to give the same p-values. The data-set was subdivided into three equal parts, each for training, validation, and testing. We also made sure that each portion had balanced cases of significant and insignificant pairs.

3.1.3 Finalisation of Base Model Architecture

A domain-induced restriction of comparative insights is that the number of inputs is two and the number of outputs is one. Here, each input is the histogram of distribution and the output is the statistical significance. Based on previous works on similar input/output constraints (Neculoiu et al., 2016; Berlemont et al., 2015), we came up with three neural network architectures, namely, a recurrent neural network (RNNA), a modified RNN (RNNB) and a siamese network (SIAM). The schematics of the RNNA architecture are shown in Figure 1. The layers Ip1 and Ip2 are input layers, each having a fixed size of 100 elements. The layers F1 and F2, are fully connected layers, each with 50 neurons activated by a Leaky Rectified Linear Unit (ReLU) function. In fact, all layers in the network except the Final layer are activated by the Leaky ReLu function. Another level of Fully connected layers, namely, F3 and F4 follow F1 and F2 respectively. We chose the number of neurons in each of these

layers to be 20, which is lesser than the preceding layer, to have a compressed representation of the input signal. This type of compression is believed to help in transforming the input from the spacial domain to the feature domain. The layers F1 and F2 are concatenated and fed to a Simple Bidirectional Recurrent Neural Network (RNN) with 100 units. The rationale behind using an RNN is that the input needs to be considered a sequence rather than a vector as the inputs belong to two different contexts. We added another fully connected layer (F5) having 100 neurons to the output of the RNN. We believe that this layer generates rich features learned from the input data. The final layer is also a fully connected layer with one neuron activated by a thresholded ReLU activation function.

The RNNB model has every layer similar to the RNNA layer, except that it has 100 neurons in the F1 and F2 layers instead of 50. This is to see if increasing neurons would increase performance for a fixed purpose and input size. The SIAM network is also similar to the RNNA architecture, except that the F3 and F4 layers are subtracted rather than being concatenated and the RNN layer is replaced by a fully connected layer with 100 neurons.

3.1.4 Base Model Training and Testing

We trained and validated the three models in a self-supervised manner using the pairs of uniform distributions (histogram). The histogram was squeezed to 100 bins and the minimum and maximum range of histograms are fixed to be the minimum and maximum range of the dataset. This allows all the histograms to be comparable. Uniform distributions were chosen due to their close resemblance to real data that is commonly encountered in insight mining tasks. In total, each of the training, validation and testing phases consisted of 100000 data samples. The training was governed by Adam optimiser with a mean-squared-error loss function. The model that gave the best performance on the test set was considered as the base model. However, in real life, the data could also arise from complex or mixed distributions. Hence we proceeded further with another level of fine training.

3.1.5 Improving the Base Model

To enhance the base model we trained it with more diverse pairs of distributions (histogram) such as Gamma, Gumbel, Laplace, Normal, Uniform and Wald. On the whole, a total of 360000 pairs of distributions were generated and were equally split into training, validation and testing sets. Each of these sets consists of 120000 pairs of distributions (20000 pairs of each distribution). Both inputs of the network are always fed the same type of distribution, but with different parameters. For example, if one input of the network is a normal distribution, the other input is also a normal distribution but with different mean, range, and cardinality. The training labels are generated earlier. The training was governed by Adam optimiser with a mean squared error loss function. Once trained, the model can be used as a smart alternative to statistical significance testing to filter significant insights among all insights.

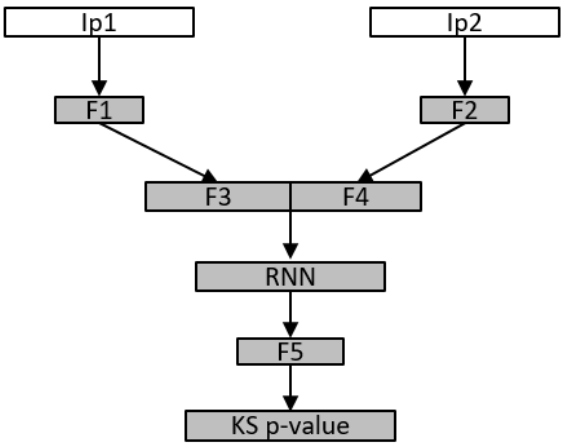


Figure 1: self-supervised neural network architecture for significance testing

3.2 Online Learning Stage

In this stage, we transformed the base model to detect interesting insights while preserving its ability to detect significant insights.

3.2.1 Problem Formulation

In this stage, apart from two distributions d_1 and d_2, we are also interested in the user model ϕ. The user's preference can be represented by a function $p_u(k)$ that generates an interestingness value for a given insight k. This function can also be considered as a user interestingness/preference model. We formulated a transfer learning approach that uses a portion of network N i.e, N' and augments it with features representations generated from another neural network Δ that uses the state vector s of the insight k. Finally, the augmented network drives the overall network O that approximates $p_u(k)$ shown in Equation 3.

Insight	Are you interested to see more of these type of insights?
On Weekdays you walk less than on Weekends	◯
Your heart rate is high on Mondays than other days	◯
when you bike, you spend less calories per minute than when you run	◯

Table 2: A Sample insight feedback form

$$p_u(k) \sim O(N^{'}(d_1, d_2), \Delta(s)) \tag{3}$$

The neural network learns the function p_u by minimising the mean squared error loss function J_2 as shown in Eq 4.

$$J_2(\theta) = \frac{1}{n}\sum_{i=1}^{n}(p_u(k) - O_\phi(N^{'}(d_{1i}, d_{2i}), \Delta(s_i))^2 \tag{4}$$

In this work, we show that any improvement in approximating p_u does not have an impact on the approximation of f in Equation 1.

3.2.2 User Model Acquisition

The online learning strategy detects more interesting insights without being instructed by the user explicitly. It uses a feedback form in a mobile application that displays a few insights that were scored high by the base model. The users may choose the insights that they are interested in and the system learns from it. A sample feedback form is shown in Table 2. In this work, we simulated the user preferences to change every month as its tracking is a problem by itself.

This feedback is equivalent to "labeling" in traditional online learning theory. To generate the insights so that our online learning system can be validated, we obtained sleep and environmental sensor data collected from a bedroom of a volunteer over a period of 4 months from May 2019 to August 2019. We logged various parameters such as the timestamp of the start of sleep, sleep duration, sleep latency, ambient light, ambient temperature, ambient sound and timestamp of waking-up. We generated insights for each day of the user using the procedure explained in (Härmä and Helaoui, 2016). The insight texts talk about the two contexts that it compares and an expression of the comparison. The number of insights per day varied between a few hundred to few thousand. We simulated the user preference given below by automatically filling the feedback form for each day.

1. May: The user is interested in Insights related to Weekdays.
2. June: Weekend insights are interesting to the user.
3. July: The user prefers to know more about his sleep duration.
4. August: The user is again interested to know if he/she is doing well on weekends.

All statistically significant insights per day on a given month that satisfy the corresponding preference criteria were labeled with interestingness score 1 and otherwise labeled 0. Since neural networks understand only numbers, we encoded each comparison insights into a single dimension binary vector s containing 220 elements where each element correspond to one parameter of comparison. For example, one element corresponds to each day of the week. Hence, if the comparison is related to Mondays and weekends, the elements corresponding to Mondays, Saturdays, and Sundays are assigned a binary one and the rest are assigned zero. We inject this vector to the model while transfer learning for interestingness recognition. In the following subsection, we explain how the model is transfer learned and how the online learning pipeline is implemented and evaluated.

3.2.3 Transfer Learning

Transfer learning was performed to enable the model to learn insight interestingness in addition to significance. The self-learned model was frozen from the input layers up to and including the F5 layer. The vector s is passed as input to another fully connected layer F6 with 100 neurons. This layer is concatenated with the F5 layer as shown in Figure 2. The concatenated layers are fed to another fully connected layer F7 having 100 neurons. While the layer F6 is linearly activated, the F7 layer is activated by the ReLu function. Finally, the output layer is a single neuron fully connected layer activated by a sigmoid activation function. Notice that the final layer is activated by a sigmoid function as this is a binary classification problem trained on user preferences instead of significance.

By performing this transfer learning, the model retains the features that correspond to the significance and simultaneously recognise interestingness of insights based on user preference.

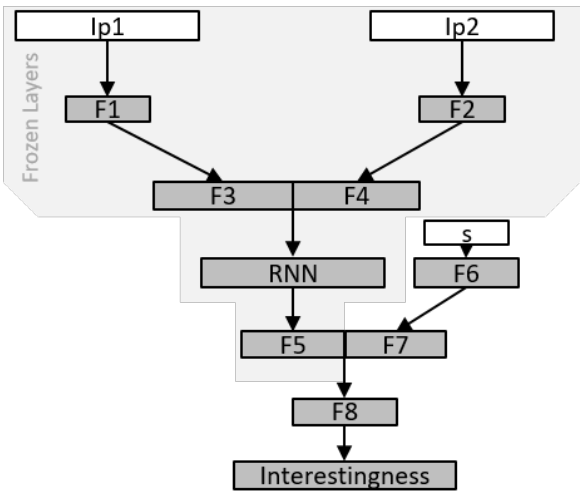


Figure 2: Augmenting the base network for online learning

3.2.4 Learning Modes

The architecture of the online learning scheme is presented in Figure 3. The scheme is executed in two modes, namely, accelerated learning mode and normal learning mode. These modes determine how much the models are trained (iterations). The accelerated learning mode, by default, starts from the first usage of the insight generator for the first ten days. Then, the normal mode begins. During the accelerated learning mode, the model learns more rigorously and during the normal mode, it learns at a normal phase. This is achieved by varying the number of iterations of training for each day. This the accelerated training mode has more iterations of training.

3.2.5 Training and Validation Switch Logic

Every day, the insights are assigned an iterestingness value based on user feedback and are scored by the model. Based on the learning modes, two scenarios can happen that impacts whether the insights are used to train or validate the model.

1. If the system is in accelerated training mode and the insight has a prediction error of less than 0.3. The training and validation switch pushes a copy of the insight to both training and validation pools. Therefore, the model trains and validates these insights.
2. If the mode is the normal training mode
   - If the prediction error is less than a preset threshold (0.10) and 50% random chance is satisfied and the fraction of interesting insights in the validation pool (if updated) will be between 0.42 to 0.6, the switch pushes the insight into the validation pool.
   - Else, if the percentage of interesting insights in the training pool (if updated) will be between 42% to 60% (arbitrarily chosen), the switch pushes the insight into the training pool.

If the user does not give any feedback, the insights continue to get pooled and trained based on the older feedback. This implicitly assumes that the user's preference is unchanged. However, we allow a small error to occur so that the system also has the ability to pick other insights at times instead of strictly catering to the user preference.

3.2.6 Pool Maintenance Logic

Both the pools are maintained to hold only a maximum limit of days of data. We fixed this arbitrarily to be 14 days. Here we assume a user's interestingness remains fairly unchanged for a period of two weeks. Every 20 days, the model forcefully pops 7 days of data in a FIFO fashion. This helps to avoid overloading the training and validation pools and forgetting older preferences. Additionally, the validation pool is completely emptied at the beginning of the first day of the normal learning phase.

3.2.7 Update Logic and Metrics

At the end of every day, a copy of the model is trained on the training pool and validated on the validation pool. If the validation accuracy exceeds a set limit (here 70%), the old model is replaced by the recently trained model. However, as an exception in the accelerated learning mode, the model is updated every day irrespective of its performance. This purposefully over-fits the model to the insights during accelerating learning mode. The performance of online learning is monitored using statistical measures, namely, sensitivity, specificity, and accuracy in predicting the interestingness of insights. Additionally, we introduce the significance preservation score, which is calculated as shown in Equation 5.

$$P_s = N_a/N_p \tag{5}$$

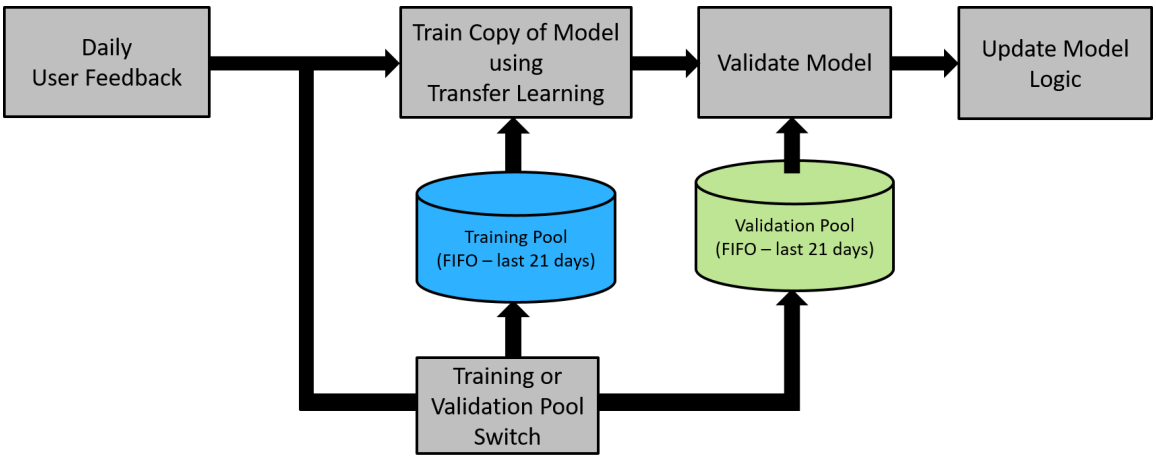


Figure 3: Online learning through user feedback

where, N_a and N_p are the number of actual interesting insights in the validation pool and the number of predicted interesting insights during validation, respectively. The P_s is not defined when N_p is zero. This is a limitation of the metric.

4 Experimental Results ad Discussions

In this section, we present the results that we obtained at each stage.

4.1 Choosing The Base Model Architecture

An example of histograms of significant and insignificant pairs of normal distributions is shown in Figure 4. It also demonstrates the variation of magnitude, range and cardinality (more samples have a smoother curve) of the synthetic data. Each of the base model architecture, namely, RNNA, RNNB, and SIAM were Trained, validated and tested using the dataset containing only normal distributions. The performance of each model is presented in Table 3. We observed that the RNNA model exhibits a test accuracy of 92% in predicting whether an insight is interesting or not. The performance of RNNA is thereby comparatively better than that of RNNB. This shows that more neurons do not always lead to improved performance. Also, RNNA exhibits slightly better performance than the SIAM network. This could be due to the sequential treatment of the data by the RNN which is part of the network. Additionally, since the SIAM network has fewer neurons, it also provides evidence that lesser neurons might not help either. In our view, the neural model should have an adequate number of neurons and parameters and an explainable architecture, which is, unfortunately, missing in recent works in this field. Hence, the RNNA architecture is chosen as the base model and considered for further analysis.

Table 3: Performance of different models while training and testing with normal distribution

MODEL	DESCRIPTION	ACCURACY
$\alpha = 0.05$		
RNNA	BIDIRECTIONAL RNN LAYER	0.92
RNNB	MORE NEURONS	0.86
SIAM	SIAMAESE NETWORK	0.87

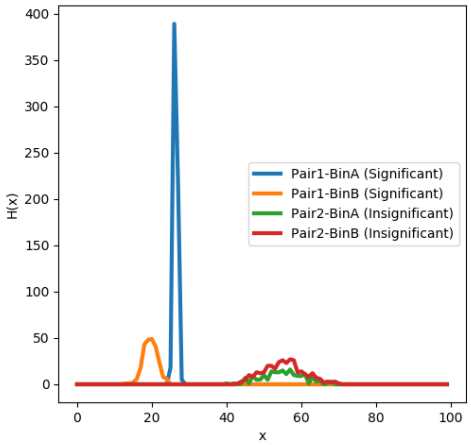


Figure 4: Pair of normal distributions without significant difference

4.2 Improving Base Model Training

We trained the base model using diverse pairs of distributions (histogram) such as Gamma, Gumbel, Laplace, Normal, Uniform and Wald. We observe that when we tested each distribution as shown in

Figure 5, we find out that the performance of the model to normal distribution remained at 0.92, but the uniform was even higher at 0.97. The worst performance was observed on Wald distribution. We have additional evidence that this is a limitation of the actual KS test that is being reflected in the neural model. It is also found that few distributions exhibit improved performances as alpha increases and few showed weaker performance as alpha increases.

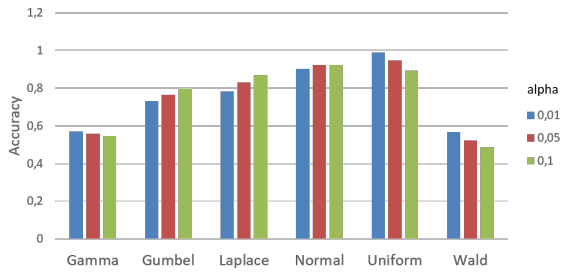


Figure 5: Gaussian trained model on mixed distributions

4.3 Online Learning

We initiated the online learning scheme and the performance metrics are presented in Figure 6. We stated the system in the accelerated learning mode for the first 10 days. It is observed that the accuracy, sensitivity, and specificity were unstable during the first 4 days of the accelerated learning phase. From the fifth day onwards, the three measures show improvement and are in the range of 0.9 to 1. The P_s measure is not defined when there are no significantly valid insights that are interesting. This is observed till day 3 and on Day 4, 100% P_s is observed. This implies that the model exhibits significance preservation starting at least from day 4 onwards. The performance is rather stable all the while during the remaining days of May and the entire June. Even though there is a transition between weekday insights and weekend insights, the model seems to adapt very well. In the months of July and August, there are visible drops in the performance around the 10^{th} day of the month even though the preference changed on the 1st of both months. This could be an instability caused due to the sudden rise in the training pool and reduction of validation pool data as shown in Figure 7. In General, the pool maintenance logic is able to control the number of training and test data points. Although the first half of July saw a huge influx of training data, the maintenance logic prevented the training pool from overloading. Otherwise, there would have been a huge chance of exposing the model to noise in the data. The mean squared error (MSE) curve shows that the error between predictions and ground truth is not very high. The MSE decreased more steeply during the accelerated learning mode compared to the normal mode. There are periodic valleys in the training pool count and validation pool count denoting the reach of the 20-day window for cleanup of the pool. Also, additional cleanups are done every day when the number of days of insights in the pool exceeds 14. All cleanups on the training and validation pool are indicated by fain red vertical lines in Figure 7.

5 Conclusions and Future Scope

In this work, we propose a neural model capable of learning the Kolmogorov-Smirnov statistical significance test and we augment architecture to learn user preference with an online-learning scheme. To model statistical validity tests, we chose a base neural model, for which three architectures, namely, a simple neural network with recurrent neural network layers with fewer neurons, similar networks with more neurons and a slightly different siamese network were investigated. The neural network with the recurrent neural network layers having lesser neurons exhibited the best performance. We continued to develop a smarter network that can not only identify an insight but also learn its interestingness in an online setting. For this, we used transfer learning and online learning approaches. We froze a part of the base model and augmented it with an additional input layer that reads a binary filter vector that describes an insight. We trained it on a real dataset while simulating user preference. The model was generally stable with few transients when the user preference changed. We were able to show that the model preserved its knowledge about statistical significance while learning interestingness. This made the network unique in an intelligent way as this is the first attempt, that a single neuron could perform more than one functionality. In the future, we would like to test the capability of the online learning module in a scenario where user preference can take multiple states at the same time.

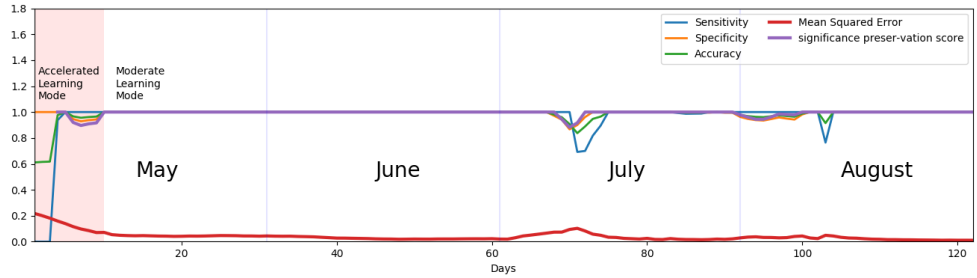


Figure 6: Timeline of Online Learning with Performance Indicators

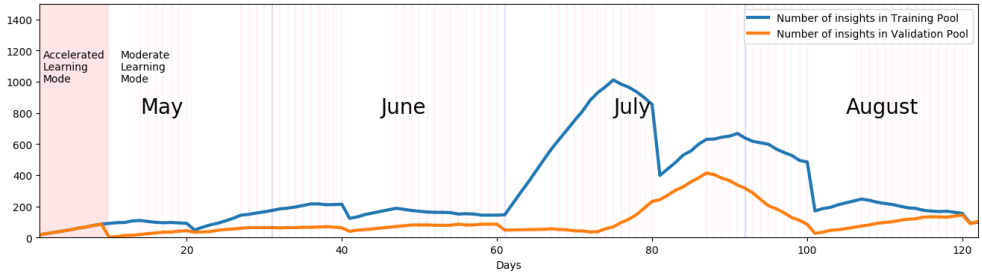


Figure 7: Size of Training and Validation Pool

Acknowledgments

This work was supported by the Horizon H2020 Marie Skłodowska-Curie Actions Initial Training Network European Industrial Doctorates project under grant agreement No. 812882 (PhilHumans).

References

Rakesh Agrawal and John C Shafer. 1996. Parallel mining of association rules. *IEEE Transactions on knowledge and Data Engineering*, 8(6):962–969.

Harm op den Akker, Miriam Cabrita, Rieks op den Akker, Valerie M Jones, and Hermie J Hermens. 2015. Tailored motivational message generation: A model and practical framework for real-time physical activity coaching. *Journal of biomedical informatics*, 55:104–115.

Samuel Berlemont, Grégoire Lefebvre, Stefan Duffner, and Christophe Garcia. 2015. Siamese neural network based similarity metric for inertial gesture classification and rejection.

Daniel Braun, Ehud Reiter, and Advaith Siddharthan. 2018. Saferdrive: An nlg-based behaviour change support system for drivers. *Natural Language Engineering*, 24(4):551–588.

Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37–37.

Alex A Freitas. 1999. On rule interestingness measures. In *Research and Development in Expert Systems XV*, pages 147–158. Springer.

Aki Härmä and Rim Helaoui. 2016. Probabilistic scoring of validated insights for personal health services. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6. IEEE.

John P Higgins. 2016. Smartphone applications for patients' health and fitness. *The American journal of medicine*, 129(1):11–19.

Melanie Hingle and Heather Patrick. 2016. There are thousands of apps for that: navigating mobile technology for nutrition education and behavior. *Journal of nutrition education and behavior*, 48(3):213–218.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. 2015. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Paul Neculoiu, Maarten Versteegh, and Mihai Rotaru. 2016. Learning text similarity with siamese recurrent networks. In *Proceedings of the 1st Workshop*

on Representation Learning for NLP, pages 148–157.

Samuel J O'Malley, Ross T Smith, and Bruce H Thomas. 2012. Data mining office behavioural information from simple sensors. In *AUIC*, pages 97–98.

Ehud Reiter, Roma Robertson, and Liesl M Osman. 2003. Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1-2):41–58.

Burr Settles. 2009. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

Burr Settles. 2011. From theories to queries: Active learning in practice. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, pages 1–18.

Somayajulu G Sripada and Feng Gao. 2007. Linguistic interpretations of scuba dive computer data. In *2007 11th International Conference Information Visualization (IV'07)*, pages 436–441. IEEE.

Nandan Sudarsanam, Nishanth Kumar, Abhishek Sharma, and Balaraman Ravindran. 2019. Rate of change analysis for interestingness measures. *Knowledge and Information Systems*, pages 1–20.

Jie Tao and Xing Fang. 2020. Toward multi-label sentiment analysis: a transfer learning based approach. *Journal of Big Data*, 7(1):1–26.

Neural Language Generation for a Turkish Task-Oriented Dialogue System

Artun Burak Mecik*, **Volkan Ozer***, **Batuhan Bilgin***, **Tuna Cakar**, **Seniz Demir**†
Department of Computer Engineering,
MEF University, Istanbul, TURKEY
{mecika, ozerv, bilginba, cakart, demirse}@mef.edu.tr

Abstract

Rapidly growing language and speech-enabled technologies contribute to the development of task-oriented dialogue systems. The demand for better user engagement has been increasing at an accelerating pace and this brings new remarkable challenges including the generation of informative and natural system utterances. In this work, our ultimate goal is to develop a Turkish task-oriented dialogue system that enables users to navigate over a map in order to get informed about dining venues that best match their preferences and make reservations based on received recommendations. This paper presents the pipeline architecture of our dialogue system with a particular focus on the language generator. We utilize an open source framework for building the components of our system and develop a sequence-to-sequence (Seq2Seq) neural model for language generation. This pioneering work is the first that proposes the use of a neural generation model in a Turkish conversational system. Our evaluations suggest that Turkish neural generation from meaning representations given in the form of dialogue acts is effective, but still in need of further improvements.

1 Introduction

In the last decades, task-oriented dialogue systems with human-like communication capabilities (Chen et al., 2017; Zhao et al., 2019) have been widely deployed in applications with commercial value such as restaurant reservation (Henderson et al., 2019) and online shopping (Yan et al., 2017). As opposed to open-domain dialogue systems without a clear dialogue goal, these systems present adequate intelligence in understanding user utterances and taking actions in response to accomplish constrained tasks. Task-oriented dialogue systems that can converse naturally with users through text or auditory conversation have received increasing attention of language and speech communities. Conventional task-oriented dialogue systems combine different modules in a pipeline architecture (Raux et al., 2005): i) language understanding (Gupta et al., 2019), ii) dialogue state tracking (Lee and Stent, 2016), iii) dialogue policy (English and Heeman, 2005), and iv) natural language generation (Zhu et al., 2019). These modules are independently trained and optimized with separate objective functions. Pipeline architectures often suffer from cascaded error propagation and a change in the output representation of a previous module also affects subsequent modules. Recent end-to-end task-oriented dialogue systems (Liu and Lane, 2018; Wen et al., 2017) mitigate these problems by training a single model directly from data without distinguishing individual modules and optimizing a single objective function. Although end-to-end systems enable multi-domain adaptation by minimizing laborious feature engineering, they unfortunately might generate generic utterances or utterances that are repetitive.

End users face utterances generated by dialogue systems and their satisfaction heavily depends on the quality and semantic coherence of these productions. The natural language generation module is mainly responsible for producing informative and fluent utterances that engage users and improve their experiences. The input to this module is often a dialog act given in a semantic form that either conveys or requests information as directed by the dialogue policy (Zhao and Kawahara, 2019). A dialogue act is a meaning representation of an action (i.e., system or user) that can be realized using one or more sentences. Depending on the action type (e.g., greeting, inform, or confirm), dialog acts contain one or more slots (attributes) of different types (e.g., numeric or string) to fulfill the meaning (e.g., *inform(name="Green Food",phone=415986223)*).

*These authors contributed equally to the work.
†Corresponding author

Early research methods of language generation for task-oriented dialogue systems include manually-crafted rules and templates. This kind of generation is adequate to cover all information captured in a dialog act, but it lacks preferred flexibility, requires heavy manual effort, and necessitates domain expertise. Although these issues hinder scalability across different domains, they can be addressed by statistical generation approaches which can learn human writing patterns directly from annotated data. Recently, neural generation models have become a common approach for joint learning of sentence planning to cover all selected information and surface realization to incorporate that content in a fluent text. However, it is not straightforward to find large amounts of domain-specific labeled data (real conversational data) for training statistical or neural generation models, and it is yet infeasible for some languages including the morphologically rich language Turkish.

In this study, we describe our efforts towards building a task-oriented dialogue system for Turkish that enables users to navigate over a map and reach descriptive information of dining venues based on their preferences until a venue is booked for reservation. The system, implemented as a mobile application, interacts with users through an interface where textual and visual modalities are employed. In the current version, all venues that match user preferences are listed on a map and the user is presented with a single sentence description of any venue selected on that map. Although our goal is to enhance this work to a venue recommendation and reservation system where more sophisticated human-like conversations can take place, the system currently engages in a limited dialogue with end users mainly due to the lack of labeled conversational corpora for Turkish in this domain. We use the RASA open-source machine-learning based framework (Bocklisch et al., 2017) to develop natural language understanding and dialogue management components of the system. We also leverage knowledge obtained from a human-annotated English conversational data in restaurant reservation domain to imitate humans while building our dialogue policies.

In this paper, our focus is on the language generation component of the system which is implemented as a sequence-to-sequence (Seq2Seq) neural model. To our best knowledge, this work is the first that utilizes a neural generation model for producing task-oriented Turkish utterances. The literature does not report any study to show how effective neural models are in generating Turkish sentences from dialog acts in terms of coverage and correspondence to human generated texts. In this study, we report the system performance using automatic evaluation metrics over our corpus of 4200 pairs of dialog acts and reference sentences collected via crowdsourcing. In our experiments, we also assess the impact of delexicalization on the quality of generated utterances where verbalizations of rare words in dialogue acts are targeted.

2 Related Work

Previous research on pipelined dialogue systems has focused on improving the performance of individual components in the architecture. Rule-based parsing methods (Denis et al., 2006), multiclass classification algorithms such as SVMs (Sarikaya et al., 2016), and deep convex networks (Tur et al., 2012) were shown to be effective in detecting user's intent. Promising results were also achieved with the use of recurrent (Yao et al., 2013) and recently hierarchical (Zhao and Kawahara, 2019) neural networks. Mapping textual spans of an utterance to slots in a dialogue act was often considered as a sequence tagging problem and quite good results were achieved with maximum entropy models such as conditional random fields (CRFs) and stochastic finite state transducers (Raymond and Riccardi, 2007). Deep belief networks (Deoras and Sarikaya, 2013), convex networks (Deng et al., 2012), and bidirectional long short-term memory networks (Jaech et al., 2016) were later shown to outperform CRF-based approaches. A variety of different approaches have emerged for dialogue state tracking. A tracker that benefits from domain independent rules and basic probability (Wang and Lemon, 2013), and a CRF-based discriminative approach (Ren et al., 2013) achieved comparable performances to machine-learning based methods. The effectiveness of neural models was also exploited for state tracking task. One pioneering work combined an RNN model with delexicalized feature representations in order to generalize it to unseen slots and values, and with an online unsupervised adaptation approach to exploit unlabeled data (Henderson et al., 2014). An RNN model was later used to train a state tracker capable of working across different domains (Mrkšić et al., 2015). Recently, dialogue state tracking was tackled as

a reading comprehension problem and addressed using an attention-based neural network (Gao et al., 2019). Reinforcement learning was heavily utilized for learning dialogue policies (Cuayáhuitl, 2017; Shah et al., 2016; Weisz et al., 2018). Recent experiments suggested that utilizing pre-trained language models in task-oriented dialogue components is a promising approach (Wu et al., 2020).

Although many generation methods have been proposed so far, they can be broadly classified into three types. Rule or template based approaches require significant expertise and human effort, and the number of manually constructed templates is limited (Jurčíček et al., 2014; Mitchell et al., 2014). On the other hand, stochastic or statistical approaches enable less monotonic generation by training a generator from data directly (Mairesse et al., 2010; Mairesse and Walker, 2011; Oh and Rudnicky, 2000). Recent developments in neural networks have enabled generation to be handled as a transformation from meaning representations to system responses via a single model. In a work that simulates the few-shot learning setting with scarce annotated data, a multilayer transformer model was trained for generating responses and generalization to new domains was achieved by utilizing pre-trained language models (Peng et al., 2020). The work of Wen et al. (Wen et al., 2015a) jointly utilized recurrent and convolutional neural networks for realizing the content of a dialog act, and the RNN-based generator that encodes one-hot representation of the dialog act as its initial state was trained with semantically unaligned data. Semantically controlled long short-term memory was also explored for training a generator from unaligned data where sentence planning and surface realization are jointly optimized (Wen et al., 2015b). A recent work employed a Seq2Seq generator with attention using GRU cells to capture the semantic content of dialog acts and used a language model to achieve naturalness in generated utterances (Zhu et al., 2019). Our work is most similar to the work of Dušek and Jurčíček (Dušek and Jurčíček, 2016) but their dialog act representation formed by concatenating triples of act type, slot name, and slot value differs from our input representation.

Turkish, a morphologically rich language with free-constituent order, has been in focus of language processing research for many years (Oflazer and Saraclar, 2018). However, Turkish language generation has been relatively less-studied up to now. Scarcity of available data and lack of annotations are some of the obstacles to developing robust systems with high performances. Previous generation literature is restricted to some well-known problems of surface form generation (Cicekli and Korkmaz, 1998; Ayan, 2000) and text summarization (Nuzumlalı and Özgür, 2014; Çagdas Can Birant et al., 2016). Recently, template-based language generation was employed in a venue recommendation system (Elifoğlu and Güngör, 2018) where a distinct template for each venue property is used. To our best knowledge, Turkish text generation from structured data has not been yet exploited. Moreover, there is no prior knowledge as to whether the use of neural models in generating utterances from dialog acts is effective or not, especially in domains with a very limited amount of annotated data. Our work reports first empirical evaluations that measure the usability and effectiveness of a neural model in this task.

3 System Architecture

Our task-oriented dialogue system is implemented as a mobile application and exhibits the traditional pipeline architecture. A user utterance is processed by three downstream components before a dialog act is transferred to the language generation component. In the rest of this section, the mobile application, and the language understanding and dialogue management components are described in detail.

3.1 Mobile Application

Users interact with our mobile application through an interface where they rely on menus that display listings of choices for different properties of dining venues. At any time while using the application, users can search for venues exhibiting different properties by choosing any of these alternatives. As shown in Figure 1-a, a user is initially asked to specify venue properties being sought (i.e., its location, customer rating, price range, and type of served food). All venues that exhibit these properties are listed on a map of the selected region (Figure 1-b) and the user can navigate between these venues. If the user selects a listed venue on the map, a single sentence description of the venue along with some of the matching properties are presented to the user in a separate window at the bottom of the screen. That description is produced by our neural generator using the meaning representation passed from the system. On this map view, the user can

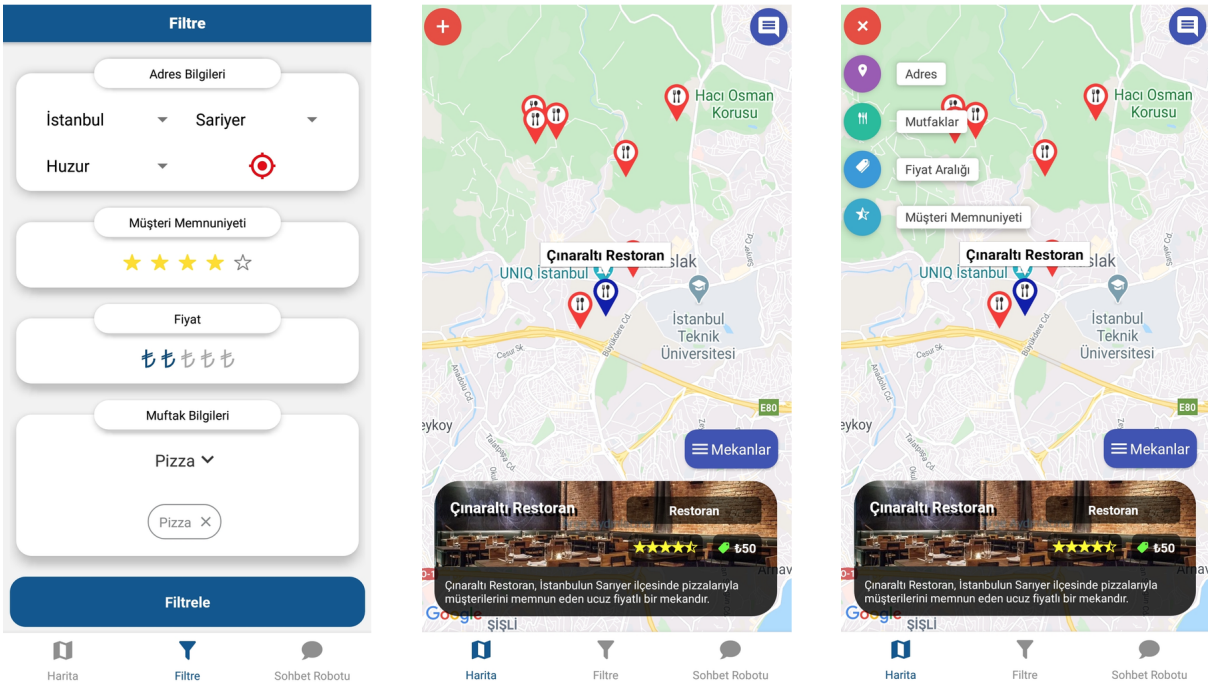


Figure 1: (a) Opening screen view (b) Map listing view (c) Map listing+New search view.

also update venue properties from the menu given on the upper left corner (the red icon) and start a completely new search (Figure 1-c). Although it is not fully implemented yet, the user will engage in a dialogue with the system over this map view (using the blue icon on the upper right corner), and get recommendations/make reservations in the future.

3.2 Natural Language Understanding

This component identifies user's intent from a given utterance by classifying it into predefined classes. Moreover, it extracts information related to that intent and uses them to fill corresponding slots. In the current implementation, we use the RASA NLU framework (Bocklisch et al., 2017) for building our language understanding component. The RASA NLU combines embeddings of word tokens that appear in a sentence in order to obtain a representation of the sentence. An SVM classifier trained on these sentences then classifies a given utterance into one or more intents. For entity extraction, the framework offers different extractors and we train a CRF extractor using our custom entities. To train a Turkish intent classifier and an entity extractor, we use our dataset and some manually translated examples from an English dataset in the restaurant domain (Novikova et al., 2017). For each sentence in our collection, we manually determine the intent and annotate text spans that correspond to different entities with appropriate tags. For instance, Figure 2 shows a sentence and a part of its annotation.

```
{"text": "İstanbul, Karaköy'de yüksek memnuniyete sahip
        kafeterya ürünleri satan mekanları arıyorum.",
 "intent": "request",
 "entities":
  {
   "start": 0,
   "end": 8,
   "value": "İstanbul",
   "entity": "region"
  },
  {
   "start": 21,
   "end": 27,
   "value": "yüksek",
   "entity": "customersatisfaction"
  }, ....
 }
```

Figure 2: An annotated training data example for NLU.

3.3 Dialogue Management

This component maintains the current dialogue state by keeping user's intents and a dialogue history (dialogue state tracker). Its main responsibility is to estimate the user's goal at each turn of the dialogue. The dialogue history is treated as an abstraction of previous dialogue turns. Moreover, it behaves as the decision maker of the whole system and takes appropriate actions according to a policy by considering the current dialogue state. Due to lack of available Turkish dialogue conversations that we can use for training a dialogue management component, we first analyze the E2E dialogue challenge dataset that consists of English conversations in the restaurant reservation domain (Li et al.,

2018). By processing the provided dialogues and manually filtering intents and entities that are out of our scope, we then compile training data for our dialogue manager. Since our focus here is to mimic natural conversations rather than modeling the language, this data collection approach enables us to train our language-independent dialogue manager with 2800 different representations of actual conversations of varying length. Using an RNN-based approach, the RASA Core dialogue engine learns policies from our training data.

4 Neural Turkish Generation Component

We develop a sequence to sequence (Seq2Seq) model (Liu et al., 2017; Sha et al., 2018) as our generation component. The model utilizes a dialog act as input and produces a single Turkish sentence to preferably convey all the information expressed in that act. Since there is no available data that we can use to train the model, we first conduct human subject experiments in order to collect a small-sized corpus as our starting point.

4.1 Corpus Collection

A dialog act is a logical representation of meaning that might be expressed using single or multiple sentences. Each dialog act contains an action type (i.e., what is intended to be conveyed by the system or user) and a set of slot-value pairs associated with that action (e.g., the properties of a venue in focus). Since our goal is to engage in dialogue with end users, restricting the system to only describe properties of a venue is not adequate. Moreover, the number of slots that might be associated with an action type is too large to be listed in a single sentence with a moderate complexity. In order to determine action types and slots that would be utilized, we explore similar well-studied datasets compiled for other languages (SFRest (Wen et al., 2015b), E2E (Novikova et al., 2017), Bagel (Mairesse et al., 2010)). Nine different action types are incorporated into the current version but these action types and slots will be populated in the future:

- **greeting:** Greet the user
- **goodbye:** Farewell the user
- **inform:** Present all properties of a venue
- **inform_only:** State the uniqueness of a venue with specified properties
- **inform_not:** State the non-existence of a venue with specified properties
- **inform_all:** Present all venues with specified properties
- **request:** Query existence of venues with specified properties
- **compare:** Compare two venues with respect to a property
- **compare_only:** Compare a venue with a number of other venues with respect to a property

One or more slots are defined for each action type as shown in Table 1. For instance, the action type inform might contain up to six slots. The values of some slots are verbatim strings whereas the remaining values are selected from a catalog.

Actions	Slots	Types
greeting, goodbye	Message	String
inform, inform_only, inform_not, inform_all, request, compare, compare_only	Name, Region, Near, Customer Satisf., Price Range, Cuisine	String String String Catalog Catalog Catalog
compare, compare_only	Other Venues' Names, Other Venues' Cust. Satisf., Other Venues' Price Range	String Catalog Catalog

Table 1: Action types and slots.

We conduct a data collection study with 90 participants where each participant is presented with 45-50 dialog acts of different action types. The participants are asked to express a given dialog act in a single sentence and to use all slots given in the act. Moreover, they are told to not rely on their commonsense knowledge or use any information that might be inferred from the given ones. In the study, greeting and goodbye actions are not used. Each dialog act contains two to four randomly chosen slots in addition to the name of the venue in focus. It is guaranteed that a participant receives different sets of slots for the same action type even if the number of slots are the same. We use both real and artificial data in order to fill in slot values. Information about a small set of dining venues is obtained from an online restaurant search service and that information is augmented with artificial information in order to expand the collection. For instance, new dialog acts are produced by adding new neighbour restaurants to existing dialog acts without any neighbourhood information. Each dialog act is presented to four different participants. At the end, 4200 dialog act and reference sentence pairs are collected. Figure 3 shows two dialog acts with three reference sentences from our collection.

(type='inform', name='Lezzet Mekanı', customer_satisfaction='Yüksek', cuisines='Tatlı, Dünya Mutfağı Yemekleri', price_range='Pahalı', region='Caddebostan, İstanbul')

i) Lezzet Mekanı, İstanbul Caddebostan'da, tatlı ve dünya mutfağı yemekleri servis eden pahalı fakat lezzetli yemekleriyle müşteri memnuniyetini üst seviyede tutan bir mekandır.
(Lezzet Mekanı is a place in Caddebostan, Istanbul that serves sweet and world cuisine and keeps customer satisfaction at the highest level with its expensive but delicious dishes.)

ii) Dünya mutfağına ait yemekler ve tatlılar bulabileceğiniz, müşteri memnuniyeti konusunda çok başarılı olması rağmen fiyatları pahalı olan Lezzet Mekanı, İstanbul Caddebostan'da bulunmaktadır.
(Lezzet Mekanı, where you can find desserts and dishes from the world cuisine, is very successful in customer satisfaction though it is expensive, and is located in Caddebostan, Istanbul.)

iii) İstanbul Caddebostan'da tatlılar ile dünya mutfağına ait yemekler yenebilecek Lezzet Mekanı, pahalı fiyata yemekler sunan ve müşterilerin çok memnun olduğu bir restorandır.
(Lezzet Mekanı in Istanbul Caddebostan, where you can eat desserts and dishes from the world cuisine, is a restaurant that offers expensive dishes and where customers are very satisfied.)

(type = 'compare', name = 'Cafe Botanica', price_range = 'Ortalama', other_venues_names = 'Mayday Cafe Bar, Mevlana Lokantası, Cafe de Kedi', other_venues_price_range = 'Ucuz')

i) Cafe Botanica; ucuz fiyatlı Mayday Cafe Bar, Mevlana Lokantası, Cafe de Kedi'ye kıyasla ortalama fiyatlı bir mekandır.
(Cafe Botanica is an average-priced venue compared to the cheaply priced Mayday Cafe Bar, Mevlana Lokantası and Cafe de Kedi.)

ii) Cafe Botanica ortalama fiyatlardayken Mayday Cafe Bar, Mevlana Lokantası ve Cafe de Kedi ucuz mekanlardır
(While Cafe Botanica is at average prices, Mayday Cafe Bar, Mevlana Restaurant and Cafe are cheap venues.)

iii) Ortalama fiyatlarıyla bilinen Cafe Botanica, Mayday Cafe Bar, Mevlana Lokantası ve Cafe de Kedi gibi mekanların ucuz menülerine kıyasla pahalı kalmaktadır
(Cafe Botanica which is known with its average prices is expensive compared to the venues with cheap menus Mayday Cafe Bar, Mevlana Lokantası and Cafe de Kedi.)

Figure 3: Examples of dialog acts and reference sentences.

4.2 Input Representation

A dialog act is represented as a sequence of field value pairs (e.g., field$_1$= value$_1$) where the first pair corresponds to the action type and the rest are slot value pairs. The value of a field might contain a single word or a sequence of words. The field name (f_x) and its position in the value sequence (p_x) are used to represent each word (w_x). To represent the position of a word in a sequence, its position from the beginning of the sequence (p_x+) and from the end of the sequence (p_x-) are used. Therefore, a word that appears in a field value is represented as $R_x = (f_x , p_x+ , p_x-)$. All punctuation characters in field values are represented similarly. Table 2 shows the representations of all words in the dialog act *(type = 'inform_only', name = 'Denizaltı Restaurant', cuisine = 'Kafeterya Ürünleri, Türk Yemekleri', region = 'Urla, İzmir', near = 'VVapiano').* In this example, the value of the name field consists of two words, namely Denizaltı and Restaurant. The word Denizaltı is the first word starting from the beginning of value sequence and the second word from the end of the sequence. Therefore, its representation is *(name,1,2).*

Each word in a field value (w_x) and its representation (R_x) are encoded into four embeddings and then concatenated to form the final input embedding of the encoder ($i_e = w_e \oplus f_e \oplus p_e+ \oplus p_e-$). A reference sentence already has a sequence of word tokens and thus each token is encoded into a word embedding only:

- Word embedding: Vector representation of the word (w_e)
- Field embedding: Vector representation of the field name (f_e)
- Beginning position embedding: Vector representation of the position from the beginning of the field value (p_e+)
- End position embedding: Vector representation of the position from the end of the field value (p_e-)

Field	Value	Word	Represent.
type	inform_only	inform_only	(type,1,1)
name	Denizaltı Restaurant	Denizaltı	(name,1,2)
		Restaurant	(name,2,1)
cuisine	Kafeterya Ürünleri, Türk Yemekleri	Kafeterya	(cuisine,1,5)
		Ürünleri	(cuisine,2,4)
		,	(cuisine,3,3)
		Türk	(cuisine,4,2)
		Yemekleri	(cuisine,5,1)
region	Urla, İzmir	Urla	(region,1,3)
		,	(region,2,2)
		İzmir	(region,3,1)
near	VVapiano	VVapiano	(near,1,1)

Table 2: Word representations.

4.3 Sequence-to-Sequence Generation Model

To capture temporal processing and feedback requirements of sequences in learning, we approach to the generation problem using a recurrent neural network (RNN) based solution. RNN models are of great utility in computing current output with respect to previous computations kept in hidden states and their processing power makes them widely applicable to speech recognition (Hsu et al., 2016; Prabhavalkar et al., 2017) and language processing studies (Socher et al., 2011; Daza and Frank, 2018). In our work, dialog acts and reference sentences are sequences of

variable-length. Thus, we formulate our generation task as sequence-to-sequence (Seq2Seq) learning (Sutskever et al., 2014), a type of an RNN with encoder-decoder. Our model uses a long short-term memory (LSTM) based RNN to encode the input sequence into hidden states. A second LSTM-based RNN is used to decode hidden states and generate the output sequence. Given that x_t and h_t are the input and hidden state at time step t; i, f, and o are input, forget and output gates; and C and $\tilde{C}$ are cell and candidate cell states, the computations used with LSTM units are as follows:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_i.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \\
\mathbf{o}_t &= \sigma(\mathbf{W}_0.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_0) \\
\tilde{\mathbf{C}}_t &= \tanh(\mathbf{W}_C.[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C) \\
\mathbf{C}_t &= \mathbf{f}_t * \mathbf{C}_{t-1} + \mathbf{i}_t * \tilde{\mathbf{C}}_t \\
\mathbf{h}_t &= \mathbf{o}_t * \tanh(\mathbf{C}_t)
\end{aligned} \tag{1}$$

5 Evaluation

Neural models often suffer from rare words while generating text from data since their verbalization cannot be predicted properly. Delexicalization is one of the mostly studied solutions to this issue where words are replaced with placeholders in data before being used for training. Texts produced by the generation model are then processed to replace these placeholders with actual words that appear in the original data. For this study, we delexicalize our input collection (-Del) and obtain a second version of our dataset (+Del). We only replace content words of slots with verbatim strings (e.g., name and region in Table 1) and leave those with categorical values (e.g., cuisine and price range) untouched. We have different dialog acts that differ only in slot values that are not replaced during delexicalization. Therefore, these dialog acts are counted as different acts in the second dataset. The number of placeholders in our delexicalized dataset corresponds to 17.71% of all words in reference sentences. Table 3 presents token-based statistics for both datasets.

We train two models on both original and delexicalized datasets. The first model is the sequence-to-sequence model described in Section 4.3 (Model_Att-) and the same model augmented with an attention mechanism (Model_Att+). We perform experiments to finetune model parameters by optimizing BLEU score on the development set. The models reported here use a single hidden layer and 700 LSTM units in encoder and decoder. Word embeddings of length 400, field embedding of length 50, and position embedding of length 5 are used. The epoch number is set to 10 and Adam optimizer with a learning rate of 0.003 is utilized. We compare our models with a prior Seq2Seq generation model (Liu et al., 2017) (Model_SA) whose primary focus is to generate one sentence biographies from Wikipedia infoboxes where the structure and content of infobox tables are modeled separately. In addition to learning what to convey in the output, the model also learns how to order the selected content. To train this structure-aware generation model with dual attention, we process all dialog acts in our dataset as infobox tables where the action type is considered as infobox table type and remaining slot value pairs as field value pairs of infobox tables. The same model parameters are used in learning.

Property	**Input Data**	**Delexicalized Data**
Input Dictionary Size	2966	1247
Output Dictionary Size	2827	1177
Avg. DA Length	8.23	5.85
Avg. Ref. Text Length	15.13	11.96

Table 3: Properties of input datasets.

Act Type	**Training**	**Validation**	**Test**
inform	1690	220	200
inform_only	448	57	45
inform_not	662	81	93
inform_all	109	14	20
request	217	24	34
compare	120	11	12
compare_only	114	13	16

Table 4: Distribution of action types in datasets.

Our input collection of dialog act and reference sentence pairs is splitted into training set of 3360, validation set of 420, and test set of 420 pairs. Table 4 presents the distribution of action types in these sets. In our experiments, we evaluate the efficiency of models in producing utterances from dialog acts and leave an evaluation of fluency and naturalness of these productions to future work. Here, we report performances using three evaluation metrics, BLEU (Papineni et al., 2002), ROUGE-n and ROUGE-L fmeasures (Lin, 2004), and Slot error rate (SER) (Riou et al., 2019). The slot error rate is computed as (M+R)/N where M and R correspond to the number of missing and redundant slots in the generated utterance, and N is the total number of

		BLEU	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4	ROUGE-L	SER%
Model_Att-	*-Del*	0.017	0.161	0.033	0.010	0.002	0.129	70.2
	+Del	0.056	0.373	0.142	0.065	0.030	0.280	57.5
Model_SA	*-Del*	0.125	0.417	0.216	0.122	0.063	0.337	53.5
	+Del	0.323	0.849	0.632	0.459	0.328	0.543	35.6
Model_Att+	*-Del*	0.144	0.677	0.490	0.335	0.220	0.439	40.6
	+Del	0.302	0.857	0.634	0.452	0.314	0.524	35.5

Table 5: Performance scores of different models.

slots in the corresponding dialogue act.

For each model, we perform 5 runs with different random initializations on both datasets. Table 5 presents computed average scores. The model without attention (Model_Att-), not surprisingly, fails to learn the generation effectively and receives the lowest performance scores in all metrics. In addition, repetitive slot values and very similar sentence productions for different dialog acts are highly observed in the productions. On the other hand, we observe that our model with attention (Model_Att+) achieves highest BLEU and ROUGE scores on the original dataset (-Del). However, our model is behind the structure-aware model (Model_SA) on the delexicalized dataset (+Del) with respect to the BLEU score and over high order n-grams (ROUGE-3 and ROUGE-4). This less significant difference might be attributed to the fact that structure-aware model performs better in producing longer matching sequences than our model, which is also validated by ROUGE-L scores. Both models exhibit large performance improvements on the delexicalized dataset where BLEU scores are more than doubled. The measured positive impact of delexicalization on structure-aware model is more than what we observe with our model. The contribution of delexicalized dataset to model Model_SA is mainly observed on longer word sequences (e.g., from 0.063 to 0.328 in ROUGE-3).

Although BLEU and ROUGE evaluations validate word-based performances of these models, they do not provide any insights into the content quality, particularly the accuracy of selected content and the slot coverage of these models. On both datasets, our model with attention achieves the best slot error rates where delexicalization improves the performance by approximately 5%. The structure-aware model performs similarly only on delexicalized dataset, but the achieved improvement is more substantial than that seen in our model. These results demonstrate that both models need further improvements to better cover slot values resulting in fewer repeated or omitted information in produced utterances.

There are two major drawbacks of our model. First, it is learning from a corpus which is relatively small in comparison with many available datasets compiled for other languages. Second, it suffers from semantically similar entities in the dataset (e.g., cuisine or region) and entities that appear more frequently than others in the training data are selected by the model regardless of what is provided in the dialogue act. We argue that with a larger training corpus and more effective attention mechanism, our generation performance would be improved in the future.

6 Conclusion

This work presents our efforts towards developing a Turkish task-oriented dialogue system for venue recommendation and reservation. The current system is implemented using a pipeline approach, and natural language understanding and dialogue management components are built using the RASA open-source framework. In order to generate utterances from dialogue act representations, we develop a sequence-to-sequence neural model with attention. The model is trained with a small-sized Turkish corpus consisting of pairs of dialogue acts and reference sentences. To the best of our knowledge, this work is the first that investigates the use of Turkish neural generation in dialogue systems and measures the effectiveness of conversational generation from structured input on a morphologically rich language. In the future, we plan to collect a larger corpus and improve the performance of our generator. Moreover, enhancing the dialogue capabilities of our overall system and qualitatively evaluating the performance of the generation model are some of our future plans.

Acknowledgments

This work is supported by TUBITAK-ARDEB under the grant number 117E977.

References

Burcu Karagol Ayan. 2000. Morphosyntactic generation of turkish from predicate-argument structure. In *Proceedings of the COLING Student Session*.

Tom Bocklisch, Joey Faulkner, Nick Pawlowski, and Alan Nichol. 2017. Rasa: Open source language understanding and dialogue management. *ArXiv*, abs/1712.05181.

Çagdas Can Birant, Özgün Kosaner, and Özlem Aktas. 2016. A survey to text summarization methods for turkish. *International Journal of Computer Applications*, 144:23–28.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *Special Interest Group on Knowledge Discovery in Data Explorations Newsletter*, 19(2):25–35.

Ilyas Cicekli and Turgay Korkmaz. 1998. Generation of simple turkish sentences with systemic-functional grammar. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, page 165–173, USA. Association for Computational Linguistics.

Heriberto Cuayáhuitl. 2017. *SimpleDS: A Simple Deep Reinforcement Learning Dialogue System*. Springer.

Angel Daza and Anette Frank. 2018. A sequence-to-sequence model for semantic role labeling. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 207–216, Melbourne, Australia. Association for Computational Linguistics.

li Deng, Gokhan Tur, Xiaodong He, and Dilek Hakkani-Tur. 2012. Use of kernel deep convex networks and end-to-end learning for spoken language understanding. In *2012 IEEE Workshop on Spoken Language Technology, SLT 2012 - Proceedings*, pages 210–215.

Alexandre Denis, Matthieu Quignard, and Guillaume Pitel. 2006. A deep-parsing approach to natural language understanding in dialogue system: Results of a corpus-based evaluation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*.

Anoop Deoras and Ruhi Sarikaya. 2013. Deep belief network based semantic taggers for spoken language understanding. In *INTERSPEECH*.

Ondřej Dušek and Filip Jurčíček. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–51.

M. Elifoğlu and T. Güngör. 2018. A restaurant recommendation system for turkish based on user conversations. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

Michael English and Peter Heeman. 2005. Learning mixed initiative dialog strategies by using reinforcement learning on both conversants. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 1011–1018.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Tagyoung Chung, and Dilek Hakkani-Tur. 2019. Dialog state tracking: A neural reading comprehension approach. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 264–273.

Arshit Gupta, Peng Zhang, Garima Lalwani, and Mona Diab. 2019. CASA-NLU: Context-aware self-attentive natural language understanding for task-oriented chatbots. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1285–1290.

Matthew Henderson, Blaise Thomson, and Steve J. Young. 2014. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. *2014 IEEE Spoken Language Technology Workshop (SLT)*, pages 360–365.

Matthew Henderson, Ivan Vulic, Inigo Casanueva, Paweł Budzianowski, Daniela Gerz, Sam Coope, Georgios Spithourakis, Tsung Hsien Wen, Nikola Mrksic, and Pei-Hao Su. 2019. Polyresponse: A rank-based approach to task-oriented dialogue with application in restaurant search and booking. In *Proceedings of the 2019 EMNLP and the 9th IJCNLP*, pages 181–186.

Wei-Ning Hsu, Yu Zhang, and James R. Glass. 2016. A prioritized grid long short-term memory rnn for speech recognition. *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 467–473.

Aaron Jaech, Larry Heck, and Mari Ostendorf. 2016. Domain adaptation of recurrent neural networks for natural language understanding. *arXiv preprint arXiv:1604.00117*.

Filip Jurčíček, Ondřej Dušek, Ondřej Plátek, and Lukáš Žilka. 2014. Alex: A statistical dialogue systems framework. In *Text, Speech and Dialogue*, pages 587–594. Springer International Publishing.

Sungjin Lee and Amanda Stent. 2016. Task lineages: Dialog state tracking for flexible interaction. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 11–21.

Xiujun Li, Sarah Panda, Jingjing Liu, and Jianfeng Gao. 2018. Microsoft dialogue challenge: Building end-to-end task-completion dialogue systems. *arXiv preprint arXiv:1807.11125*.

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

Bing Liu and Ian Lane. 2018. End-to-end learning of task-oriented dialogs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 67–73.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2017. Table-to-text generation by structure-aware seq2seq learning. In *CoRR*.

François Mairesse, Milica Gašić, Filip Jurčíček, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. 2010. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561.

François Mairesse and Marilyn A. Walker. 2011. Controlling user perceptions of linguistic style: Trainable generation of personality traits. *Computational Linguistics*, 37(3):455–488.

Margaret Mitchell, Dan Bohus, and Ece Kamar. 2014. Crowdsourcing language generation templates for dialogue systems. In *Proceedings of the INLG and SIGDIAL 2014 Joint Session*, pages 172–180.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 794–799.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proc. of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206.

Muhammed Yavuz Nuzumlalı and Arzucan Özgür. 2014. Analyzing stemming approaches for Turkish multi-document summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 702–706.

Kemal Oflazer and Murat Saraclar. 2018. *Turkish Natural Language Processing*, 1st. edition. Springer.

Alice H. Oh and Alexander I. Rudnicky. 2000. Stochastic language generation for spoken dialogue systems. In *Proceedings of the 2000 ANLP/NAACL Workshop on Conversational Systems - Volume 3*, page 27–32, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog.

Rohit Prabhavalkar, Kanishka Rao, Tara N. Sainath, Bo Li, Leif Johnson, and Navdeep Jaitly. 2017. A comparison of sequence-to-sequence models for speech recognition. In *Proceedings of the 18th International Speech Communication Association (Interspeech)*.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let's go public! taking a spoken dialog system to the real world. In *Proceedings of Interspeech 2005*.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Proceedings of the Eighth Annual Conference of the International Speech Communication Association*, pages 1605–1608.

Hang Ren, Weiqun Xu, Yan Zhang, and Yonghong Yan. 2013. Dialog state tracking using conditional random fields. In *Proceedings of the SIGDIAL 2013 Conference*, pages 457–461.

Matthieu Riou, Bassam Jabaian, Stéphane Huet, and Fabrice Lefèvre. 2019. Reinforcement adaptation of an attention-based neural natural language generator for spoken dialogue systems. *Dialogue & Discourse*, 10:1–19.

R. Sarikaya, P. A. Crook, A. Marin, M. Jeong, J. P. Robichaud, A. Celikyilmaz, Y. B. Kim, A. Rochette, O. Z. Khan, X. Liu, D. Boies, T. Anastasakos, Z. Feizollahi, N. Ramesh, H. Suzuki, R. Holenstein, E. Krawczyk, and V. Radostev. 2016. An overview of end-to-end language understanding and dialog management for personal digital assistants. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 391–397.

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Proc. of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 5414–5421.

Pararth Shah, Dilek Hakkani-Tur, and Larry Heck. 2016. Interactive reinforcement learning for task-oriented dialogue management. In *Workshop on Deep Learning for Action and Interaction (NIPS)*.

Richard Socher, Cliff Chiung-Yu Lin, Andrew Y. Ng, and Christopher D. Manning. 2011. Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the 28th International Conference on International Conference on*

Machine Learning, page 129–136, Madison, WI, USA. Omnipress.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112, Cambridge, MA, USA. MIT Press.

G. Tur, L. Deng, D. Hakkani-Tür, and X. He. 2012. Towards deeper understanding: Deep convex networks for semantic utterance classification. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5045–5048.

Zhuoran Wang and Oliver Lemon. 2013. A simple and generic belief tracking mechanism for the dialog state tracking challenge: On the believability of observed information. In *Proceedings of the SIGDIAL 2013 Conference*, pages 423–432.

Gellert Weisz, Pawel Budzianowski, Pei-Hao Su, and Milica Gasic. 2018. Sample efficient deep reinforcement learning for dialogue systems with large action spaces. *IEEE/ACM Transactions Audio, Speech and Language Processing*, 26(11):2083–2097.

Tsung-Hsien Wen, Milica Gašić, Dongho Kim, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015a. Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 275–284.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015b. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve J. Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 438–449. Association for Computational Linguistics.

Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020. Tod-bert: Pre-trained natural language understanding for task-oriented dialogues.

Zhao Yan, Nan Duan, Peng Chen, Ming Zhou, Jianshe Zhou, and Zhoujun Li. 2017. Building task-oriented dialogue systems for online shopping. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4618–4625. AAAI Press.

Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. 2013. Recurrent neural networks for language understanding. In *Proceedings of Interspeech*, pages 2524–2528.

Tianyu Zhao and Tatsuya Kawahara. 2019. Joint dialog act segmentation and recognition in human conversations using attention to dialog context. *Computer Speech & Language*, 57:108 – 127.

Yin Jiang Zhao, Yan Ling Li, and Min Lin. 2019. A review of the research on dialogue management of task-oriented systems. *Journal of Physics: Conference Series*, 1267:012025.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2019. Multi-task learning for natural language generation in task-oriented dialogue. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1261–1266.

How are you? Introducing stress-based text tailoring

Simone Balloccu, Ehud Reiter
University Of Aberdeen
United Kingdom
simone.balloccu@abdn.ac.uk
e.reiter@abdn.ac.uk

Alexandra Johnstone, Claire Fyfe
Rowett Institute, University of Aberdeen
United Kingdom
alex.johnstone@abdn.ac.uk
c.fyfe@abdn.ac.uk

Abstract

Can stress affect not only your life but also how you read and interpret a text? Healthcare has shown evidence of such dynamics and in this short paper we discuss customising texts based on user stress level, as it could represent a critical factor when it comes to user engagement and behavioural change. We first show a real-world example in which user behaviour is influenced by stress, then, after discussing which tools can be employed to assess and measure it, we propose an initial method for tailoring the document by exploiting complexity reduction and affect enforcement. The result is a short and encouraging text which requires less commitment to be read and understood. We believe this work in progress can raise some interesting questions on a topic that is often overlooked in NLG.

1 Introduction

Healthcare can benefit greatly from NLG as communication plays a big role in therapy outcome (Stewart, 1995) and electronic documentation is useful (Ross et al., 2016) but heavy and time-consuming to be produced (Arndt et al., 2017). NLG can speed up the process and there is evidence of increasing adoption of data-to-text in healthcare (Pauws et al., 2019). Despite NLG being valid for producing standardised documentation, the situation is more complex when the targeted reader is a patient: people are all different and a single text is often not enough. Healthcare-NLG often addresses the problem by adopting user tailoring, which is the practice of generating a text customised to the user traits and preferences (Kukafka, 2005). In this work, we focus on user stress, a factor which is often not considered in NLG, as a feature for text-tailoring. We show that stress does matter when reading text and provide a real-world example (the dietary domain) along with some initial discussion and examples on how stress-based tailoring could be carried out. Motivated by the PhilHumans project[1] and its goal of improving the interactions between personal health apps and users, we explore the benefits of personalising generated texts based on how much stressed the user is. Stress impacts people's health, behaviour, receptivity to advice, and the ability to absorb complex information, so basing tailoring on this element could help health software support patients better. The structure of the paper is the following: in Section 2 we recap the state of user tailoring in NLG, emphasizing that no known system does properly address user stress; in Section 3-3.1 we provide a real-world example in the dietary domain, showing the results of statistical analysis that, in line with previous healthcare studies, hints at the correlation between stress and eating behaviour; Section 4.1 discusses possible methodologies to assess user stress, prioritising the non-intrusive ones; in Section 4.2 we propose two theoretical ways to tailor the text for a stressed user, along with an example; Section 5 closes the paper with our final considerations.

2 Related Work

Natural Language Generation has been used in the healthcare domain in a variety of ways, with different aims. Since the spread of e-health, various works have focused on automating the creation of documents. some examples of these are clinical encounters (Finley et al., 2018), chief complaints (Lee, 2018) and handover reports (Schneider et al., 2013). There has also been some research on producing multiple texts from the same source, as in the BabyTalk project (Portet et al., 2008) which generated premature infants reports for doctors(Portet et al., 2009), nurses (Hunter et al., 2011) and parents (Mahamood and Reiter, 2011) in Neonatal Intensive Care Unit (NICU) environ-

[1] PhilHumans Project page: https://www.philhumans.eu/

ment. Another scenario in which NLG contributed to healthcare is decision support (Binsted et al., 1995) (Hommes et al., 2019), where the text can help patients understand their clinical situation, and prepare them for shared decision making (Seminar, 2011) (Stiggelbout et al., 2012). The majority of these systems don't need any personal information about the reader because their final purpose is purely informative. It is, however, different when we move to behavioural change, as the text aims to trigger a lifestyle improvement in the reader. In this case we're dealing with persuasive communication, which is the act of communicating to make the reader perform certain actions or collaborate in various activities (Guerini et al., 2011). Persuasive communication systems can be distinguished in vertical and horizontal approaches (Maimone et al., 2018): vertical systems tends to be specialised, hence they are very effective but harder to move between different domains, while horizontal systems are often conceptual and focused on domain-independent knowledge and strategies. When persuasive communication has to be obtained through NLG, a popular approach is to profile the user to model the text accordingly. This practice is known as text tailoring (Kukafka, 2005) and has shown promising results (Kreuter and Wray, 2003). Morevoer NLG tailoring is particularly helpful, since manual text customisation is often time-consuming (Pauws et al., 2019). NLG-driven tailoring was tested for lifestyle improvements and behavioural change in various domains. Vertical approaches includes solutions for smoking cessation (Reiter et al., 2003), driving conduct improvement (Braun et al., 2018), diet management (Anselma et al., 2018)(Anselma and Mazzei, 2018)(Anselma and Mazzei, 2017) and therapy recommendation (Skinner et al., 1994). Horizontal approaches deals with more general topics, like the key challenges and model design for behaviour change (Oinas-Kukkonen, 2013)(Kelders et al., 2016)(Oinas-Kukkonen and Harjumaa, 2009), argumentative persuasive communication (Zukerman et al., 2000) (Reed et al., 1996) or the inclusion of generic psychological traits and basic human values into the model (Ding and Pan, 2016). Finally, there have been some attempts to partially merge horizontal and vertical models (Maimone et al., 2018)(Dragoni et al., 2018)(Donadello et al., 2019). When the text must be built according to user traits, it is important to identify the variables which could influence not only the behaviour but also user engagement. These include general psychological tailoring techiques (Hawkins et al., 2008); user numeracy, literacy (Williams and Reiter, 2008) and domain knowledge (McKeown et al., 1993)(Paris, 1988); affect (Mahamood and Reiter, 2011). User stress is a factor which, to our knowledge, is largely ignored in the existing literature. Some NLG works have considered stress, but not in a dynamic way. Some works in tactical NLG (text which tries to induce a certain emotional status) (Van Der Sluis and Mellish, 2008) considered stress, but only to assess the text effect. This makes sense since tactical NLG aims to induce an emotional status instead of detecting or managing it. BabyTalk-Family project (Mahamood and Reiter, 2011) tried to actively tailor the document by estimating stress value, but without detecting it from the user itself. Again, the choice is reasonable since having children in NICU can be a traumatic experience, therefore directly asking questions about stress could be intrusive and dangerous. Beside these boundary cases we argue that stress-based tailoring is of primary importance as stress has been shown to significantly impact individual reading skills (Rai et al., 2015)(Peng et al., 2018), potentially causing temporary difficulty in understanding the given text. Therefore stress data can be exploited to guide the tailoring process with precious hints on how to re-realise the text. This motivates researching how user stress assessment can be done in the least intrusive way.

3 A real world example: NeuroFAST

Before going into the details of how stress-based tailoring could be achieved, we want to introduce some preliminary evidence that stress does matter and can influence user behaviour. We exploit a dataset which is derived from the EU-funded project NeuroFAST[2], regarding the socio-psychological forces that could influence eating behaviour. Our dataset (Malone et al., 2015) contains a detailed food diary for 413 different workers, together with a "Daily Hassles" questionnaire in which the participants noted stressful events. Each entry is scored in a range between 0 and 4, with a higher score indicating a higher stress level. For example, we can find an individual that put "Caught in a traffic jam on the way to work" as a daily hassle. This resulted in a score of 2/4. We analysed the

[2]NeuroFAST project page: https://cordis.europa.eu/project/id/245009

dataset to test the following hypothesis: does stress influence workers' diet? This is not a novelty in healthcare, as different studies linked stress to eating disorders (Scott and Johnstone, 2012)(Torres and Nowson, 2007)(Puddephatt et al., 2020)(Riffer et al., 2019) but we still provide an additional, post hoc, analysis on said data to show a concrete example. Additionally, even if the original study was not meant to inspect such correlation, both data were readily available, making it possible to inspect it.

3.1 NeuroFAST stress analysis

To test our hypotheses we extracted the data, isolating the absolute calorie change for every couple of days: this was necessary since the dataset itself doesn't provide any kind of dietary goal (such as the daily calorie target) and hence we could only calculate how each participant energy intake changed during the week. This gave use 1410 data points, each one represented as a pair in which the first element is the absolute energy shift (how much did the intake increase/decrease compared to the previous day?) and the second one is the daily stress score. Overall, the dataset contained only one week worth of data for each worker, so we couldn't further inspect their eating behaviour. We also had to cut out some participants cause they didn't fill in the "Daily Hassles" form. Now we proceed to show the results of our statistical analysis, to test whether stress does affect individual diet or not. It must be considered that a higher stress score doesn't necessarily imply overeating: people react differently to stress (some will eat more, other will eat less) and that's another reason why we calculated the absolute calorie change. For the first analysis, we looked for a correlation between the calorie shift and some stress trends which can occur in workers' life. We identified four stress patterns by comparing the score for every given pair of days:

- **Stress variation:** the score changes (increases or decreases) while staying positive
- **Stress drop:** the score drops to zero from any given value
- **Stress stasis:** the score doesn't change but stays positive
- **Stress absence:** the score stays equals to zero during the two days

Table 1: Stress trend analysis result (ONE-WAY ANOVA)

Trend	p-value
variation - absence	0.520
drop - absence	0.886
stasis - absence	**0.0317**
drop - variation	0.326
stasis - variation	0.722
stasis - drop	0.052

By defining these patterns we were able to check if a particular stress trend could be related to a calorie shift. We ran a one-way ANOVA which showed statistically significant difference in the distributions ($p \approx$ **0.008**; F-value = **3.9**; Df = **3**; $\eta^2 \approx$ **0.008**). An additional Tukey Honestly Significant Difference (HSD) test revealed that the only statistically significant difference was between the Stress Absence and Stress Stasis trends (-tbfp $\approx$ 0.031). In other words, there was a significant difference in how people ate when they were consistently stressed during the weeks compared to when they weren't stressed. The analysis results are summarized in Table 1.

For the second analysis we considered a simpler scenario: we aggregated every stress trend except for the absence, to inspect a link between being generally stressed in any possible way or calm and the calorie shift. In this regards we ran a Welch Two-Sample T-Test which gave us a significant result ($p \approx$ **0.046**). We want to remind that the analysis we ran is a post-hoc one: original experiment wasn't mean to test our hypotheses. Hence our results can only be taken as an hint that stress could have influenced participants' eating habits.

4 Stress-based tailoring

We now proceed to detail a hypothetical way to tailor a text-based on user stress. To do so we do stay in the dietary domain and consider producing a diet-coaching report by using NLG. To do so, we take into consideration our previously proposed architecture for dynamic tailoring in healthcare (Balloccu et al., 2020), which we summarise in Figure 1. The framework is designed to extract data regarding user diet, then produce a text which is tailored by taking into account both general tailoring techniques and user individual traits. What we do in here is to formulate a theoretical way to

enrich the tailoring logic with details about the user stress which will guide the size and content of the final text.

4.1 Assessing user's stress

As a first step, we must determine whether the user is stressed and find a measure for this data: this is a major challenge, as it should be done in the most reliable and least intrusive possible way. Detecting stress is usually pursued through two different tools: body sensors and self-assessment tools(Plarre et al., 2011). Intrusion-wise, sensors are the least desirable approach: since our target is to deliver a report with a high frequency, it is unrealistic to expect the user to wear detection devices daily. Even if the high-frequency constraint is omitted, sensors tend to be reliable in controlled environments only (Healey and Picard, 2005)(Madan et al., 2011)(Plarre et al., 2011), as they tend to struggle to distinguish stressful events from normal activities which alter user's psychophysical condition (like sports activities). On the other hand, self-assessment tools typically involve questionnaires like DASS-21 (Lovibond and Lovibond, 1995) or PANAS(Watson et al., 1988). Such tools can produce an estimation of users' stress related to a relatively short period (a week for example). Moreover, it could be even possible to assess daily stress: this would make the system capable of tailoring the text "on the fly", potentially fine-tuning the text details every single day. A possible way to address daily stress is, for example, the "Daily Hassles" form that was used in NeuroFAST project. However, measuring stress every day reduces self-assessment tools reliability, and reduces their validity to some specific scenarios (Adams et al., 2014) only. Finally, it must be considered that forcing the user to fill a form every single day could end up being a stressful operation as well. From a discussion with experts from NeuroFAST project, it turned out that the stress values were collected by using a mix of sensors and forms, an experience which the medical staff itself described as potentially stressful for the participants, especially when they were shift workers.

4.2 Enriching user tailoring with stress

Given what has been said until now, we proceed to formulate two key parameters on top of which the tailoring will take place:

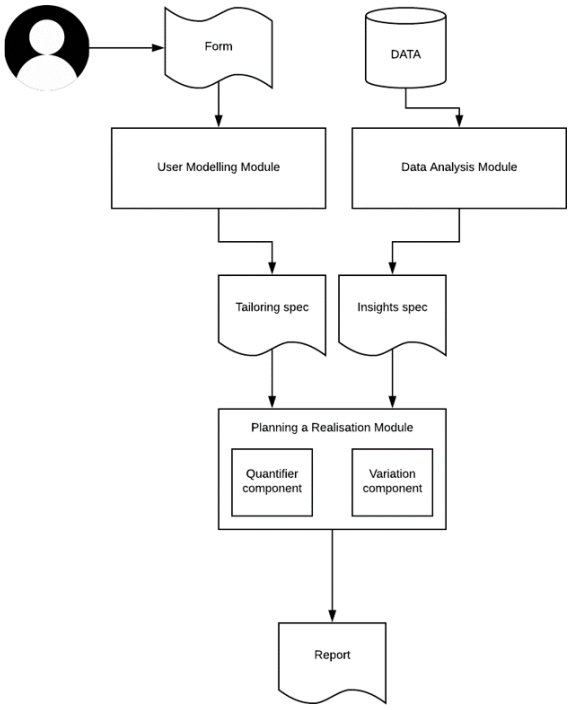


Figure 1: Original framework architecture

- **Complexity:** this involves both the length and terminology of the given text. There have been proofs that stress does impact on working memory (Rai et al., 2015)(Peng et al., 2018). This means that being stressed potentially influences reading skills. Addressing this scenario can be done by delivering a shorter text and simplifying complex terminology, in line with previous work (McKeown et al., 1993)(Paris, 1988).

- **Affect:** when dealing with stressful events, like work or personal problems, (Scott and Johnstone, 2012)(Torres and Nowson, 2007)(Puddephatt et al., 2020)(Riffer et al., 2019) users could feel demoralised after reading a report which states they didn't meet their current goals. This applies well to the dietary domain but it might reasonably make sense in other areas. A possible way to address this problem is adopting an affective tone in the text, in line with previous studies (Mahamood and Reiter, 2011)(De Rosis and Grasso, 1999). Omitting certain information could be a valid option as well under some circumstances. For example (Van Deemter and Reiter, 2018), showed how "lying" could be useful in the NICU environment, since pre-

venting the child grandparents from knowing about negative development could have avoided potential health issues (i.e. heart attacks).

Regarding complexity, good tailoring should address both reducing text length and simplifying the content itself. This can be split into two complementary document planning (length) and microplanning (terminology) steps. This is especially important given that, even without considering stress, patients typically struggle to understand complex medical language (Elhadad, 2006)(Zeng-Treitler et al., 2008). We leave terminology simplification as future work and focus, for now, on obtaining a shorter text. We do consider an output example from our system (Figure 2), which informs the user with some weekly insights about his/her diet. The text itself is lengthy, hence we try to shorten it to get an immediately understandable content. To do so we remove superfluous information which doesn't contribute to giving the user facts about their diet. A hypothetical result is showed in Figure 3. The text is now around a quarter of its original length but delivers the same information (i.e.: calories and nutrients). Potentially traumatic information was removed, like adverse effects, and some of the tailoring techniques got lost during the shortening phase: these include both psychological practices (Hawkins et al., 2008) and user choices like adverse effects. We argue that, by sacrificing a bit of customisation, we're gaining a lot of readability which is a critical parameter when the user is stressed. Moreover, the majority of user choices are still being kept. It is, however, fair to underline that the system should offer the user the chance to see the original uncut text, to access the omitted details. It is now necessary to re-introduce the affective bits inside the text as stated before: despite the new text being much faster to be read in terms of length, it sounds quite cold compared to the original one. Therefore we re-insert some motivational frames without being redundant and with the primary target of keeping the text short and simple. A mockup with an affect component can be seen in Figure 4: the improved strategy is pretty much the same we originally adopted (Balloccu et al., 2020) and focuses on lowering the weight of bad news and encouraging the user to pursue their goals (which were re-introduced into the text). Overall the text is still significantly shorter than the original one.

Hi Paul, this is your weekly diet report. You told us that you want to lose weight and gain self-confidence. So we wrote this to help you out.

You did a great job on calories; you had some problems with sugar and sodium.

Going a bit deeper, Monday was your best day! Your calories were about perfect then. Friday gave you some problems, as you ate about a third more than what you should.

Your sugar consumption is around twice more than what it should be, and much of it came from Coca Cola. We know it's hard to change what you eat, but sugar excess is bad for your health and can cause pale skin, anxiety and fatigue. Try having less sugary foods as it would turn in weight loss, less dental plaque and more energy.

Your sodium consumption is around half more than what it should be. A lot of it came from Pringles chips. Keep in mind that sodium excess can lead to nausea, headache and seizures. Less salty foods mean memory improvement, less bloating and lower blood pressure.

Paul, we came up with some suggestions based on your needs. Regarding sugar, you could replace sugary drinks with tea or coffee. Also, sodium will be lower if you avoid salty snacks and opt for fruits instead.

Figure 2: Starting text

Hi Paul, calories were good this week, especially Monday, but you had problems on Friday. Nutrient-wise, you had around twice more than your sugar target (cut a bit on Coca-Cola) and around a half more sodium than what you should get (try reducing chips).

Figure 3: Reduced example

Hi Paul, this week you did a great job on calories, especially Monday. You had some problem Friday: we're sure that next week will be better! Nutrient-wise, you had around twice more than your sugar target (cut a bit on Coca-Cola) and around a half more sodium than what you should get (try reducing chips). Changing what you eat is a long path, keep up will succeed in losing weight and gaining self-confidence!

Figure 4: Reduced example + affect

5 Conclusion and future developments

In this short paper, we highlighted the importance of addressing user stress when performing text tailoring. To the best of our knowledge there's not a single system which tries to use stress as a variable to tailor textual generation. We think that ignoring this factor prevents current systems from reaching a further step towards user engagement, especially in the domain of behavioural change, as stress does impact both reading capabilities and user motivation. We showed a real-world case in which stress does matter, the NeuroFAST project: our post hoc statistical analysis hints at the fact that the workers who took part into the experiment changed their diet significantly when stressed. This result is aligned with previous well-known evidence which state that diet and stress are linked. We then took our previously proposed framework, which dynamically tailors the user to communicate health data, and enriching its components to exploit stress value as an additional tailoring parameter. By doing so, we showed a few preliminary ideas on tailoring text based on the stress level. The new tailoring logic is mainly achieved by reducing the text size, hence drastically reducing the information delivery time, and enforcing its affective component to counterbalance the lack of motivation which stress can cause when following a diet.

5.1 Implementation

Given that this is a work-in-progress project and all of the given examples are mockup (no computational steps or formal definition of the problem were given, as this paper is purely conceptual), we intend to proceed to work on this thematic, including the implementation of a proper stress-based tailoring system to evaluate its effectiveness. Moreover, the proposed text lacked terminology simplification, a phase which is particularly important in healthcare, where the patient often ignores the meaning of medical terminology. Along with an actual system implementation, it is in our interest to design an evaluation framework, which could help us get a realistic idea of the tailoring strategy effects. This would also include considering and inspecting all of the ethical challenges that comes when working with stressed people.

5.2 Exploring different domains

We focused on the dietary domain, but we remind that behaviour change is a vast domain and we're looking forward applying stress-based tailoring in different areas: at the moment we believe that sleeping apnea therapy (CPAP) could take great advantage of this technique as sleep deprivation is indeed a stressful event and tailored text showed to be a promising tool in increasing therapy adherence(Tatousek, 2016). Overall we do hope that this work raised some interesting research questions around a variable which is, as for now, not considered in NLG and user-tailoring.

5.3 Acknowledgments

This research was funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 812882.

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7-KBBE-2010-4 under grant agreement No: 266408. Collaborators from the University of Aberdeen, Rowett Institute gratefully acknowledge financial support from the Scottish Government as part of the Strategic Research Programme at the Rowett Institute.

References

Phil Adams, Mashfiqui Rabbi, Tauhidur Rahman, Mark Matthews, Amy Voida, Geri Gay, Tanzeem Choudhury, and Stephen Voida. 2014. Towards personal stress informatics: Comparing minimally invasive techniques for measuring daily stress in the wild. In *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, pages 72–79.

Luca Anselma, Simone Donetti, Alessandro Mazzei, and Andrea Pirone. 2018. Checkyourmeal!: diet management with nlg. In *Proceedings of the Workshop on Intelligent Interactive Systems and Language Generation (2IS&NLG)*, pages 45–47.

Luca Anselma and Alessandro Mazzei. 2017. An approach for explaining reasoning on the diet domain. In *1st Workshop on Natural Language for Artificial Intelligence co-located with 16th International Conference of the Italian Association for Artificial Intelligence (AI* IA 2017)*, volume 1983, pages 4–17. CEUR Workshop Proceedings (CEUR-WS. org).

Luca Anselma and Alessandro Mazzei. 2018. Designing and testing the messages produced by a virtual dietitian. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 244–253.

Brian G Arndt, John W Beasley, Michelle D Watkinson, Jonathan L Temte, Wen-Jan Tuan, Christine A Sinsky, and Valerie J Gilchrist. 2017. Tethered to the ehr: primary care physician workload assessment using ehr event log data and time-motion observations. *The Annals of Family Medicine*, 15(5):419–426.

Simone Balloccu, Steffen Pauws, and Ehud Reiter. 2020. A nlg framework for user tailoring and profiling in healthcare. In *SmartPhil@ IUI*, pages 13–32.

Kim Binsted, Alison Cawsey, and Ray Jones. 1995. Generating personalised patient information using the medical record. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 29–41. Springer.

Daniel Braun, Ehud Reiter, and Advaith Siddharthan. 2018. Saferdrive: An nlg-based behaviour change support system for drivers. *Natural Language Engineering*, 24(4):551–588.

Fiorella De Rosis and Floriana Grasso. 1999. Affective natural language generation. In *International Workshop on Affective Interactions*, pages 204–218. Springer.

Tao Ding and Shimei Pan. 2016. Personalized emphasis framing for persuasive message generation. *arXiv preprint arXiv:1607.08898*.

Ivan Donadello, Mauro Dragoni, and Claudio Eccher. 2019. Persuasive explanation of reasoning inferences on dietary data.

Mauro Dragoni, Tania Bailoni, Rosa Maimone, Michele Marchesoni, and Claudio Eccher. 2018. Horus. ai-a knowledge-based solution supporting health persuasive self-monitoring. In *International Semantic Web Conference (P&D/Industry/BlueSky)*.

Noemie Elhadad. 2006. Comprehending technical texts: Predicting and defining unfamiliar terms. In *AMIA annual symposium proceedings*, volume 2006, page 239. American Medical Informatics Association.

Gregory Finley, Erik Edwards, Amanda Robinson, Michael Brenndoerfer, Najmeh Sadoughi, James Fone, Nico Axtmann, Mark Miller, and David Suendermann-Oeft. 2018. An automated medical scribe for documenting clinical encounters. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–15.

Marco Guerini, Oliviero Stock, Massimo Zancanaro, Daniel J O'Keefe, Irene Mazzotta, Fiorella de Rosis, Isabella Poggi, Meiyii Y Lim, and Ruth Aylett. 2011. Approaches to verbal persuasion in intelligent user interfaces. In *Emotion-Oriented Systems*, pages 559–584. Springer.

Robert P Hawkins, Matthew Kreuter, Kenneth Resnicow, Martin Fishbein, and Arie Dijkstra. 2008. Understanding tailoring in communicating about health. *Health education research*, 23(3):454–466.

Jennifer A Healey and Rosalind W Picard. 2005. Detecting stress during real-world driving tasks using physiological sensors. *IEEE Transactions on intelligent transportation systems*, 6(2):156–166.

Saar Hommes, Chris van der Lee, Felix Clouth, Jeroen Vermunt, Xander Verbeek, and Emiel Krahmer. 2019. A personalized data-to-text support tool for cancer patients. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 443–452.

James Hunter, Yvonne Freer, Albert Gatt, Ehud Reiter, Somayajulu Sripada, Cindy Sykes, and Dave Westwater. 2011. Bt-nurse: computer generation of natural language shift summaries from complex heterogeneous medical data. *Journal of the American Medical Informatics Association*, 18(5):621–624.

Saskia M Kelders, Harri Oinas-Kukkonen, Anssi Oörni, and Julia EWC van Gemert-Pijnen. 2016. Health behavior change support systems as a research discipline; a viewpoint. *International journal of medical informatics*, 96:3–10.

Matthew W Kreuter and Ricardo J Wray. 2003. Tailored and targeted health communication: strategies for enhancing information relevance. *American journal of health behavior*, 27(1):S227–S232.

Rita Kukafka. 2005. *Tailored Health Communication*, pages 22–33. Springer New York, New York, NY.

Scott H Lee. 2018. Natural language generation for electronic health records. *NPJ digital medicine*, 1(1):1–7.

Peter F Lovibond and Sydney H Lovibond. 1995. The structure of negative emotional states: Comparison of the depression anxiety stress scales (dass) with the beck depression and anxiety inventories. *Behaviour research and therapy*, 33(3):335–343.

Anmol Madan, Manuel Cebrian, Sai Moturu, Katayoun Farrahi, et al. 2011. Sensing the" health state" of a community. *IEEE Pervasive Computing*, 11(4):36–45.

Saad Mahamood and Ehud Reiter. 2011. Generating affective natural language for parents of neonatal infants. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 12–21.

Rosa Maimone, Marco Guerini, Mauro Dragoni, Tania Bailoni, and Claudio Eccher. 2018. Perkapp: A general purpose persuasion architecture for healthy lifestyles. *Journal of biomedical informatics*, 82:70–87.

RM Malone, K Giles, NG Maloney, CL Fyfe, A Lorenzo-Arribas, DB O'Connor, and AM Johnstone. 2015. Effects of stress and mood on caffeine consumption in shift and non-shift workers. *Proceedings of the Nutrition Society*, 74(OCE1).

Kathleen McKeown, Jacques Robin, and Michael Tanenblatt. 1993. Tailoring lexical choice to the user's vocabulary in multimedia explanation generation. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 226–234.

Harri Oinas-Kukkonen. 2013. A foundation for the study of behavior change support systems. *Personal and ubiquitous computing*, 17(6):1223–1235.

Harri Oinas-Kukkonen and Marja Harjumaa. 2009. Persuasive systems design: Key issues, process model, and system features. *Communications of the Association for Information Systems*, 24(1):28.

Cecile L Paris. 1988. Tailoring object descriptions to a user's level of expertise. *Computational Linguistics*, 14(3):64–78.

Steffen Pauws, Albert Gatt, Emiel Krahmer, and Ehud Reiter. 2019. Making effective use of healthcare data using data-to-text technology. In *Data Science for Healthcare*, pages 119–145. Springer.

Peng Peng, Marcia Barnes, CuiCui Wang, Wei Wang, Shan Li, H Lee Swanson, William Dardick, and Sha Tao. 2018. A meta-analysis on the relation between reading and working memory. *Psychological bulletin*, 144(1):48.

Kurt Plarre, Andrew Raij, Syed Monowar Hossain, Amin Ahsan Ali, Motohiro Nakajima, Mustafa Al'Absi, Emre Ertin, Thomas Kamarck, Santosh Kumar, Marcia Scott, et al. 2011. Continuous inference of psychological stress from sensory measurements collected in the natural environment. In *Proceedings of the 10th ACM/IEEE international conference on information processing in sensor networks*, pages 97–108. IEEE.

François Portet, Albert Gatt, Jim Hunter, Ehud Reiter, Somayajulu Sripada, and Feng Gao. 2008. Babytalk: A core architecture to summarise icu data as tailored text.

François Portet, Ehud Reiter, Albert Gatt, Jim Hunter, Somayajulu Sripada, Yvonne Freer, and Cindy Sykes. 2009. Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8):789–816.

Jo-Anne Puddephatt, Gregory S Keenan, Amy Fielden, Danielle L Reaves, Jason CG Halford, and Charlotte A Hardman. 2020. 'eating to survive': A qualitative analysis of factors influencing food choice and eating behaviour in a food-insecure population. *Appetite*, 147:104547.

Manpreet K Rai, Lester C Loschky, and Richard Jackson Harris. 2015. The effects of stress on reading: A comparison of first-language versus intermediate second-language reading comprehension. *Journal of Educational Psychology*, 107(2):348.

Chris Reed, Derek Long, and Maria Fox. 1996. An architecture for argumentative dialogue planning. In *International Conference on Formal and Applied Practical Reasoning*, pages 555–566. Springer.

Ehud Reiter, Roma Robertson, and Liesl M Osman. 2003. Lessons from a failure: Generating tailored smoking cessation letters. *Artificial Intelligence*, 144(1-2):41–58.

Friedrich Riffer, Manuel Sprung, Hannah Münch, Elmar Kaiser, Lore Streibl, Kathrin Heneis, and Alexandra Kautzky-Willer. 2019. Relationship between psychological stress and metabolism in morbidly obese individuals. *Wiener klinische Wochenschrift*, pages 1–11.

Jamie Ross, Fiona Stevenson, Rosa Lau, and Elizabeth Murray. 2016. Factors that influence the implementation of e-health: a systematic review of systematic reviews (an update). *Implementation science*, 11(1):146.

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.

Clare Scott and Alexandra M Johnstone. 2012. Stress and eating behaviour: implications for obesity. *Obesity facts*, 5(2):277–287.

Salzburg Global Seminar. 2011. Salzburg statement on shared decision making. *Bmj*, 342.

Celette Sugg Skinner, Victor J Strecher, and Harm Hospers. 1994. Physicians' recommendations for mammography: do tailored messages make a difference? *American Journal of Public Health*, 84(1):43–49.

Moira A Stewart. 1995. Effective physician-patient communication and health outcomes: a review. *CMAJ: Canadian Medical Association Journal*, 152(9):1423.

Anne M Stiggelbout, Trudy Van der Weijden, Maarten PT De Wit, Dominick Frosch, France Légaré, Victor M Montori, Lyndal Trevena, and Glenn Elwyn. 2012. Shared decision making: really putting patients at the centre of healthcare. *Bmj*, 344:e256.

Lacroix J. Visser T. Den Teuling N. Tatousek, J. 2016. Promoting adherence to cpap with tailored education and feedback: A randomized controlled clinical trial. *Proc. Sleep 2015*.

Susan J Torres and Caryl A Nowson. 2007. Relationship between stress, eating behavior, and obesity. *Nutrition*, 23(11-12):887–894.

Kees Van Deemter and Ehud Reiter. 2018. Lying and computational linguistics. In *The Oxford Handbook of Lying*. Oxford University Press.

Ielka Van Der Sluis and Chris Mellish. 2008. Towards affective natural language generation: Empirical investigations. In *Proceedings of the Symposium on Affective Language in Human and Machine, AISB*, pages 9–16.

David Watson, Lee Anna Clark, and Auke Tellegen. 1988. Development and validation of brief measures of positive and negative affect: the panas scales. *Journal of personality and social psychology*, 54(6):1063.

Sandra Williams and Ehud Reiter. 2008. Generating basic skills reports for low-skilled readers. *Natural Language Engineering*, 14(4):495–525.

Qing Zeng-Treitler, Sergey Goryachev, Tony Tse, Alla Keselman, and Aziz Boxwala. 2008. Estimating consumer familiarity with health terminology: a context-based approach. *Journal of the American Medical Informatics Association*, 15(3):349–356.

Ingrid Zukerman, Richard McConachy, and Sarah George. 2000. Using argumentation strategies in automated argument generation. In *INLG'2000 Proceedings of the First International Conference on Natural Language Generation*, pages 55–62.

Fuzzy Logic for Vagueness Management in Referring Expression Generation[1]

N. Marín, G. Rivas-Gervilla, and D. Sánchez
Dept. Computer Science and A.I., and CITIC-UGR
University of Granada, 18071 Granada, Spain
{nicm,griger,daniel}@decsai.ugr.es

Abstract

In this work we overview some of the contributions regarding the use of Fuzzy Logic in referring expression generation. We also discuss recent advances that can help to overcome the arguments in the literature against the use of Fuzzy Logic in Natural Language Generation.

1 Introduction

Different types of vagueness are present in natural language employed for human communication (van Deemter, 2010). In this paper we are concerned with the kind of vagueness related to concepts, words, and linguistic expressions that allow for borderline cases in which fulfilment is not clear (van Deemter, 2010). This kind of vagueness cannot be properly represented and managed using classical logics, hence alternative tools are necessary (van Deemter, 2010). A widely employed tool for dealing with this kind of vagueness is Fuzzy Logic (Kacprzyk and Zadrozny, 2010).

Fuzzy Logic has been used in Natural Language Generation (NLG) for different purposes (Marín and Sánchez, 2016; Ramos-Soto et al., 2016): modeling the semantics of concepts and expressions, uncertainty representation and quality measurement, among others. In this paper we focus on the use of Fuzzy Logic for the aforementioned purposes in the setting of referring expression generation (REG), one of the crucial tasks of NLG (Gatt and Krahmer, 2018). We also briefly discuss recent contributions that enlarge the available Fuzzy Logic toolbox with new capabilities, solving some issues that have been argued in order to disregard the use of Fuzzy Logic in NLG.

[1]Corresponding author: Gustavo Rivas-Gervilla. Authors appear in alphabetical order. This work has been partially supported by the Spanish Ministry of Science, Innovation and Universities and the European Regional Development Fund - ERDF (Fondo Europeo de Desarrollo Regional - FEDER) under project PGC2018-096156-B-I00, and by the Spanish Ministry of Education, Culture and Sports grant FPU16/05199.

2 Referring expression generation

Given a set of objects $\mathcal{O}$ with properties in $\mathcal{P}$, and given $o \in \mathcal{O}$, the objective of the REG task is to provide a linguistic expression able to identify o within $\mathcal{O}$. It is usual to distinguish two steps in REG: *extraction* and *expression* (Marín and Sánchez, 2016). In the first one, the semantics of the referring expression is represented by means of some knowledge representation formalism, typically a formal logic. In the second one, an appropriate sentence in natural language is provided.

In this paper we are concerned only with the extraction phase in which, in its most basic form, a referring expression is a conjunction of properties in $\mathcal{P}$, usually represented by the set of properties that appear in that conjunction. More general structures can be employed involving negation and disjunction, as well as generalized quantifiers. Other generalizations to the problem are the reference to sets of objects, the use of relational properties represented by mathematical relations between objects, the use of collective properties defined for sets of objects, and the use of *gradual* concepts (Krahmer and Van Deemter, 2012). The latter are the object of our interest here.

3 Vagueness in words and linguistic expressions

Vagueness due to borderline/intermediate cases appears typically because of the use of gradual concepts in language[2]. One example is the concept *large* regarding the size of an object since, within

[2]The terms *gradual* or *fuzzy* are the usual ones in the Soft Computing area for this kind of properties. The term *gradable* is also common in the literature (van Deemter, 2016).

the set of all possible sizes, some values match the concept *large*, some others do not match the concept at all, whilst the rest are intermediate cases that match the concept to a certain extent. Hence, fulfilment of a concept becomes a matter of degree.

Gradual concepts are products of the human mind and abound in human language and communication. As a consequence, they are of primary interest in Artificial Intelligence, particularly when it comes to developing systems for linguistic interaction with humans. One crucial problem is how to represent the semantics of such concepts. It is well known that crisp sets are not well suited for that purpose, since elements either fully belong to the set or to its complement. The only way to represent the semantics of *large* using a crisp set is by giving a size threshold above which *large* holds. However, this solution gives counterintuitive results, since a small variation in size near the threshold is enough to turn a large size into a non-large one when, in fact, both sizes may be even indistinguishable. Indeed, when asked about whether something is *large*, humans do not always provide a yes/no answer, but expressions like "more or less", "so-so", etc. that cannot be represented by a crisp set. Another example is the definition of the concept *heap*, leading to the classical Sorites Paradox (van Deemter, 2010).

Fuzzy Logic is recognized as a suitable tool for representing the semantics of gradual concepts (Rosch, 2013). A fuzzy set F assigns a fulfilment degree in $[0, 1]$ to each value of the domain X where the concept is defined, by means of a *membership function* $\mu_F : X \to [0, 1]$. This way, the semantics of *large* can be represented by means of a continuous function on the set of sizes, in which "small" differences in size produce "small" differences in membership. That is, the transition from being *large* (1) to not being *large* (0) becomes gradual, allowing to represent intermediate cases intuitively by assigning them degrees in $(0, 1)$.

Atomic gradual concepts like *large* can be combined to form *derived* gradual concepts, by means of logical connectives. One of the particularities of Fuzzy Logic is that different operators can be used in order to compute the membership function of derived properties: a wide range of t-norms for intersection, t-conorms for union, and fuzzy negations for complement are available. Other kind of derived gradual concepts can be obtained from the application of *linguistic hedges* that modify the semantics of concepts. The semantics of such derived concepts (like *very large*) are obtained by means of a composition of a function associated to the hedge (*very*, a typical function being $very(x) = x^2$) and the membership function representing the semantics of the concept being modified (*large*).

Gradual concepts can be also employed to form more complex expressions called *protoforms*, that are one of the main objects of study of the *Computing with Words* (CW) area (Zadeh, 1999; Kacprzyk and Zadrozny, 2010). Fulfilment of protoforms is also gradual, degrees in $[0, 1]$ being "computed" from the semantics of the concepts involved.

Protoforms can be expressed linguistically, a paradigmatic example being quantified statements like "*most* of the *large* animals are *slow*", which is a particular instantiation of the protoform "Q of D are A". In this protoform, Q is a gradual quantifier (*most* in our previous example) with semantics represented by a fuzzy set $\mu_Q : [0, 1] \to [0, 1]$ assigning fulfilment degrees of the quantifier to percentages in $[0, 1]$. For instance, a particular semantics of *most* is given by the following continuous and piecewise-linear function:

$$Q(x) = \begin{cases} 0 & x \leq 0.5 \\ 4x - 2 & 0.5 \leq x \leq 0.75 \\ 1 & 0.75 \leq x \end{cases} \quad (1)$$

On its turn, both D and A are fuzzy subsets of the same set X (animals), induced by gradual concepts (*large* and *slow*, respectively). Techniques for computing the fulfilment degree of such sentences, including more complex sentences involving generalized quantifiers, are available (Delgado et al., 2014; Díaz-Hermida et al., 2018).

Graduality can appear in combination with other sources of uncertainty in protoforms, like probability (Zadeh, 1999). Besides, gradual concepts can be used for different purposes in protoforms. A particular case is the *possibilistic use*, in which gradual concepts are employed as restrictions representing the available knowledge. An example of instantiation of such protoforms is "John is *old*", where we lack some knowledge about the actual age of John, but we know it to be restricted to the set of ages that match the gradual concept *old*, membership degrees representing our preference for some ages against others if we had to guess. Note the difference with "I like *old* cars", in which the same gradual concept appears under a *veristic use* and, contrary to the previous case, there is no

uncertainty (the expression claims that I like every old car, membership degrees corresponding to the degree to which I like every car because of its age).

Let us remark that fulfilment of certain protoforms can be computed from other protoforms using rule-based inference, among other kind of reasoning, like in *granular linguistic models of phenomena* (Triviño and Sugeno, 2013).

The use of fuzzy sets has been discussed specifically for REG in (Gatt et al., 2016), where some of the properties in $\mathcal{P}$ are assumed to be gradual. As a consequence, features like referential success of referring expressions become gradual, as we shall discuss in the next section. Though in (Gatt et al., 2016) only conjunctions of atomic properties are considered, it is immediate to extend the discussion to both derived properties and protoforms like those discussed before, for instance under a possibilistic use of gradual concepts (Gatt and Portet, 2016; Marín et al., 2019). The case of quantified statements is also particularly interesting in REG, as it has been shown by using crisp generalized quantification in (Ren et al., 2010).

4 Vagueness and measures

Quality assessment is a fundamental question in any intelligent system, Data2text systems (including REG approaches) not being an exception. Quality assessment models are necessary for two fundamental reasons: (i) they must guide searching processes and (ii) they must allow the results obtained to be evaluated and compared (Bugarín et al., 2015b,a).

The look for quality models is not trivial because they are usually context-dependent and combine aspects that, in many cases, turn out to be subjective and interdependent/conflicting: it is necessary to obey the user's preferences in the context the system is executed (Marín and Sánchez, 2016). In general, it is a multidimensional problem that requires multi-objective optimization algorithms (Castillo-Ortega et al., 2012).

An important part of the mentioned quality models focuses on the definition of measures that allow to assess different aspects of quality with values in $[0, 1]$, even when gradual concepts are not involved. For example we can consider:

- The accuracy, that is, the degree of fulfillment of the expression by the target object set. We have discussed about accuracy of gradual concepts and protoforms in the previous section.
- The brevity, that is, minimizing the length of the expression. Brevity can be measured in $[0, 1]$ when length is divided by the maximum possible length.
- The salience, related to how easy is to perceive the properties employed in the expression. In general, more salient expressions are preferred by users.

As can be seen in these three examples, measures can be gradual, particularly –but not necessarily– in the presence of gradual properties. Fuzzy Logic provides both mechanisms to solve the problem of *symbol grounding* and an extensive prior knowledge in the form of a wide variety of well known fuzzy measures, which can be used as a basis for measuring such quality related aspects.

In the case of referring expressions, the case of referential success deserves special mention. In a *crisp* environment, referential success can be considered as an inherent quality of a referring expression since it makes no sense to use a referring expression lacking referential success. However, in the presence of gradual properties, determining the referential success is also a matter of degree (Gatt et al., 2016): referring expressions with a high degree of referential success are then searched, preferably the best for each target set.

As we have previusly mentioned, Fuzzy Logic permits defining gradual measures of referential success. For example, in (Gatt et al., 2016), a definition of referential success based on accuracy is proposed, measuring to what extent the accuracy occurs in the target set whilst it does not occur in the other distractors. Referential success measurement is also a clear example of how the broad background in measures of Fuzzy Logic can be useful for solving problems in the REG field. For example, studies can be found in the literature (Marín et al., 2016b, 2017a, 2018b) which show that referential success can be assessed using the well-known measures of specificity of Fuzzy Logic (Yager, 1982). In (Gatt et al., 2018) interested readers can find an experimental analysis with users regarding a variety of specificity based measures of referential success.

Additional measures have to be considered in the possibilistic setting mentioned in the previous section. Consider, for example, datasets in which we find objects satisfying a property among a given set of properties, but not knowing exactly which

one. The presence of this type of uncertainty in the information is usually translated into uncertainty in the set of objects that satisfy a given expression. In (Marín et al., 2019) a novel fuzzy set based approach to this problem can be found, where innovative notions like *possible referent* and *necessary referent* of an expression are defined and used as basis for a gradual measure of referential success.

In addition to quality measures of final referring expressions, other fuzzy measures have been proposed for guiding the REG process, such as measures of discriminatory power, for example by means of index-type specificity measures (Marín et al., 2017b), among others.

5 Discussion

Different arguments have been put forward against the use of Fuzzy Logic in modeling and reasoning with concepts in REG and NLG in general.

Fuzzy Logic was deemed unsuitable for representing and dealing with gradual concepts (Osherson and Smith, 1981). Recent work has shown that such claims are erroneous, mainly due to misunderstandings and misconceptions such as using a set theory as a theory of concepts, (Belohlávek et al., 2009; Belohlávek and Klir, 2011; Rosch, 2013). Such use is wrong since, though derived concepts can be obtained by using Fuzzy Logic operations, not every complex concept allows to obtain its semantics by operations on sets, requiring specific modeling of their membership functions instead.

Defining appropriate membership functions representing the semantics of gradual concepts is a complex problem because of subjectivity and context-dependence (Cadenas et al., 2014), among other reasons (van Deemter, 2010). This has been employed as an argument against using Fuzzy Logic as well. However, the same can be said of using crisp sets; for instance, the definition of concepts like *large* using crisp sets requires to define subjective and context-dependent thresholds, the semantics of such models being much more sensitive to small changes on thresholds than when using Fuzzy Logic. In general, the symbol grounding problem is shared by every knowledge representation formalism (Harnad, 1990).

Fortunately, more and more techniques for appropriately solving this problem in different settings are available in the literature, see (Chamorro-Martínez et al., 2017; Ramos-Soto et al., 2019) for recent proposals, the second one in relation to REG. Also related to REG is the proposal in (Marín et al., 2018a), in which the context-dependent semantics of terms like *large*, *medium*, and *small* – interpreted as *the largest*, etc. (van Deemter, 2006) – are automatically calculated according to the collection of size values of the objects in the context, without requiring the intervention of humans beyond fixing once and for all the axioms that the models must satisfy. A similar idea has been employed for modeling the semantics of crisp contextual properties in (Fernández, 2009).

Regarding operations and reasoning, the availability of different ways for performing usual set operations with fuzzy sets and the fact that they are truth-functional (which imply that no Fuzzy Set Theory is a Boolean algebra) has been pointed out as a disadvantage (van Deemter, 2010). The recent development of level-based representations (RLs) as an alternative to fuzzy sets can solve this problem (Dubois and Prade, 2008; Sánchez et al., 2008, 2012; Martin, 2015). RLs have been used in REG (Marín et al., 2016a).

The approach in (Sánchez et al., 2012) represents gradual concepts by means of functions $\rho : (0, 1] \rightarrow 2^X$ instead of fuzzy sets, going beyond Fuzzy Logic in several respects:

- Every classical set operation is extended to the gradual case uniquely in a non-truth functional way by performing the operation in each level of $(0, 1]$ independently, keeping all Boolean properties.
- Fuzzy sets are employed as input (by using α-cuts) and output (measuring membership) only. This way, the understandability and modeling resources of fuzzy sets and the algebraic properties of RLs operations are combined into *RL-systems* that solve some of the drawbacks associated to fuzzy reasoning.

This discussion leads us to believe that Fuzzy Logic, Computing with Words, and RL-systems can help in dealing with graduality/uncertainty in REG and other NLG tasks.

References

Radim Belohlávek and George J. Klir, editors. 2011. *Concepts and fuzzy logic*. The MIT Press, Cambridge, Massachussets.

Radim Belohlávek, George J. Klir, Harold W. Lewis III, and Eileen C. Way. 2009. Concepts and fuzzy sets:

Misunderstandings, misconceptions, and oversights. *Int. J. Approx. Reason.*, 51(1):23–34.

Alberto Bugarín, Nicolás Marín, Daniel Sánchez, and Gracián Triviño. 2015a. Aspects of quality evaluation in linguistic descriptions of data. In *2015 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2015, Istanbul, Turkey, August 2-5, 2015*, pages 1–8. IEEE.

Alberto Bugarín, Nicolás Marín, Daniel Sánchez, and Gracián Triviño. 2015b. Fuzzy knowledge representation for linguistic description of time series. In *2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT-15), Gijón, Spain., June 30, 2015*. Atlantis Press.

José Tomás Cadenas, Nicolás Marín, and María Amparo Vila Miranda. 2014. Context-aware fuzzy databases. *Appl. Soft Comput.*, 25:215–233.

Rita Castillo-Ortega, Nicolás Marín, Daniel Sánchez, and Andrea Tettamanzi. 2012. Quality assessment in linguistic summaries of data. In *Advances on Computational Intelligence - 14th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2012, Catania, Italy, July 9-13, 2012. Proceedings, Part II*, volume 298 of *Communications in Computer and Information Science*, pages 285–294. Springer.

Jesús Chamorro-Martínez, José Manuel Soto-Hidalgo, Pedro Manuel Martínez-Jiménez, and Daniel Sánchez. 2017. Fuzzy color spaces: A conceptual approach to color vision. *IEEE Trans. Fuzzy Syst.*, 25(5):1264–1280.

Kees van Deemter. 2006. Generating referring expressions that involve gradable properties. *Computational Linguistics*, 32(2):195–222.

Kees van Deemter. 2016. *Computational models of referring: a study in cognitive science*. The MIT Press, Cambridge, Massachussets.

Miguel Delgado, M. Dolores Ruiz, Daniel Sánchez, and María-Amparo Vila. 2014. Fuzzy quantification: a state of the art. *Fuzzy Sets Syst.*, 242:1–30.

Félix Díaz-Hermida, Martin Pereira-Fariña, Juan Carlos Vidal, and Alejandro Ramos-Soto. 2018. Characterizing quantifier fuzzification mechanisms: A behavioral guide for applications. *Fuzzy Sets Syst.*, 345:1–23.

D. Dubois and H. Prade. 2008. Gradual elements in a fuzzy set. *Soft Computing*, 12:165–175.

Raquel Fernández. 2009. Salience and feature variability in definite descriptions with positive-form vague adjectives. In *Proceedings Workshop on the Production of Referring Expressions: Bridging the gap between computational and empirical approaches to reference (PRE-CogSci 2009)*.

Albert Gatt and Emiel Krahmer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *J. Artif. Intell. Res.*, 61:65–170.

Albert Gatt, Nicolás Marín, François Portet, and Daniel Sánchez. 2016. The role of graduality for referring expression generation in visual scenes. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems - 16th International Conference, IPMU 2016, Eindhoven, The Netherlands, June 20-24, 2016, Proceedings, Part I*, volume 610 of *Communications in Computer and Information Science*, pages 191–203. Springer.

Albert Gatt, Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2018. Specificity measures and reference. In *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*, pages 492–502. Association for Computational Linguistics.

Albert Gatt and François Portet. 2016. Multilingual generation of uncertain temporal expressions from data: A study of a possibilistic formalism and its consistency with human subjective evaluations. *Fuzzy Sets Syst.*, 285:73–93.

Stevan Harnad. 1990. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1):335 – 346.

Janusz Kacprzyk and Slawomir Zadrozny. 2010. Computing with words is an implementable paradigm: Fuzzy queries, linguistic data summaries, and natural-language generation. *IEEE Trans. Fuzzy Systems*, 18(3):461–472.

Emiel Krahmer and Kees Van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2016a. A measure of referential success based on alpha-cuts. In *Scalable Uncertainty Management - 10th International Conference, SUM 2016, Nice, France, September 21-23, 2016, Proceedings*, volume 9858 of *Lecture Notes in Computer Science*, pages 345–351. Springer.

Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2016b. Using specificity to measure referential success in referring expressions with fuzzy properties. In *2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016, Vancouver, BC, Canada, July 24-29, 2016*, pages 563–570. IEEE.

Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2017a. Referential success of set referring expressions with fuzzy properties. In *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017, Santiago de Compostela, Spain, September 4-7, 2017*, pages 247–251. Association for Computational Linguistics.

Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2018a. An approximation to context-aware size modeling for referring expression generation. In *2018 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2018, Rio de Janeiro, Brazil, July 8-13, 2018*, pages 1–8.

Nicolás Marín, Gustavo Rivas-Gervilla, and Daniel Sánchez. 2019. Referring under uncertainty. In *2019 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2019, New Orleans, LA, USA, June 23-26, 2019*, pages 1–6. IEEE.

Nicolás Marín, Gustavo Rivas-Gervilla, Daniel Sánchez, and Ronald R. Yager. 2017b. On families of bounded specificity measures. In *2017 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2017, Naples, Italy, July 9-12, 2017*, pages 1–6. IEEE.

Nicolás Marín, Gustavo Rivas-Gervilla, Daniel Sánchez, and Ronald R. Yager. 2018b. Specificity measures and referential success. *IEEE Trans. Fuzzy Systems*, 26(2):859–868.

Nicolás Marín and Daniel Sánchez. 2016. On generating linguistic descriptions of time series. *Fuzzy Sets and Systems*, 285:6–30.

Trevor P. Martin. 2015. The X-μ representation of fuzzy sets. *Soft Computing*, 19(6):1497–1509.

Daniel N. Osherson and Edward E. Smith. 1981. On the adequacy of prototype theory as a theory of concepts. *Cognition*, 9(1):35–58.

Alejandro Ramos-Soto, José M. Alonso, Ehud Reiter, Kees van Deemter, and Albert Gatt. 2019. Fuzzy-based language grounding of geographical references: From writers to readers. *Int. J. Comput. Intell. Syst.*, 12(2):970–983.

Alejandro Ramos-Soto, Alberto Bugarín, and Senén Barro. 2016. On the role of linguistic descriptions of data in the building of natural language generation systems. *Fuzzy Sets and Systems*, 285:31–51.

Yuan Ren, Kees Van Deemter, and Jeff Z Pan. 2010. Generating referring expressions with OWL2. In *23rd International Workshop on Description Logics DL2010*, page 420.

Eleanor Rosch. 2013. Neither concepts nor Lotfi Zadeh are fuzzy sets. In *On Fuzziness - A Homage to Lotfi A. Zadeh - Volume 2*, volume 299 of *Studies in Fuzziness and Soft Computing*, pages 591–596. Springer.

Daniel Sánchez, Miguel Delgado, and María-Amparo Vila. 2008. A restriction level approach to the representation of imprecise properties. In *Proceedings Int. Conference on Information Processing and Management of Uncertainty IPMU'08*, pages 153–159.

Daniel Sánchez, Miguel Delgado, María-Amparo Vila, and Jesús Chamorro-Martínez. 2012. On a non-nested level-based representation of fuzziness. *Fuzzy Sets and Systems*, 192(1):159–175.

Gracián Triviño and Michio Sugeno. 2013. Towards linguistic descriptions of phenomena. *Int. J. Approx. Reasoning*, 54(1):22–34.

Kees van Deemter. 2010. *Not Exactly: in Praise of Vagueness*. Oxford University Press.

Ronald R. Yager. 1982. Measuring tranquility and anxiety in decision making: an application of fuzzy sets. *International Journal of General Systems*, 8(3):139–146.

Lotfi A. Zadeh. 1999. From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(1):105–119.