Automated Insights on Visualizations with Natural Language Generation

Richard Brath
Uncharted Software
Toronto, ON
rbrath@uncharted.software

Craig Hagerman
Uncharted Software
Toronto, ON
chagerman@uncharted.software

2021 25th International Conference Information Visualisation (IV) | 978-1-6654-3827-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IV53921.2021.00052

Abstract

Quantitative data, such as a 10k financial report, requires cognitive effort to scan the columns and rows and identify patterns and important takeaways, whether novice or subject matter expert. Visualizations can be used to summarize and reveal patterns. However, unless a visualization contains arrows or other callouts, it still requires cognitive effort to understand and rank the important conclusions to which a reader should pay attention. In this research, we aim to reduce the cognitive effort in understanding tabular data by combining charts with ranked natural language generated (NLG) bullet point statements that summarize the top takeaways. The contribution of this work is an NLG pipeline to computationally extract insights from tabular data and provide textual comments, which are then integrated with visualizations of the same data set.

Keywords—Natural Language Generation, Automated Commentary, Narrative Visualization.

I. Introduction

Many organizations create data-centric reports. Sometimes these reports are fully automated, such as reports from BI tools like Microsoft PowerBI. Often a dashboard, screen or page may include a large amount of data, as visualizations or tables, yet the viewer is unaware of the significance – which data values, which parts of the chart, contain insightful information? Sometimes human authors augment these reports to include separate commentary that indicates key findings or even rule-based text generation templates. This commentary is especially important when reports are created for external stakeholders who need help to guide them to most important data rather than blindly attempting to find meaningful insights.

Data-centric reports such as 10K financial reports (Figure 1), filed by public companies in the USA on a quarterly basis, are large grids of text that an analyst must scrutinize to determine which values are meaningful. To reduce the cognitive load associated with this task, one solution is to simply visualize the financial statement. Some approaches visualize the entire grid, and others visualize just the most important metrics using only the current and prior period. While the latter approach reduces the data, it is still not obvious which values deserve the most attention. Differences in bars are clearly visible (Figure 2), but which differences are more important? There is value if the visualization can direct attention to more meaningful observations while retaining the context of the other elements in the visualization to facilitate supplementary tasks such as comparison.

INTEL CORP (INTC)
INCOME STATEMENT

USD in millions	2016-12	2017-12	2018-12	2019-12	2020-12	TTM
Revenue	59387	62761	70848	71965	77867	77867
Cost of revenue	23196	23692	27111	29825	34255	34255
Gross profit	36191	39069	43737	42140	43612	43612
Operating expenses						
Research and development	12740	13098	13543	13362	13556	13556
Sales, General and administrative	8397	7474	6750	6150	6180	6180
Restructuring, merger and acquisition	1886	384	-72	393	198	198
Other operating expenses	-1592	-207	272	-193	-198	-198
Total operating expenses	21431	20749	20493	19712	19736	19736
Operating income	14760	18320	23244	22428	23876	23876
Interest Expense	733	646	468	489	629	629
Other income (expense)	-1091	2678	541	2119	1831	1831
Income before taxes	12936	20352	23317	24058	25078	25078
Provision for income taxes	2620	10751	2264	3010	4179	4179
Net income from continuing operations	10316	9601	21053	21048	20899	20899
Net income	10316	9601	21053	21048	20899	20899
Net income available to common shareholder	10316	9601	21053	21048	20899	20899
EBITDA	21459	29127	32870	35373	37946	37946
Earnings per share						
Basic	2.18	2.04	4.57	4.77	4.98	4.98
Diluted	2.12	1.99	4.48	4.71	4.94	4.94
Weighted average shares outstanding Basic						
Basic	4730	4701	4611	4417	4199	4199
Diluted	4875	4835	4701	4473	4232	4232

Fig. 1. Sample income statement for Intel. With many rows and columns, it is not obvious which data points are insightful.

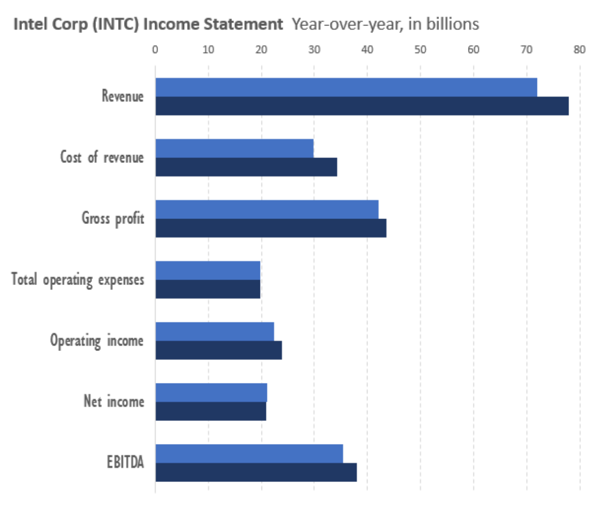

Fig. 2. Key metrics for last two periods. Even though a smaller dataset than Figure 1, it is not obvious which data points are insightful.

978-1-6654-3827-8/21/$31.00 ©2021 IEEE
DOI 10.1109/IV53921.2021.00052

Our system can identify and label the most relevant insights. It combines the most relevant observations as text integrated within a visualization for context. Our contribution is to: a) show an analysis of existing reports to identify potential insights to automate; b) introduce an NLG pipeline that computationally extracts these insights and generates human-readable sentences to accompany visualizations; and c) provide these textual sentences either visually linked or positioned spatially close to the corresponding visualization mark.

II. Prior Work

It has been claimed that text summaries can be more effective than graphs at conveying information [LFH05,Rei16,GLR17]. Others claim superiority of visualization (e.g. [Tuf83]).

Some studies have combined both, although with some degree of separation between the two. Latif et al [LLB18, LB18] provide textual insights in paragraphs (with sparklines) visually separated from visualizations, with interactions such as mousing over bold text in the paragraph to highlight the corresponding bar or portion thereof.

Chandler and Sweller show that learning and problem-solving tasks are improved when textual instructions are directly integrated into diagrams [CS91]; Narayaan and Hegarty [NH02] show improved performance when both text and images are positioned closely or when linkages are provided between the two. Larkin and Simon note reduced cognitive load by reducing cross referencing when related information is spatially proximate [LS87]. Matusiak and Brath discuss automated annotations [MB18] including textual annotations, but do not discuss automated generation of text, nor guidance on automated layout.

Other studies, such as Goffin et al. [GBW17] or Beck et al. [BSB17], focus on placing word-scale visualizations within text. Brath [Bra21] embeds visual markers within or under words to convey sizes (proportional formatting).

While there is not consensus on any particular approach, in general, research is converging on generated text placed in close proximity to visualizations.

Some commercial software such as Narrative Science and Arria [Che21], or research such as Chart-to-Text [OH20] generate textual summaries of charts. Such software does not call out what is important in the visualization. For example, for a bar chart they may create a text summary for every bar. The novel contribution of our work is that we generate comments which summarize the most important data in the visualization. A visualization curates and summarizes data, while the generated comments provide further focus on what is most important within that visual summary. We believe that this will reduce the cognitive load of interpreting a chart, especially for the non-domain specialist.

It is worth noting that finance is an area with active interest in NLG applications [GK18]. Our approach of using charts with NLG statements applied to 10K reports is a novel application likely to provide value to the finance community.

With respect to cognitive effort and charts, the effort also corresponds to user tasks (e.g. [BM13]). While all data may be present in the chart, cognitive effort may be required to estimate values, calculate ratios and so on.

III. Process

The authors have been involved in the implementation of automated commentary in three different real-world systems. This process typically includes:

1. *Analysis.* Review of existing commentary, categorization of different statement types that are made, and identification of metrics on which to focus.
2. *Insight Model*. Models are constructed to extract insights using techniques such as machine-learning classifiers or statistical thresholds per metric. If there are many insights, an additional step ranks insights to display.
3. *Natural Language Generation.* Given a set of flagged insights, appropriate natural language statements are generated.
4. *Display*. Given a visualization and accompanying text, there are many alternatives for displaying text. A few are discussed.

A. Analysis of Human-Authored Insights

In the analysis step, we review existing reports that are annotated with English-language paragraphs by experts with data commentary intended for their non-expert clients. For example, one page from a report shown in Figure 3 contains charts that correspond to data from an income statement along with commentary such as observations regarding specific data points and other notes including definitions.

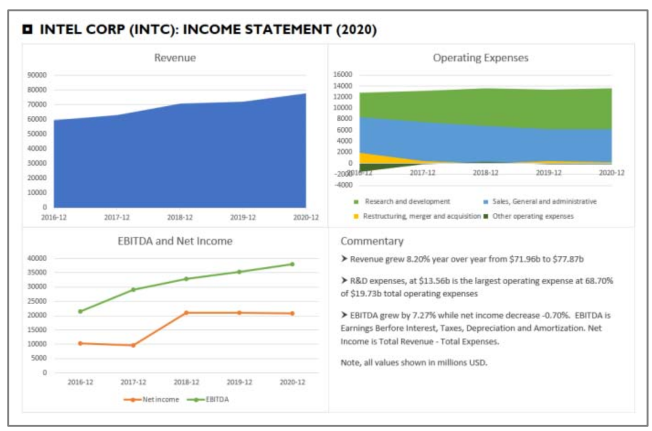

Fig. 3. Sample report page with added analyst commentary (bottom right). The text relates to the charts, but there is no visual connection. The viewer must visually search and mentally connect statements to visualization data points.

In one system, we reviewed more than 100 pages from sample reports. Within these were 189 visualizations including bar charts, pie charts, Euler diagrams, tables with spark charts, heatmaps, process diagrams, and grids of small multiples of charts. There were also 119 tables ranging in size from small with a few values, to large with multiple levels of hierarchy, detailed descriptions, and heterogenous metrics.

The majority of reports contained pages of tabular data, one or more charts that visualize some important aspect of the data and a comment box with four to six statements that call further attention to the important aspects of the data and charts. We used both manual and automated natural language processing (NLP) approaches to extract, categorize and analyze these statements.

There were 93 paragraphs with quantitative data values referring to nearby visualizations or tables. Within these paragraphs were over 300 unique comments. Of these, approximately 60 comments were non-data text (e.g., static explanations or informational content) while the remaining 240 sentences contained quantitative data values.

These quantitative sentences were analyzed and found to contain the following common patterns:

- *Comparative (110/240, 46%)*. These sentences compared two or more values, mostly indicative of a change over time. For example: "Revenue decreased from $665 million to $614 million, a decrease of 13%" or "Costs increased from 40% of revenue to 45% of revenue compared to 38% industry average."
- *Descriptive (54/240, 23%).* Descriptive statements usually focused on a single metric. For example: "The company acquired 3,456 new customers in the quarter" or "The percent of customers in urban areas grew 17.5%".
- *Ranking (38/240, 16%).* Ranking statements listed the top items within a categoric variable. For example: "Our top three products were the model X, model A and model T."
- *Savings (36/240 15%)*. There were a significant number of comments related to net savings. For example: "Total savings from the online program were $1.23 million."
- *Other*. There were a few other types of statements, including one counterfactual statement: "The outstanding amount of $12.34 million could have been transferred."

To ascertain whether the textual data corresponded to the visual data, we then reviewed these sentences and metrics in relation to nearby visualizations and table (i.e., on the same page or within +/- 3 blocks). We found:

- *Direct correspondence* (72%). These sentences had a direct correspondence to a represented value (e.g., value as a bar on a chart or cell in a table).
- *Derived data* (7%). These sentences derived data based on what was represented, such as a difference or ratio between two values shown, or the sum of multiple values shown.
- *Different data* (21%). These sentences show data not visible in any representation and cannot be otherwise derived from what is visible within the representations. For example, a comment might reflect a drill-down to the next level of data hierarchy, such as a chart showing quarterly data, but a sentence refers to monthly values ("December was particularly strong at $12.3 million," whereas the chart shows Q4 at $17.7 million).

In addition to the types of statements and their relation to visual representations, we noted some other observations.

- *Poor label cross-referencing*. Some visualizations use acronyms to label a bar or point (e.g., CAGR), whereas the text may use the full name (e.g., Compound Annual Growth Rate). This mismatch could increase the cognitive load on the user, requiring an added mapping to relate the text and visual representation.
- *Poor numerical cross-referencing*. Some visualizations and table cells indicate values with only a few significant digits (e.g., $1.23m) whereas the text might show the data without units (e.g., $1,230,000) or with more significance (e.g., $1,234,567). Both of these could increase the cognitive load, requiring the user to adjust the mapping between the two representations.
- *Conflated comparison*. Some sentences mix many metrics across clauses, thereby making inadvertent juxtapositions of unrelated values. This can result in the reader making comparisons between items that are not comparable. For example, a comment may indicate a recent significant change in a metric but conflate that with beginning and ending values to give the erroneous impression of a greater than actual change.
- *Awkward prose*. Some sentences appear to be the result of a template-filling approach to constructing comments. Using simplistic templates without refining the input or editing the results can lead to unnatural-sounding statements. This can be the result of there being no significant change in the metric statement (e.g., "0 out of 4 indicators were responsible for") or long phrases are inserted into a template.

The above analysis is a novel contribution. Next, we used this analysis to find insights and design statements for an NLG system for top insights.

Following the analysis of chart comments, we identified generalizable rules for how the analysts wrote their comments by conducting a linguistic analysis of the statements and patterns. In many cases different statements can be reduced to what is essentially a decision tree for choosing and combining constituents into a grammatically correct sentence.

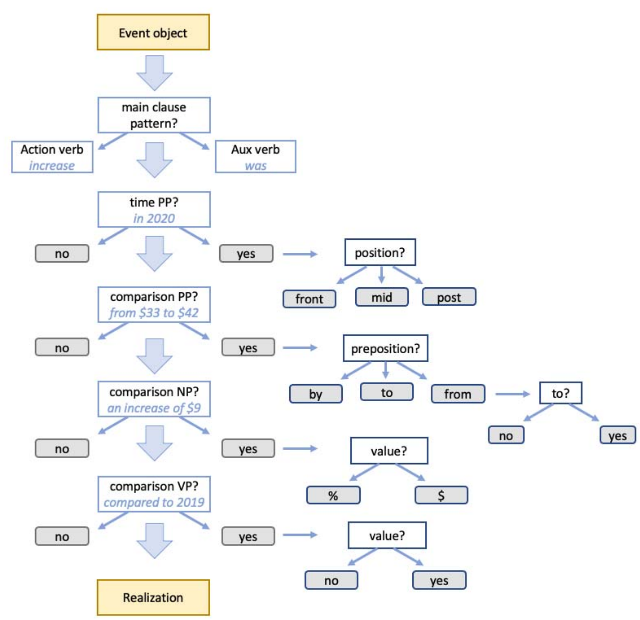

Fig. 4. Example of a decision tree for comparative statements.

For example, many statements in the *Comparative* category consist of a main clause (e.g., "Intel revenue increased", "Revenue increased for Intel", etc.) accompanied by one or more optional front or post modifier phrases. As show in Figure 4, this can be thought of as a template.

B. Natural Lanuage Generation Pipeline

Natural language generation systems typically use a pipeline [RD97] with the following stages:

1. Data analysis (extract facts, patterns and insights from data)
2. Data interpretation (weight and rank insights)
3. Document planning (organize data to use for insights)
4. Micro planning (package information into sentences)
5. Realization (produce grammatically correct surface text)

We follow this pipeline with one caveat. For our use case, the *document planning* phase is less important or relevant. Rather than creating long-form narrative text (for example, [TRS20] is a system that creates a sports news story from tabular basketball data), our goal is to generate a small number of bullet-point comments. Thus, in our system the sentence-ordering aspect of document planning is a simple heuristic that uses the top-ranked insights identified by the *data interpretation* phase.

There are multiple approaches to data analysis and interpretation steps. One approach is a statistical model. We can normalize the most frequently cited metrics and compute the thresholds which typically trigger the presence of a statement. For example, we may find that a change in revenue above one standard deviation (compared to trend) triggers a statement, whereas a change in expenses typically needs to be 2.5 standard deviations to trigger a statement.

Another approach uses supervised machine learning algorithms to surface insights. In this case, a classifier can be trained on historical data to identify classes of interest. A benefit of a machine-learning approach is that other variables are also considered. For example, comments regarding expenses are more frequent when earnings are negative (i.e., a loss), implying that losses can be corrected by adjusting expenses. However, there are several barriers involved, the primary being the availability of a sufficient volume of processed, labeled data. This would include a collection of tabular data and accompanying comment texts with the comments annotated to identify type of comment (comparison vs. rank, etc.), a weighting of the importance of each comment, and a label column to indicate the insight score.

As illustrated in Figure 3, there can be many metrics shown on a page and many insights could be triggered. Insights can be ranked, for example, by using normalized z-scores across the triggered insights in the statistical model or using the prediction score associated with the classification generated by the machine learning model. This data interpretation reduces the hundreds of insights down the most relevant insights.

In many instances, a large amount of training data may be unavailable, insufficient, or have quality issues. The 250 statements described in the previous section were insufficient for machine learning approaches. We therefore further describe a statistical heuristic approach to insight models.

The *data analysis* phase of the NLG pipeline runs several analytics on tabular data, computing row analytics (delta differences in periods, standard deviation, etc.), aggregations (comparing factors to identify relative contribution and ranking) and extracting values for top-level factors frequently used in descriptive comments. These analytics create *Event* objects for each factor using values extracted from the table. The *Event* objects are then enriched with additional computed statistics as required.

In the *data interpretation* phase, the enriched values for each *Event* are then combined to create an overall importance score for each category of insight. This allows all *Event* objects to be compared and ranked by relative importance.

The *document planning* phase uses the weights and rankings computed in the previous stage to identify roughly 2–8 *Event* objects about which to create auto-comments.

The earliest and simplest NLG systems simply involve fill-in-the-blank templates (e.g., "The probability of precipitation is 34%"). This may be sufficient for very narrow use cases, but is too limited for most NLG applications.

Generating even slightly longer, more complex or less regimented text that sounds like natural language is a more complex task. Almost all current state-of-the-art commercial NLG systems use templates [Rei18,GK18], but they are far more complex or scripted than the fill-in-the-blank weather example above. These systems mimic natural language by making stochastic, context-aware decisions about communicative goals, sentence length, word order, word choice, syntax selection and inflection.

We take such a programmatic, scripted template approach that allows for stochastic generation of natural language insights. These can produce a variety of grammatically correct statements to convey the insight contained in an *Event* object.

In the example of 10k reports, the insight model may flag the change in the metric *revenue* from the prior year as an insight. It is feasible to generate dozens of statements based on a simple comparison of two variables. For example, statements generated for a revenue change (using the Intel data from Figure 1), include:

- In 2020, revenue increased.
- Intel revenue increased 8.9% in 2020.
- In 2020, revenue increased from $18.3b to $19.9b.
- Revenue increased for Intel, in 2020, from $18.3b to $19.9b, an increase of $1.6b.
- Revenue increased for Intel, from $18.3b to $19.9b, an increase of $1.6b, compared to $18.3b in 2019.

Note how the amount of detail varies. The first statement accurately states that revenue increased, but contains no quantitative values. The last statement is excessive, repeating the 2019 value. Some of the middle statements provide variants that show base values (revenue of $18.3b and $19.9b), which may be useful as the level of accuracy at three digits is higher resolution than can be determined from simply viewing the visualization and estimating position. Some of the middle statements show derived values (a difference of $1.6b, a percent change of 8.9%), which helps the viewer avoid mental arithmetic thereby reducing the cognitive effort.

The constant in all of these statements is that "revenue increased" for Intel. This is the core of the *Event* object that the data analysis stage has flagged as important. Additional phrases and modifiers may be added to provide more detail and context, and alternate verbs may be used. This can result in thousands of unique variations in how the essence of a single *Event* object may be expressed. These choices are controlled by three considerations:

Event context and importance: The category of the *Event* object determines the choice of main clause and modifiers. For example, a *Comparative* main clause might be "Revenue increased…" whereas a *Ranking* main clause might be "The three main drivers of revenue were…". Moreover, the values and derived statistics of each *Event* object inform the choice of verbs and modifiers. For example, defining verb choices as ['increase', 'grow'] vs ['decrease', 'shrink'] or choosing to modify a verb with an adjective ('slightly', 'moderately', 'significantly') are controlled by whether the metric in question increased or decreased and by how much. The Event context also adjusts the probabilistic choices made in statement generation using heuristics to favor greater detail in some cases (e.g., more explanation might be favored when there is a significant negative change) and less in others (e.g., if revenue is important, but the delta change has been very small, it might be sufficient to simply state "In 2020, revenue remained relatively unchanged").

Narrative context: The context of other *Event* objects used for comment statements influences statement generation. While many NLG systems produce narrative paragraphs, our goal is to produce grammatically correct, natural-sounding bullet-point statements. Therefore, we are not concerned with linking phrases (e.g., "On the other hand…") or coordinating referring expression to create narrative flow. However, part of being 'natural-sounding' in bullet-point statements involves being context-aware. This is accomplished by a few simple heuristics such as "don't repeat yourself", "vary the message" and "vary sentence length". There are probabilities attached to each of the choices made in generating a statement. If, for example, a previous statement used the phrase "in 2020", the system will then automatically lower the probability that a time period prepositional phrase will be used in a subsequent statement. To vary message and sentence length, we similarly alter probabilities to make it unlikely that the system will generate sequential statements with the same surface pattern.

These heuristics allow for randomness from probability functions while trying to mimic usage patterns found in natural language human-generated statements.

Randomness: After taking into consideration framing of the Event itself and the context of previous statements, we try to mimic natural language by making stochastic choices for statement constituents.

Deterministic and stochastic choices made in consideration of Event context ensure that the content is accurate. Adjustments made in consideration of the narrative context help ensure that the statements sound less robotic. However, within these confines we favor randomness in the actual statement generated.

This approach is highly scalable. It does not use machine learning and thus does not require a large amount of training data. Each data table is handled on its own. However, the tabular data is expected to have sufficient breadth for internal comparisons to be made. For example, the table ought to contain columns representing multiple time periods, or a column of categories in order to have something comparable to comment on.

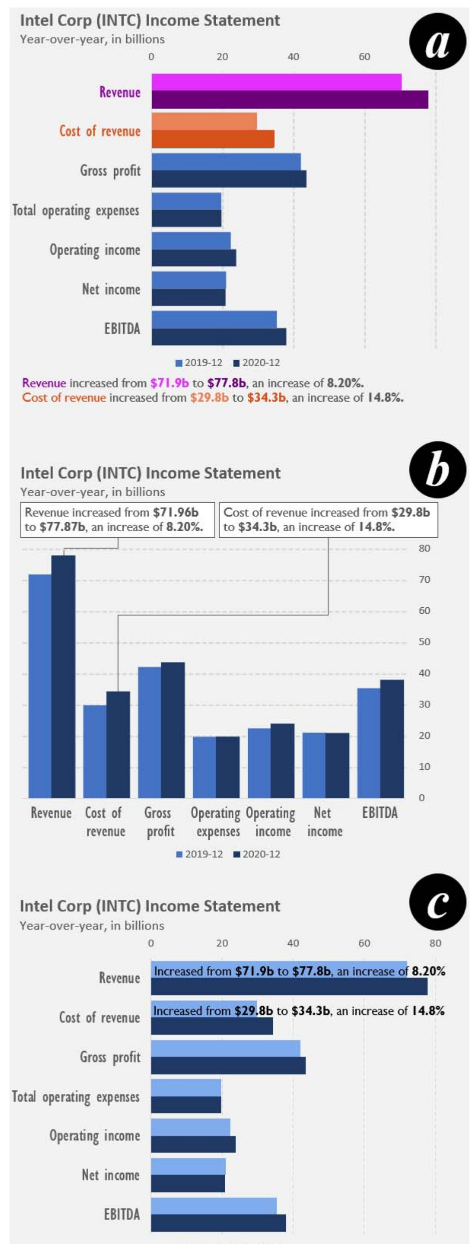

Fig. 5. Example insight sentences placed within visualizations. Sentences are visually connected to the corresponding visual element, guiding attention to relevant data point.

C. Visual Representation

Insight statements can be related to visualizations in different ways (Figure 5):

a) *SparkWords* [Bra21], wherein the text remains in separate prose outside of the visualization and is linked visually to the chart using a common visual attribute such as color. This approach coud be further enhanced with interactive linking [LLB18].

b) *Callouts* [MB18], place the text within the visualization linked by a visual connector, such as a leader line. The implementation in this example is straight-forward with all text in boxes above the plot area Comments are ordered to prevent leader lines from overlapping.

c) *Proportional encoding* [Bra21] places the text directly on the relevant marker. In this example, the text is superimposed on horizontal bars. The contrast between text and background needs to be sufficient for the text to remain legible. Unlike the other two layouts, there is no provision for text wrap, so line length is limited. This approached will not scale to other chart types such as pie charts or scatterplots. Finally, the metric name is already indicated in the bar label and does not need to be repeated in the insight text.

IV. Observations

A. Text Analysis, Insight Model, NLG

Identifying patterns in analyst comments and creating templates and decision trees to encapsulate those patterns was a largely manual, bottom-up process. The end result is a well-tuned NLG macro and micro planning that produces accurate, grammatically correct statements. A more efficient, scalable solution could be achieved if pattern identification and template creation were more fully automated.

Development of the micro planning stage of the NLG pipeline involved several iterations to identify issues and fine-time probabilities. Some of the early generated statements were repetitive or awkward (e.g., "In 2020, revenue increased in 2020…", "… increased by 3.2% to $77867 or 3.2%.") or un-grammatical (e.g., "… grew by 3.2% an increased of $5902"). Fixing these involved identifying the problematic branch or variable in the decision tree and editing or adding logic to address the issue. This was largely a manual task of printing out a large number of variations for a given Event, scanning the generated texts for problems and then adjusting the decision tree. This is a workable approach for the small number of statement patterns we produced but a more automated approach would be needed to scale to a wider number of categories.

B. User Feedback

Among the three different systems for which we developed automated insights, user feedback varied. For example, it was reported that "people didn't understand" SparkWords (Figure 5a), as "it was outside of their normal conventions."

Two systems deployed with callouts containing leader lines had mixed reactions. Labels on leader lines follow a convention with which people are familiar (for example, parts diagrams, furniture assembly, technical illustrations, infographics), but the extra leader lines added clutter to the display. If there are more than two or three insights, leader lines can potentially overlap and causing difficulty to visually trace and reduce the effectiveness of callouts. The text may also be visually distant, so users may not notice the text when focusing the bars. In the first system, there was a noted preference for fewer insights with shorter text instead of longer insights and/or many insights.

Overlaid micro-text has not been deployed in any system as of the time of writing this paper. Response to the technique in presentations is mixed; users find text written directly on bars to be unfamiliar, particularly when the text exceeds the length of the bar or needs to cross the end of the bar. We hypothesize that constraining the text to entirely within the bar or entirely outside and adjacent to the bar would be much more familiar to users (e.g., bar charts with numerical values showing) and increase acceptance.

In the third system, implementation of the insights has yet to occur, but user preferences are leaning toward callouts immediately at the top of the chart, possibly with no leader lines and simple color markers which then appear in beside both the text and the chart label (e.g. ■ ■).

V. Conclusion

We show an end-to-end method for visualizations with embedded textual insights, based on three real-world systems. We note several challenges with the analysis and creation of the NLG pipeline model. In particular, the promise of machine learning to automate the data analysis and interpretation would be ideal; without a large collection of prior statements, semi-supervised machine learning approaches could be explored in the future. Further, the document planning and generation stages of NLG would benefit from future automation to reduce fine-tuning effort. From a visualization perspective, more advanced layout approaches should be considered. For example, instead of leader lines with text boxes above the plot areas (Figure 5 b), there is sufficient space to create the text box within the plot area without overlapping the visual marks – which could be solved with a collision detection algorithm.

Note, images by the authors in this document are available as open source under CC BY-SA 4.0.

References

[BM13] Brehmer, Matthew, and Tamara Munzner. "A multi-level typology of abstract visualization tasks." *IEEE transactions on visualization and computer graphics* 19.12, 2013: 2376-2385.

[Bra21] Brath, Richard. *Visualizing with Text.* CRC Press. 2021.

[BSB17] Beck, Fabian, Hafiz Ammar Siddiqui, Alexandre Bergel, and Daniel Weiskopf. "Method Execution Reports: Generating text and visualization to describe program behavior." *2017 IEEE Working Conference on Software Visualization (VISSOFT)*. IEEE, 2017.

[Che21] Lynn Cherny. Dwaft2Text, GhostWeather.com, 2021 https://ghostweather.slides.com/lynncherny/dwarf2text

[CS91] Chandler, Paul and John Sweller, "Cognitive Load Theory and the Format of Instruction" *Cognition and Instruction*: 8(4) (1991), 293-332. http://ro.uow.edu.au/edupapers/128

[GBW17] Goffin, Pascal, Jeremy Boy, Wesley Willett, and Petra Isenberg. "An exploratory study of word scale graphics in data rich text documents."

IEEE transactions on visualization and computer graphics 23, no. 10 (2017): 2275 2287.

[GK18] Gatt, Albert, and Emiel Krahmer. "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation." *Journal of Artificial Intelligence Research* 61 (2018): 65-170.

[GLR17] Gkatzia D, Lemon O, Reiser V (2017). "Data-to-Text Generation Improves Decision-Making Under Uncertainty" https://dimitragkatzia.files.wordpress.com/2016/02/gkatzia-cim2017.pdf

[LB18] Latif, Shahid, and Fabian Beck. "Vis author profiles: Interactive descriptions of publication records combining text and visualization." *IEEE transactions on visualization and computer graphics* 25.1 (2018): 152-161.

[LFH05] Law A, Freer Y, Hunter J, Logie R, McIntosh N, Quinn J (2005). "A Comparison of Graphical and Textual Presentations of Time Series Data to Support Medical Decision Making in the Neonatal Intensive Care Unit" https://link.springer.com/article/10.1007/s10877-005-0879-3

[LLB18] Latif, Shahid, Diao Liu, and Fabian Beck. "Exploring Interactive Linking Between Text and Visualization." *EuroVis Short Papers*. (2018).

[LS87] Larkin, Jill and Herbert Simon. "Why a diagram is (sometimes) worth ten thousand words*." Cognitive science* 11.1 (1987): 65-100.

[MB18] Matusiak, M and Richard Brath. “Automated Annotations.” *Visualization for Communications (VisComm) Workshop at IEEE VisWeek* (2108).

[NH02] Narayanan, Hari and M. Hegarty. “Multimedia design for communication of dynamic information” *International Journal of Human Computer Interaction Studies*, 57, (2002). 279-315.

[OH20] Obeid, Jason, and Enamul Hoque. "Chart-to-Text: Generating Natural Language Descriptions for Charts by Adapting the Transformer Model." *arXiv preprint arXiv:2010.09142* (2020)

[Rei16] Reiter, Ehud. "Text or Graphics?". 2016-12-26. https://ehudreiter.com/2016/12/26/text-or-graphics/

[Rei18] Ehud Reiter. Lecture: “Data-To-Text: Generating Textual Summaries of Complex Data” https://www.youtube.com/watch?v=kFRw-wk5YOA

[RD97] Reiter, Ehud, and Robert Dale. "Building applied natural language generation systems." *Natural Language Engineering* 3.1 (1997): 57-87.

[TRS20] Thomson, Craig, Ehud Reiter, and Somayajulu Sripada. "SportSett: Basketball-A robust and maintainable dataset for Natural Language Generation." *IntelLanG: Intelligent Information Processing and Natural Language Generation*. 2020.

[Tuf83] Tufte, Edward. The Visual Display of Quantitative Information”, *Graphics Press*, 1983.