Generating English Summaries of Time Series Data Using the Gricean Maxims

Somayajulu G. Sripada and Ehud Reiter and Jim Hunter and Jin Yu
Dept. of Computing Science
University of Aberdeen
Aberdeen, Scotland, U.K.
+44 (0) 1224 272295

{ssripada,ereiter,jhunter,jyu}@csd.abdn.ac.uk

ABSTRACT

We are developing technology for generating English textual summaries of time-series data, in three domains: weather forecasts, gas-turbine sensor readings, and hospital intensive care data. Our weather-forecast generator is currently operational and being used daily by a meteorological company. We generate summaries in three steps: (a) selecting the most important trends and patterns to communicate; (b) mapping these patterns onto words and phrases; and (c) generating actual texts based on these words and phrases. In this paper we focus on the first step, (a), selecting the information to communicate, and describe how we perform this using modified versions of standard data analysis algorithms such as segmentation. The modifications arose out of empirical work with users and domain experts, and in fact can all be regarded as applications of the Gricean maxims of Quality, Quantity, Relevance, and Manner, which describe how a cooperative speaker should behave in order to help a hearer correctly interpret a text. The Gricean maxims are perhaps a key element of adapting data analysis algorithms for effective communication of information to human users, and should be considered by other researchers interested in communicating data to human users.

Categories and Subject Descriptors

I.2.7 [Natural Language Processing]: Language Generation.

Keywords

Time series data, Summarization, Natural Language Processing, Gricean maxims.

1. INTRODUCTION

Current computer systems present time-series data to human users either graphically or as tables. In contrast, when a human presents time-series data to another human, he or she often uses language to do so. This is especially true when the recipient does not have a lot of domain expertise; as Petre [11] pointed out, graphical presentations of information often work better for experts than for novices. Textual summaries are also useful when information (such as weather and stock market reports) needs to be communicated over low bandwidth devices such as mobile phones.

SIGKDD '03, August 24-27, 2003, Washington, DC, USA.

We are developing techniques to allow computers to generate English summaries of time-series data. In this paper we give an overview of our project, focusing on how we have adapted standard data analysis algorithms, in particular segmentation [6] for the textual summarization task. Although all the adaptations were motivated by empirical work with users and domain experts, it turns out that they can all be viewed as applications of the Gricean maxims of cooperative communication [4]. This suggests that perhaps the Gricean maxims are generally good guidelines for researchers to consider when developing systems that communicate data linguistically. The adaptations we describe in this paper worked well in our three real-world domains. Whether they work in other domains is an open question.

Summarization of time series has been studied from two points of view. One focus has been on determining all the 'significant' or 'novel' or 'surprising' patterns or trends in time series data sets. [2] [6] [8] [10] are all examples of such studies (also see the survey [19]). Such studies are important, but they do not address the human communication issues discussed in this paper. The other focus addresses the visual presentation of results of data analysis to the user. [27] presents visualization of time series on spirals. [26] presents a calendar based visualization of clusters of time series data at multiple time resolutions. In [13] a novel visualization environment called LifeLines is described for presenting patient medical records. Query based visualization methods are described in [7]. These later efforts consider issues of communication separately from data analysis aspects, and do not address linguistic communication.

Several researchers have proposed using fuzzy logic to select words to linguistically communicate information to human users [28]. Such research addresses only a small aspect of the problem of linguistically communicating data; it also to date has ignored the key issue of inter-speaker variability [12][15].

Section 2 of this paper gives background information about Natural Language Generation (NLG), our **SUMTIME** project, and the Gricean maxims. Section 3 describes in a Gricean framework how we have adapted segmentation and other data analysis algorithms for generating textual summaries of data; again we emphasise that these adaptations are all based on knowledge acquisition activities and direct feedback from users and domain experts, none of whom had never heard of Grice. Section 4 summarises our evaluation (to date and planned), and Section 5 presents concluding thoughts.

2. BACKGROUND

2.1 Natural Language Generation

Natural Language Generation (NLG) is the subfield of Artificial Intelligence and Natural Language Processing that uses AI techniques to generate texts in English or other human languages, typically from some non-linguistic input. NLG systems often have three stages [14]:

Document Planning – Decides what information to communicate in the generated texts, and how this information should be organized in the text. Document planning algorithms usually are based primarily on domain knowledge instead of linguistic knowledge, and often are adaptations of AI reasoning techniques such as planning.

Microplanning – Decides how the selected information should be communicated linguistically; for example, which nouns and noun phrases should be used to describe entities in the domain. Microplanning algorithms are based on both domain and linguistic knowledge. Microplanning does not effect the information content of generated texts, but it has enormous impact on the readability of generated texts.

Realisation – Produces grammatically correct output text; it is the realiser's job to ensure that the (often quirky) rules of English grammar are obeyed. Realisers must also ensure that genre and sublanguage constraints are obeyed [5]. Realisation is mostly based on linguistic knowledge.

2.2 SumTime

The goal of the **SUMTIME** project is to develop generic techniques for summarizing time series data. For more information about **SUMTIME**, see *http:\\www.csd.abdn.ac.uk\research\sumtime*. **SUMTIME** is working in three different domains:

- *Meteorology* (**SUMTIME-MOUSAM**) [25]: the input to the system is numerical weather parameters (such as wind speed, temperature, and precipitation) which are predicted by a numerical weather prediction (NWP) model. The output is a weather forecast; to date we have concentrated on marine forecasts for offshore oilrigs. **SUMTIME-MOUSAM** is being developed in collaboration with WNI/Oceanroutes. Table 1 shows some of the wind data predicted by an NWP model. This is part of the input to the **SUMTIME-MOUSAM**. Figure 1 shows an extract from a generated forecast showing all the fields of a marine forecast. The Wind10M and Wind50M statements are generated from the data shown in table 1.

- *Gas turbines* (**SUMTIME-TURBINE**) [29]: the input to the system is sensor readings from a gas turbine, such as fuel flow and exhaust temperature. The output is a textual summary of significant patterns and trends in the data, which is intended to give an engineer a quick summary of the turbine's behavior over a time period. **SUMTIME-TURBINE** is being developed in collaboration with Intelligent Applications. Figure 7 shows sample sensor data from a gas turbine and Figure 9 shows part of the corresponding summary generated by an initial prototype of **SUMTIME-TURBINE**.

- *Intensive care* (**SUMTIME-NEONATE**) [23]: the input to the system is physiological data such as mean blood pressure and heart rate, for a baby in a neonatal intensive care unit. The output is a summary of important trends; it is intended more as an aid to retrospective analysis of a baby's progress than as a decision-support tool. **SUMTIME-NEONATE** is being developed in conjunction with the Simpson Maternity Hospital in Edinburgh, Scotland. Figure 6 shows a sample heart rate (HR) plot and Figure 8 shows its corresponding summary generated by an initial prototype of **SUMTIME-NEONATE**.

Table 1. Example Weather Data

Day	Hour	Wind Dir	Wind Speed 10m	Wind Speed 50m	Gust 10m	Gust 50m
13-06-02	0000	WSW	12.0	15.0	15.0	19.0
13-06-02	0300	WSW	15.0	19.0	19.0	23.0
13-06-02	0600	WSW	19.0	24.0	24.0	30.0
13-06-02	0900	WSW	18.0	22.0	22.0	28.0
13-06-02	1200	W	17.0	21.0	21.0	27.0
13-06-02	1500	W	15.0	19.0	19.0	23.0
13-06-02	1800	WSW	13.0	16.0	16.0	20.0
13-06-02	2100	WSW	11.0	14.0	14.0	17.0
14-06-02	2400	WSW	11.0	14.0	14.0	17.0

3. FORECAST 0 - 24 GMT, Thu 13-Jun 2002
WIND(KTS)
10M: WSW 10-15 increasing 17-22 by early morning, then gradually easing 9-14 by midnight.
50M: WSW 13-18 increasing 22-27 by early morning, then gradually easing 12-17 by midnight.
WAVES(M)
SIG HT: 0.5-1.0 rising 1.5-2.0 by early morning, then falling 0.5-1.0 by midnight.
MAX HT: 1.0 or less rising 2.0-2.5 by early morning, then falling 1.0-1.5 by midnight.
PERIOD(SEC)
WAVE PERIOD: 2-4 rising 5-7 by morning, then falling 3-5 by midnight.
WINDWAVE PERIOD: 2-4 rising 5-7 by morning, then falling 3-5 by midnight.
SWELL PERIOD: 5-7 rising 8-10 by midday, then falling 5-7 by midnight.
WEATHER: Cloudy with light rain becoming partly cloudy around midnight.
VIS(NM): Greater than 10 reduced to 5-8 in precipitation.
AIR TEMP(C): 9-11 rising 12-14 in the early evening falling 10-12 around midnight.
CLOUD(OKTAS/FT): 6-8 ST/SC 100-300 lifting 2-4 ST/SC 500-700 around midnight.

Figure 1. Forecast Text for one forecast period showing all the fields of a marine forecast. The first two fields (wind10M and wind50M) are produced from the data shown in Table 1.

The most mature **SUMTIME** system is the meteorological one, **SUMTIME-MOUSAM**, and we will focus on it in this paper, although we will also refer to the other systems as well.

SUMTIME-MOUSAM is currently deployed and in operational use at WNI/Oceanroutes, as described in Section 4.

All the **SUMTIME** systems use the three-stage architecture described in section 2.1. The focus of this paper is on document planning. Very briefly, microplanning in **SUMTIME** was based on rules derived from an analysis of corpora of human-written textual summaries of time-series data [15][24]. These rules were revised based on feedback from users and domain experts [17].

Realisation in **SUMTIME** is done in a fairly straightforward fashion, using software that was adapted from a previous project. This software can produce output in Word or HTML as well as plain text, although we do not currently use this facility.

Table 2. Example Wind Speed and Direction Data

Day	Hour	Wind Direction	Wind Speed (Knots)
20-1-01	600	S	8
20-1-01	900	S	6
20-1-01	1200	S	7
20-1-01	1500	S	10
20-1-01	1800	S	12
20-1-01	2100	S	16
21-1-01	0000	S	20

2.2.1 Document Planning in SUMTIME

Probably the most important part of **SUMTIME**, and the focus of this paper, is document planning, that is selecting the content of the generated texts. For example, consider the meteorological time series data sets of wind direction and wind speed shown in Table 2. There are 7 data points in this time series, and (oversimplifying to some degree) the job of the **SUMTIME-MOUSAM** document planning module is to decide which of these 7 points should be mentioned in the textual summary. At least one point must be mentioned, so there are 2**7-1 = 127 possible contents for this summary, some of which are shown in Figure 2. As is standard in weather reports, we report numbers as ranges around the actual data value, and do not repeat wind direction if it is unchanged (such details are taken care of by the microplanner and realiser in **SUMTIME**).

Document planning in **SUMTIME-TURBINE** and **SUMTIME-NEONATE** is a bit different in detail, because these systems work with much denser time series. Typically they generate summaries of one-second data over a period of several hours, so listing all data points is inconceivable. But the same basic problem arises in these systems after an initial processing step has detected patterns (**SUMTIME-TURBINE**) or trends (**SUMTIME-NEONATE**): which of the detected patterns and trends should be mentioned in the text summary?

In **SUMTIME-MOUSAM**, we conducted an initial set of knowledge acquisition (KA) activities, based on working with experts and on analyzing a corpus of human-written forecasts [20][25]. Based on these activities, we decided to use linear segmentation [6] for document planning. That is, we would fit linear line segments to the data using standard segmentation algorithms, and only mention in the text the endpoints of these segments. In the example of Table 2, for instance, wind direction is steady while wind speed rises fairly steadily. Hence, this data can be approximated by a single line segment from 0600 to 0000, and hence only these points should be mentioned in the text. In other words, according to this model the best summary is the second one in Figure 2, "S 06-10 INCREASING 18-22 BY MIDNIGHT".

S 08-12 (*just 1500 value*)

S 06-10 INCREASING 18-22 BY MIDNIGHT (*0600, 0000 vals*)

S 06-10 DECREASING 05-09 BY MIDDAY, INCREASING 10-14 BY EARLY EVENING AND 18-22 BY MIDNIGHT (*0600, 1200, 1800, 0000 values*)

S 06-10 DECREASING 04-08 BY LATE MORNING, INCREASING 05-09 BY MIDDAY, 08-12 BY AFTERNOON, 10-14 BY EARLY EVENING, 14-18 BY MID EVENING, AND 18-22 BY MIDNIGHT (*0600, 0900, 1200, 1500, 1800, 2100, 0000 values*)

Figure 2. A few possible summaries of Table 2 Data

Segmentation worked reasonably well in most cases, but as we conducted more knowledge acquisition activities, expanded the range of the system to cover more cases, and began to get feedback from forecasters who were attempting to use the system to generate real forecasts, we realized that the basic segmentation algorithm needed to be adjusted and tweaked in several ways, in order to be suitable for the task of generating textual summaries. In other words, segmentation needs to be adjusted if it is used to communicate information to people, instead of for classic data mining. As we get more experience with **SUMTIME-TURBINE** and **SUMTIME-NEONATE** (which are less advanced than **SUMTIME-MOUSAM**), we are also discovering that changes need to be made to the data analysis algorithms used in these systems, in order to enhance their effectiveness at communicating information to humans.

These observations, plus comments made to us by domain experts during knowledge acquisition exercises, suggest that analyzing data for the purpose of communicating a textual summary to a person is not identical to analyzing data for data mining purposes. Communication involves two agents (speaker/writer and hearer/reader) and successful communication with a human requires the speaker/writer (in our case, the computer) to obey the pragmatic rules associated with human communication.

In fact, after we had accumulated sets of algorithm modifications and adjustments for all of the **SUMTIME** systems, based on empirical work with users and domain experts, we realized that most of these modifications could be regarded as instances of the Gricean maxims, which are a well known set of pragmatic rules for linguistic communication between humans.

2.3 Gricean Maxims

The Gricean maxims of conversational implicature are a set of pragmatic principles which essentially state how human hearers expect human speakers to behave [4][9]. The maxims are shown in Figure 3.

Maxim of Quality: Try to make your contribution one that is true. More specifically:

1. Do not say what you believe to be false.
2. Do not say that for which you lack adequate evidence.

Maxim of Quantity:

1. Make your contribution as informative as is required (for the current purposes of the exchange).
2. Do not make your contribution more informative than is required.

Maxim of Relevance: Be relevant.

Maxim of Manner: Be perspicuous. More specifically:

1. Avoid obscurity of expression.
2. Avoid ambiguity.
3. Be brief.
4. Be orderly.

Figure 3. The Gricean Maxims (excerpted from [Grice 1975; page 65])

The maxims are vaguely defined and need task (and domain) specific interpretations to be used in computational models. There are many complexities to the precise definition and implementation of the maxims (for example, [1]). However, in **SUMTIME** we do not attempt to precisely define the maxims, instead we treat them as general categories which suggest the types of changes that need to be made to data analysis algorithms when they are used for linguistic communication.

3. APPLICATION OF GRICEAN MAXIMS

In this section we describe how each of the maxims presented in section 2.3 are applied to the task of generating textual summaries of time-series data in our three domains.

3.1 Maxim of Quality

The maxim of Quality states that the information in a text should be true; in a sense, it can also be called the maxim of truthfulness. In our case, it means that a summary should report true values from the input data set.

For example, **SUMTIME-MOUSAM** uses linear segmentation to decide what information to mention in forecasts, as described above. Probably the most common form of segmentation in data mining applications uses linear regression to fit line segments to data [6]. However, the endpoints of line segments produced by such segmentation in some cases are generally not actual data set values, as illustrated below.

Consider the wind speed data set shown in Table 2. For this data set, segmentation using linear regression lines yields one segment as shown in Figure 4. For the same data set, segmentation using linear interpolation lines (line joining end points) also yields one segment as shown in Figure 5.

This means that both methods fit all the input data into one rising trend. However, the end points of the "best-fit" line segment produced by the linear regression shown in Figure 4 are not true data points; in particular, at 0600 this line segment has a wind speed value of about 5 knots, whereas the true data value is 8 knots. The end points of the line segment produced by linear interpolation (shown in Figure 5), in contrast, are true data values. Although visually regression looks better, forecasters DO NOT use it while writing forecasts.

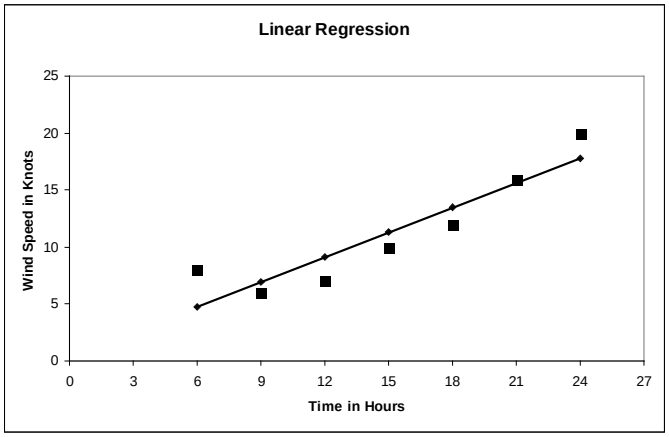

Figure 4. Linear Regression of Data from Table 2.

In other words, using linear regression would suggest a forecast such as S 03-07 INCREASING 16-20 BY MIDNIGHT, whereas using linear interpolation would suggest a forecast such as S 06-10 INCREASING 18-22 BY MIDNIGHT. In such cases human forecasters consistently take the second approach. They believe that data points mentioned in the forecast should accurately reflect what is in the source data (Numerical Weather Prediction); in other words, they are following the Gricean maxim of Quality. Hence **SUMTIME-MOUSAM** also uses this strategy.

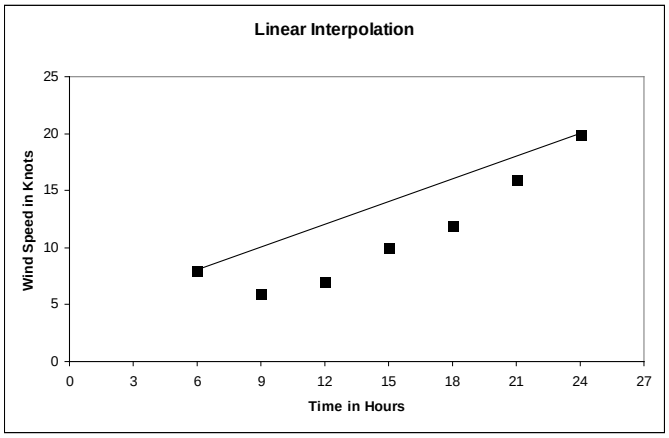

Figure 5. Linear Interpolation of Data from Table 2.

The maxim of Quality also states that texts should not include information for which there is inadequate evidence.

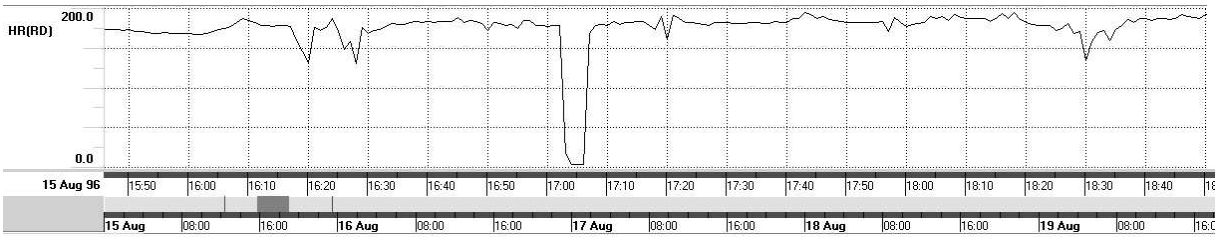

Figure 6. Plot of Heart Rate.

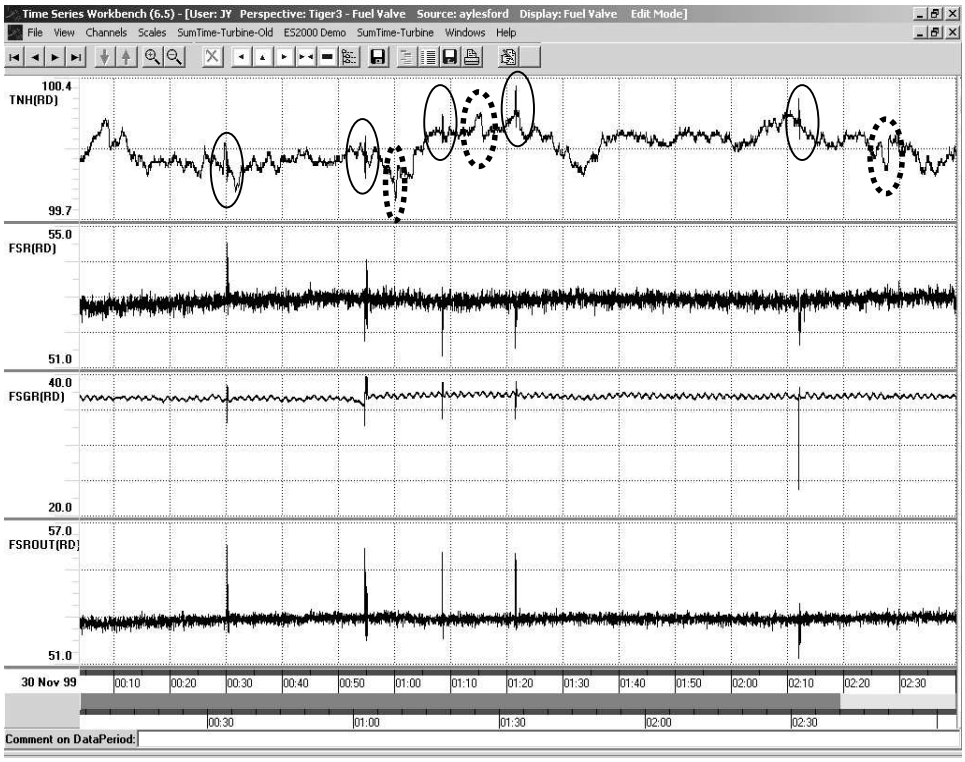

Figure 7. Context dependent spike detection. Spikes in marked with dashed circles do not correspond to spikes in other channels.

In our case, this means that if there is significant uncertainty about the data, this should be communicated to the user. This is important in **SUMTIME-MOUSAM**, since the input to the system is a numerical weather prediction (output of a numerical weather simulation), and there will be some differences between the predicted weather values and the actual weather outcome. For this reason human forecasters deliberately make forecasts vague, by using ranges for wind speeds (10-14 instead of 12), by not giving precise directions (S instead of 176 degrees), and by using vague time phrases (BY AFTERNOON instead of 1500). **SUMTIME-MOUSAM** does likewise.

3.2 Maxim of Quantity

The maxim of Quantity states that texts should present as much information as needed, but no more. In our case this in particular means that texts should describe patterns and trends that are helpful to the user, and omit patterns and trends that are not helpful; even if the non-helpful patterns and trends are larger than the helpful ones.

For example, in **SUMTIME-NEONATE**, consider the heart rate plot shown in Figure 6. Usually raw data from ICU such as the sample data shown in Figure 6 contains a number of artifacts due to external events such as baby handling and blood sampling. These artifacts need to be separated from the input data before summarizing. The example data shown in Figure 6 contains one blood sampling artifact at 17:04 when heart rate drops almost to 0. One of the doctors, with whom we did KA explained that physiological data without artifacts reflect the 'true physiology' of the underlying baby. Therefore, while summarizing physiological data such as the one shown in Figure 6, it is important to remove all artifacts from raw data. In our work we have used artifact removal algorithms developed in the **NEONATE** project [3] and generated the summary shown in Figure 8. Thus although the drop at 17:04 is the most visually dramatic event, because it is an artifact it should not be reported.

The HR is steady at about 180-190 throughout the period.

Figure 8. Prototype generated summary for the data shown in Figure 6.

A related phenomenon happens in **SUMTIME-TURBINE**. This system describes spikes and other patterns, and a key part of generating its text summaries is deciding whether a spike represents a real event in the turbine or whether it is noise or an artefact. **SUMTIME-TURBINE** uses contextual information to make this decision (for example, what is happening in other data channels at the time a potential spike is detected), which means that it may report one spike but not another, even if both spikes have exactly the same size (measured either absolutely or relative to a channel's standard deviation). For example, in Figure 7 spikes marked by circles and dashed circles are both roughly of the same size. However, spikes in dashed circles do not correspond to spikes in other channels and therefore are not reported to the user as shown in Figure 9.

During this period, spikes simultaneously occur around 00:29, 00:54, 01:08, 01:21, and 02:11 in these channels.

Figure 9. Extract from the Prototype generated summary for the data shown in Figure 7.

3.3 Maxim of Relevance

The maxim of Relevance states that the information should be relevant to the user. Obviously there is a considerable overlap between Relevance and Quantity, but from our perspective we interpret Relevance to mean that generated texts should be relevant to a specific user. In other words, textual summaries should present information that is relevant to a particular user or set of users; thus, systems that generate textual summaries should be controlled by models of the particular interests and tasks of specific users.

This is described in more detail in [18]. To take one example, in most cases light winds have little effect on oil rig operations, so there is no need to report small changes in direction when the wind is light. However, different wind speeds are considered light for different oilrigs (in the sense of having little effect on operations). For example, North Sea oilrigs generally consider winds of 15 knots or less to be light, but Persian Gulf oil rigs, which are built for less severe weather conditions, generally consider winds of 10 knots or less to be light. **SUMTIME-MOUSAM** allows users to specify such information via a control table as shown in Table 3.

Table 3. Example control table showing the direction changes for a particular oilrig

Wind Speed	Direction Change
0 - 15	44
15 - 40	22
40 - 65	22
> 65	22

Table 3 shows a control table compiled for one particular oilrig in the North Sea. This table is used for segmenting wind speed and wind direction data. Here, the first column shows the different speed ranges. For each of these ranges, the second column shows the changes in direction that should be reported. For example, below 15 knots of wind speed changes in wind direction below 44 degrees need not be reported.

In principle, summaries should also take into account a user's detailed plans. For example, small changes in wind direction are important even in light winds if certain unusual operations (such as flaring gas) are planned. Unfortunately it is not currently possible for **SUMTIME-MOUSAM** to obtain detailed information about a user's plans, so summaries cannot take this information into account.

3.4 Maxim of Manner

3.4.1 Avoid Obscurity of Expression

This submaxim says that information should be expressed in the most appropriate *linguistic* manner, which may not be the most natural way from the perspective of a data analysis algorithm.

In **SUMTIME-MOUSAM**, for example, segmentation sometimes produces flat segments where values do not change over a time period. From a data analysis perspective, there is no difference between flat segments and segments in which values are rising or falling. From a linguistic perspective, however, the standard way of writing weather reports is not to explicitly mention times when the wind does not change; since this is what users are used to, it is important that **SUMTIME-MOUSAM** respect this convention.

For example, consider the wind data set shown in Table 4.

Table 4. Example Wind Speed Data

Day	Hour	Wind Speed (Knots)
20-1-01	0600	8
20-1-01	0900	14
20-1-01	1200	22
20-1-01	1500	22
20-1-01	1800	22
20-1-01	2100	26
21-1-01	0000	30

The wind speed data shown in Table 4 initially rises sharply from 8 to 22, then remains at 22 for a time period, and finally rises more gradually from 22 to 30. Segmentation divides this data into three segments (0600-1200; 1200-1800; 1800-000), and our initial idea was to explicitly describe each segment in the forecast, for example

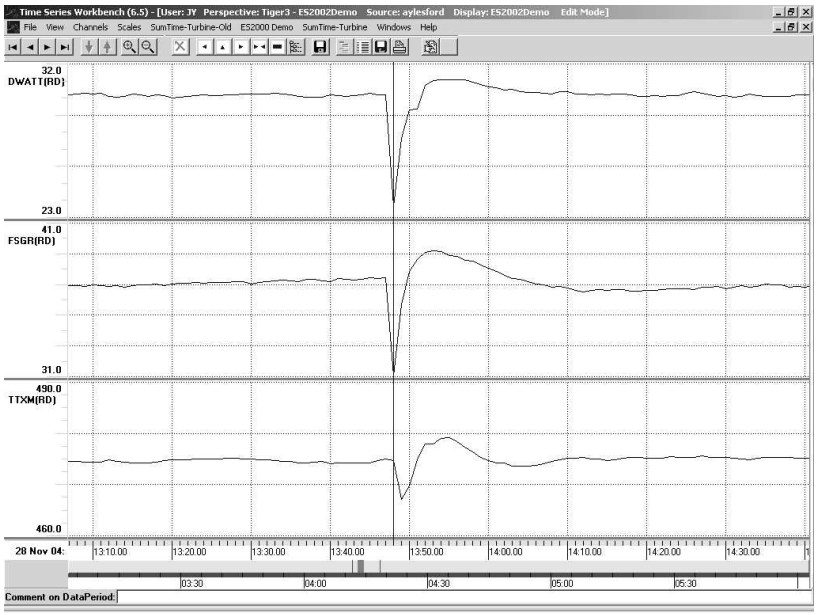

Figure 10. Spike in the bottom channel occurs later than the spikes in the first two channels

06-10 RAPIDLY INCREASING 20-24 BY MIDDAY, REMAINING 20-24 UNTIL EVENING, THEN RISING 28-32 BY MIDNIGHT.

However, human forecasters do not write in this manner. Instead, they would just describe the two segments where the wind speed changes, using a *from* clause to indicate when the second rise for example

06-10 RAPIDLY INCREASING 20-24 BY MIDDAY, RISING FROM EVENING TO 28-32 BY MIDNIGHT.

Thus, although the data analysis algorithm treats flat segments and rising/falling segments identically, these are described differently in language, and this is something a text summarization system must be sensitive to.

3.4.2 Avoid Ambiguity

This submaxim states that generated texts should not be ambiguous. In all of the **SUMTIME** systems, this is an issue when describing events in multiple channels. In **SUMTIME-TURBINE**, for example, if similar patterns are seen in several channels which measure related sensor information, and these patterns are very close temporally but not at exactly the same time, then these patterns probably reflect the same underlying physical event, and they should be described together as occurring at roughly the same time. For example, in Figure 10 it is better to say that *spikes in DWATT, FSGR and TTXM channels at around 04:13* instead of *spike in Channel DWATT and FSGR at 04:13:48, and in TTXM at 04:13:49*. Consistency is also very important in **SUMTIME-MOUSAM**, and human forecasters try hard to present consistent messages about the weather in the different descriptors. For example, it is not acceptable to report in the weather descriptor that cloud and rain are expected, and then to report in the visibility descriptor that good visibility is expected (please refer to Figure 1 to see the fields of a typical marine forecast). Our knowledge acquisition sessions have suggested that human forecasters form a qualitative overview of the weather data before actually writing a forecast [21], and this is a key part of ensuring consistency and avoiding ambiguity. We are currently working on generating an initial overview in **SUMTIME-MOUSAM** and using this to ensure consistency in the texts.

3.4.3 Be Brief

This submaxim states that texts should be brief. In a sense this is a fundamental imperative behind all summarization, to try to produce brief texts that summarize important aspects of the data. This submaxim also effects the way information is expressed linguistically, because it encourages repeated information to be omitted (elided). For example, consider the difference between

S 8-12 RISING S 18-22 BY MIDDAY, RISING S 35-40 BY EVENING

S 8-12 RISING 18-22 BY MIDDAY AND 35-40 BY EVENING

In the second version, which is what human forecasters and **SUMTIME-MOUSAM** would produce, the wind direction is omitted from the second and third wind descriptors, and the verb RISING is omitted from the third desciptor. Such aggregation and ellipsis [14] is a useful technique for making texts shorter than they otherwise would be.

3.4.4 Be Orderly

This submaxim essentially states that information should be ordered using a consistent strategy without confusing the reader. This is an issue when **SUMTIME-TURBINE** and **SUMTIME-NEONATE** are describing events in multiple channels: these descriptions can be ordered either by channel or by time. We have observed that humans aim to be consistent in their strategy while ordering information and do not switch between strategies in the same description.

4. EVALUATION

Initial evaluations of **SUMTIME-MOUSAM** and **SUMTIME-TURBINE** were reported in [22][29]. However, these evaluations focused on mathematical features such as text length and differences between segments and the data, they did not involve users and domain experts and they did not evaluate the communicative issues which are the focus of this paper.

We are currently conducting a post-edit evaluation of **SUMTIME-MOUSAM.** That is, we are analyzing the changes that human forecasters make to **SUMTIME-MOUSAM** forecasts before releasing them to the end users.

More precisely, WNI currently use **SUMTIME-MOUSAM** as shown in Figure 11. The raw NWP data is first edited by WNI staff using the WNI Editing Tool, based on their specialized meteorological knowledge. **SUMTIME-MOUSAM** is then run on the output of the Editing Tool (Data 1 in Figure 11), in order to describe the edited data to the forecaster himself. This produces Forecast Text 1 in Figure 11. The forecaster then adjusts the data (not the text) in cases where he believes the NWP prediction is incorrect; this produces modified data (Data 2). **SUMTIME-MOUSAM** is then run again on the modified data, to produce a draft forecast for the customer; this is Forecast Text 2. The forecaster then manually edits this text, and produces the final Forecast Text 3.

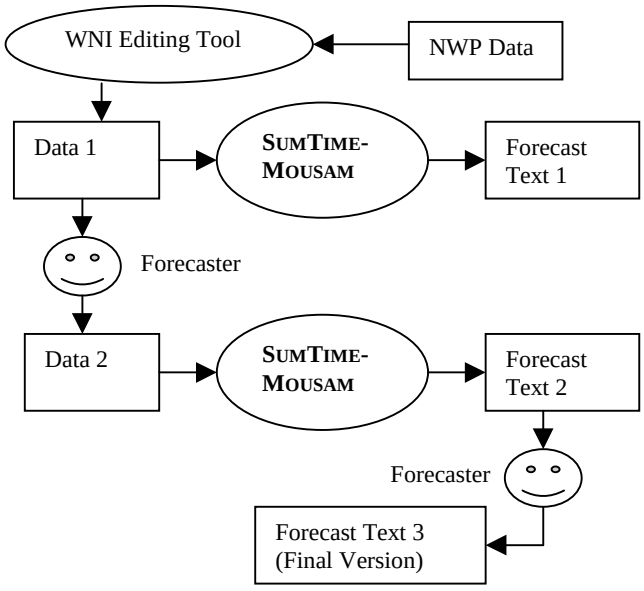

Figure 11. Testing Scheme used in Industrial Evaluation

Our currently running evaluation measures the edits made by the forecaster in the final step, when he produces Forecast Text 3 from Forecast Text 2. This measures how many problems need to be fixed in the generated texts before they are fit to be used to communicate information effectively for humans. The procedure used for evaluation is explained below with the following (real) example:

1. Computer Output (Forecast Text 2)

WIND(10M): SSE 18-23 gradually backing SE 16-21.

2. Forecaster edited text (Final Forecast Text)

WIND(10M): SSE 18-22 gradually backing SE.

Step 1: We first divide the forecast text into its constituent phrases. For instance, forecast text 2 has two phrases, '*SSE 18-23*' and '*gradually backing SE 16-21*'. Similarly forecast text 3 has two phrases, '*SSE 18-22*' and '*gradually backing SE*'.

Step 2: Next, we align the phrases from forecast text 2 with the corresponding phrases from forecast text 3 for comparison. The first phrase in forecast text 2 is always forced to align with the first phrase in forecast text 3. For instance, phrase '*SSE 18-23*' of forecast text 2 is aligned with '*SSE 18-22*' of forecast text 3. Subsequent phrases are aligned based on the parameter values such as 'wind direction' and 'wind speed range'. For instance, the second phrase '*gradually backing SE 16-21*' is aligned with '*gradually backing SE*' because wind direction SE is same in both these phrases. When parameter alignment is not possible then aligment is based on time phrases, and change verbs. In some cases all alignment criteria fail, usually because there is a difference in the number of segments produced by the forecaster and **SUMTIME-MOUSAM**. In other words, in these cases the forecaster segmented the input data differently from **SUMTIME-MOUSAM**.

Step 3: In this step we compare the aligned phrases from forecast text 2 and forecast text 3 word by word and categorized the additions, deletions and changes made by the forecaster. For instance, the second phrase in forecast text 2 has been edited once by deleting the speed range '16-21', we categorise this as an ellipsis change (additional phrase deleted).

Table 5. Frequencies of Different Types of Forecaster Edits

Type	Edits	Frequency
1	Data changes	117
2	Lexical changes: connectives (125), time phrases (79), verbs (61), size of speed range (35), adverb (2)	302
3	Ellipsis changes: additional phrases deleted (88), deleted phrases restored (50).	138
4	Other changes: adverb deleted (57), adverb added (13), gusts or showers added (21), gusts or showers deleted (16)	107

We have carried out the above mentioned procedure manually on the Wind (10M) texts produced by WNI for the month of February 2003. The total number of phrases analysed was 607. Out of this, 125 phrases (20%) could not be aligned. As stated above these phrases account for segmentation differences between forecaster and our system. Almost all of these changes reflected a difference in segment merging; typically forecasters merged more than **SUMTIME-MOUSAM**. We believe this is because forecasters were using more complex user-sensitive merging controls than the simple table used by **SUMTIME-MOUSAM**; in other words, we didn't go far enough in implementing the maxim of Relevance.

178 phrases (30%) matched completely word to word. In other words, in these cases forecaster did not feel the need for any modifications before communicating to the end user. In all the

other phrases forecasters made 664 edits. We have noticed that some of the edits had cascading effect, edits in the initial phrases of forecast texts lead to further edits in subsequent phrases. We have categorized all the edits into types and counted their frequencies as shown in Table 5. The middle column shows the edits with their individual frequencies shown in parantheses.

Edit type 1, which is made 117 times (18%), is changes made to the data reported in the text. For example, the forecaster changing the wind direction from SSW to SW. WNI has suggested to us that this could be due to forecasters wishing to make additional small data changes, but making these directly to the forecast text instead of changing Data 2 (in Figure 11), as it is quicker to change the text directly if the change is small.

Edit type 2 is changes in the words or phrases (lexemes) used in the forecasts. For example, the forecaster replacing the connective *then* by the connective *and*. These account for 302 edits (46%) out of the total 664 edits. Some of these edits reflect problems in the system; for example forecasters try to avoid repeating the same word, whereas **SUMTIME-MOUSAM** is happy to use a word several times in a sentence. Others are due to forecasters having different individual preferences about which words and phrases they use in their forecasts [15] [16]. Because users value consistency, **SUMTIME-MOUSAM** does not allow forecasters to specify individual lexical preferences, but forecasters can directly edit texts to reflect their preferences.

Edit type 3 is changes to ellipsis (discussed in Section 3.4.3, under the Maxim of Manner). These account for 138 edits (21%). Forecasters generally performed more ellision than **SUMTIME-MOUSAM**; this suggests that as with the Maxim of Relevance and segment merging, we did not go far enough in implementing this maxim.

Finally, edit type 4 is additions or deletions to the information expressed in forecast phrases. These account for 107 edits (16%). Some of these edits (especially adverbs added/deleted) have already been analysed and our system has been modified to reflect these changes. Initial feedback about these changes from the forecasters has been positive and we intend to confirm it with further evaluation.

We also plan to conduct an evaluation with users (oil company staff) later in 2003. In this evaluation we will show the users three versions of a forecast text (manually authored, computer generated as described above, computer generated by standard segmentation without the communicative algorithm adjustments described here), and ask them to order the versions, stating which they like best and which they like least.

At some point in the more distant future we may also conduct a task-oriented evaluation, where we ask users to make decisions about weather-dependent tasks after reading a forecast, and measure the time needed to make the decision and whether the decision is correct. This type of evaluation is the most difficult and costly to perform, so we will not start this until we have results from the forecaster-edit and user-preference evaluations.

5. CONCLUSION

We have adapted existing data analysis techniques to summarize time series data. The adaptations we made were based on empirical work with users and experts, but turn out to fit well into Grice's maxims for human speakers. This is summarized in Figure 12.

The fact that our independently-suggested modifications do fit the Gricean framework suggests that perhaps Grice's rules are a key element of adapting data analysis algorithms for the generation of textual summaries of data; it would be very interesting to explore this in other types of summaries of numeric data, beyond time series. The details of adaptation may, however, vary from domain to domain.

Maxim of Quality: Try to report true values from the input data. More specifically:

1. Use linear interpolation instead of linear regression.
2. Communicate uncertainty in the input data to the user.

Maxim of Quantity:

1. Describe trends and patterns that are helpful to the users.
2. Omit trends and patterns that are not helpful even if they are the same size or larger.

Maxim of Relevance: Present only information that is relevant to a particular user or set of users.

Maxim of Manner:

1. Avoid obscurity of expression – information should be expressed in the most appropriate linguistic manner.
2. Avoid ambiguity – summary texts should give a consistent message about what is happening.
3. Be brief – summarize only the important aspects of the input data, use aggregation and ellipsis to reduce length.
4. Be orderly – describe events using a consistent ordering strategy.

Figure 12. The Gricean Maxims interpreted for data summarization

We also wonder if Grice's rules could usefully be applied to other techniques for communicating information about time-series data, such as visualization. We are not experts in visualization, but we encourage colleagues who are interested in visualization to explore this, it certainly would be very interesting if there were common rules behind the effective communication of data to people regardless of whether the communication was graphical or linguistic.

6. ACKNOWLEDGEMENTS

Many thanks to our collaborators at WNI/Oceanroutes, Intelligent Applications and Neonate project, especially Ian Davy, Dave Selway, Rob Milne, Jon Aylett, and Neil McIntosh; this work would not be possible without them! This project is supported by the UK Engineering and Physical Sciences Research Council (EPSRC), under grant GR/M76881.

7. REFERENCES

[1] Dale R. and Reiter E. Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions. Cognitive Science 19:233-263, 1995.

[2] Dasgupta, D. and Forrest, S. Novelty Detection in Time Series Data using Ideas from Immunology. In: Proceedings of the 5th International Conference on Intelligent Systems, Reno, June 19-21, 1996.

[3] Ewing G., Ferguson L., Freer Y., Hunter J. and McIntosh, N. Observational Data Acquired on a Neonatal Intensive Care Unit. Technical Report AUCS/TR0205, Dept. of Comp. Science, Univ. of Aberdeen, 2002.

[4] Grice, H. P. Logic and Conversation. In Cole P. and Morgan J. (Eds), Syntax and Semantics: Vol 3, Speech Acts. Academic Press, New York, pp.43-58, 1975.

[5] Grishman R, Kittredge R (Eds). Analyzing Language in Restricted Domains: Sublanguage Description and Processing. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1986.

[6] Keogh, E., Chu, S., Hart, D. and Pazzani, M. An Online Algorithm for Segmenting Time Series. In: Proceedings of IEEE International Conference on Data Mining, 2001, pp 289-296.

[7] Keogh, E., Hochheiser, H. and Shneiderman, B. An Augmented Visual Query Mechanism for Finding Patterns in Time Series Data. In: T Andreasen et al (Eds), Proceedings of the 5th International Conference on Flexible Query Answering Systems, pp. 240-250. October 27 - 29, 2002, Copenhagen, Denmark. Springer, LNAI 2522, 2002.

[8] Keogh, E., Lonardi, S and Chiu, W. Finding Surprising Patterns in a Time Series Database In Linear Time and Space. In: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (July 23 - 26, 2002). Edmonton, Alberta, Canada. pp 550-556, 2002.

[9] Levinson S. C. Pragmatics. Cambridge University Press, 1983.

[10] Lin, J. Keogh, E. Patel, P. & Lonardi, S. Finding motifs in time series. In: Proceedings of the 2nd Workshop on Temporal Data Mining, at the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (July 23-26, 2002). Edmonton, Alberta, Canada, 2002.

[11] Petre, M. Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. Communications of the ACM, 38(6), pp.33-44, 1995.

[12] Parikh, R. Vagueness and utility: The semantics of common nouns. Linguistics and Philosophy, 17:521-535, 1994.

[13] Plaisant, C., Mushlin, R., Snyder, A., Li, J., Heller, D., and Shneiderman, B. LifeLines: Using Visualization to Enhance Navigation and Analysis of Patient Records. Revised version in 1998 American Medical Informatic Association Annual Fall Symposium (Orlando, Nov. 9-11, 1998), p. 76-80, AMIA, Bethesda MD, 1998.

[14] Reiter, E. and Dale, R. Building Natural Language Generation Systems. Cambridge University Press, 2000.

[15] Reiter E. and Sripada, S. Human Variation and Lexical Choice. Computational Linguistics 28:545-553, 2002.

[16] Reiter E. and Sripada, S. Learning the Meaning and Usage of Time Phrases from a Parallel Text-Data Corpus. In: Proceedings of HLT-NAACL03 workshop on Learning Word Meaning from Non-Linguistic Data, pp78-85, 2003.

[17] Reiter E., Sripada, S. and Robertson, R. Acquiring Correct Knowledge for Natural Language Generation. Journal of Artificial Intelligence Research, 18: 491-516, 2003.

[18] Reiter E., Sripada, S. and Williams, S. Acquiring and Using Limited User Models in NLG. In Proceedings of ENLGW 2003, Budapest, Hungary, pp 87-94, 2003.

[19] Roddick, J. F., Hornsby, K. & Spiliopoulou, M. An Updated Bibliography of Temporal, Spatial and Spatio-Temporal Data Mining Research, pp 147-16**.** Lecture Notes in Artificial Intelligence, 2001.

[20] Scott, A., Clayton, J. and Gibson, E. A Practical Guide to Knowledge Acquisition. Addison-Wesley, 1991.

[21] Sripada, S., Reiter, E., Hunter J., and Yu, J. A Two-stage Model for Content Determination. In: Proceedings of ENLGW-2001, pp3-10, 2001.

[22] Sripada, S., Reiter, E., Hunter J., and Yu, J. Segmenting Time Series for Weather Forecasting, . In: Macintosh, A., Ellis, R. and Coenen, F. (ed) Applications and Innovations in Intelligent Systems X, Proceedings of ES2002, pp. 193-206, 2002.

[23] Sripada, S., Reiter, E., Hunter J. and Yu J. Summarizing Neonatal Time Series Data. Proceedings of the Research Note Sessions of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL'03), pp. 167-170, Budapest, Hungary, 2003.

[24] Sripada, S., Reiter, E., Hunter J. and Yu J. Exploiting a parallel TEXT-DATA corpus. In Proceedings of Corpus Linguistics 2003, p. 734-743. Lancaster, U.K. 2003.

[25] Sripada S., Reiter, E., Hunter, J. Yu J. and Davy, I. Modelling the Task of Summarising Time Series Data using KA Techniques. In: Macintosh, A., Moulton, M. and Preece, A. (ed) Applications and Innovations in Intelligent Systems IX, Proceedings of ES2001, pp183-196, 2001.

[26] van Wijk J. J. and van Selow E. R. Cluster and calendar-based visualization of time series data. In Proc. IEEE Symposium on Information Visualization, pages 4-9, Oct. 25-26, 1999.

[27] Weber M., Alexa M., and Müller W. Visualizing Time-Series on Spirals. In: Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01).

[28] Yager, R.R., "On Linguistic Summaries of Data," in Knowledge Discovery in Databases, Piatetsky-Shapiro, G. & Frawley, B. (eds.), Cambridge, MA.: MIT Press, 347-363, 1991.

[29] Yu, J., Hunter, J., Reiter E., and Sripada, S. **SUMTIME-TURBINE**: A Knowledge-Based System to Communicate Time Series Data in the Gas Turbine Domain. To appear in The 16th International Conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems, Loughborough, UK, June 23-26, 2003.