Standard Quality Criteria Derived from Current NLP Evaluations for Guiding Evaluation Design and Grounding Comparability and AI Compliance Assessments

Anya Belz, Simon Mille and **Craig Thomson**
DCU Natural Language Generation Research Group
ADAPT, Dublin City University, Dublin, Ireland
anya.belz@dcu.ie

Abstract

Research shows that two evaluation experiments reporting results for the same quality criterion *name* (e.g. Fluency) do not necessarily evaluate the same aspect of quality. Not knowing when two evaluations are comparable in this sense means we currently lack the ability to draw conclusions based on multiple independently conducted evaluations. It is hard to see how this issue can be fully addressed other than by the creation of a standard set of quality criterion names and definitions that the evaluations in use in NLP can be grounded in. Taking a descriptivist approach, the QCET *Q*uality *C*riteria for *E*valuation *T*axonomy derives a standard set of 114 quality criterion names and definitions from three surveys of a combined total of 933 evaluation experiments in NLP, and structures them into a reference taxonomy. We present QCET and its uses in (i) establishing comparability of existing evaluations, (ii) guiding the design of new evaluations, and (iii) assessing regulation compliance.

1 Introduction

Natural Language Processing (NLP) uses a wide variety of different names to refer to what is assessed in system evaluations, estimated to number over 200 for text-generating systems alone (Howcroft et al., 2020). Details of evaluations and definitions of what they assess are mostly patchy (Belz et al., 2023a,b; Ruan et al., 2024; Schmidtova et al., 2024), and it is often impossible to tell if the same aspect of quality was assessed in two evaluations, resulting in unclear comparability and low repeatability (Cohen et al., 2018). Consider the following evaluations which use the same name, Fluency, but each with a different definition:

1. Yu et al. (2020): "judging the question fluency."
2. Van de Cruys (2020): "grammatical and syntactically well-formed."
3. Pan et al. (2020): "follows the grammar and accords with the correct logic."

The first two evaluations assess single but different criteria (utterances can be grammatical but not very fluent), the third assesses two (an utterance can be grammatical yet illogical), so none of them assess the same aspect of quality. It is common for what was actually evaluated to be at odds with the name (or even definition) given in papers (Howcroft et al., 2020), e.g. the second evaluation above claims to be evaluating Fluency, but actually evaluates Grammaticality. In this situation, not only is it misleading to report, say, that a system improves Fluency, but any comparisons with other Fluency assessments are also unsafe.

As has been argued (van der Lee et al., 2019; Howcroft et al., 2020; Belz et al., 2020; van der Lee et al., 2021; Gehrmann et al., 2023), this is problematic, in particular for human evaluation which has always been considered the Litmus test of quality in NLP. It is hard to see how the current misalignments between (i) what is actually evaluated vs. what it is named, and (ii) what different researchers mean by the same quality criterion name, can be addressed other than by a standard reference set of criterion names and definitions that those actually in use can be grounded in. Deriving such a resource from reported evaluations has been our aim in the work reported here, resulting in the QCET *Q*uality *C*riteria for *E*valuation *T*axonomy, consisting of the following resources (browser at https://nlp-qcet.github.io/; other items at https://github.com/DCU-NLG/qcet_code):

1. An interactive taxonomy browser;
2. Extended description and usage guidance;
3. At-a-glance diagram of taxonomy (see also in Figure 3).

2 Standardising Quality Criteria

QCET is based on the notion of **quality criterion (QC)**, i.e. the specific aspect of system quality that is assessed in an evaluation, and the level at which we would expect two (well-designed) evaluations

Findings of the Association for Computational Linguistics: ACL 2025, pages 26685–26715

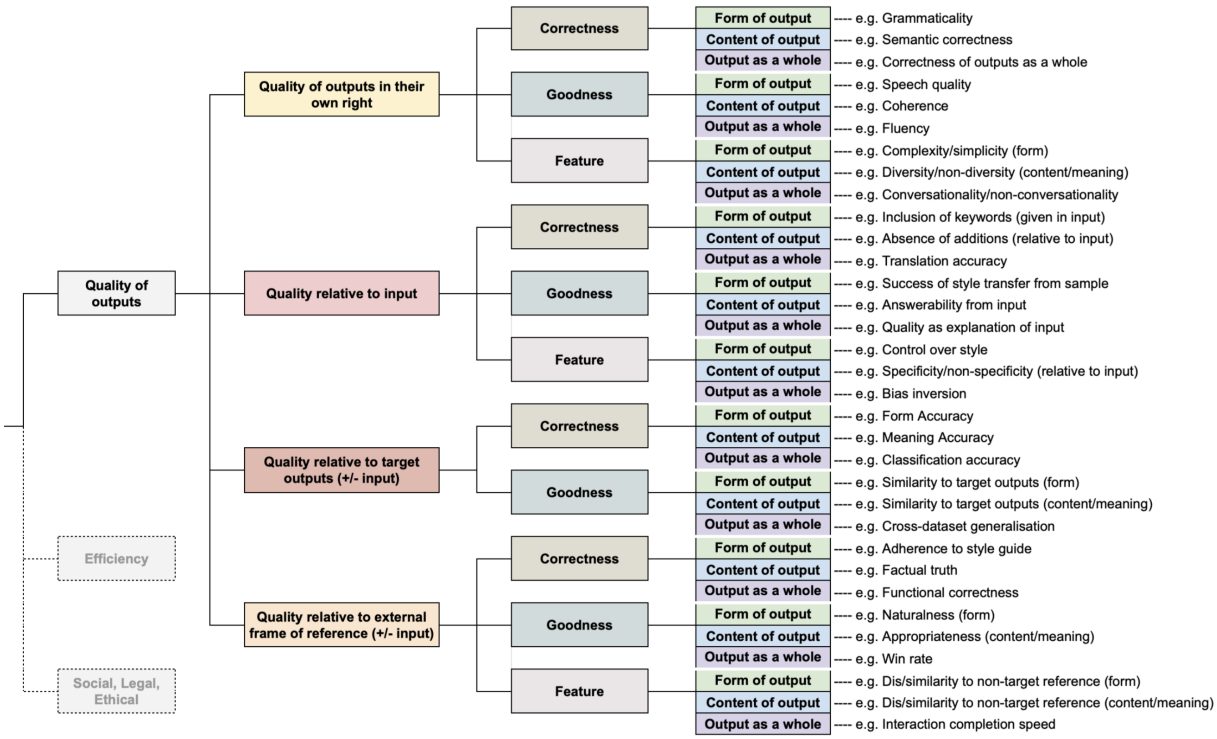


Figure 1: High-level view of QCET taxonomy structure with example leaf nodes (total leaf nodes at same branch).

(that assess the same QC) to support the same conclusions about which of two systems is better.

Consider these descriptions of two evaluations:

> "[...] eight users were given flight reservation tasks that required them to access the airline schedule [...]. The system logged the total completion time of a dialogue (Total Completion Time) [...]." (Qu and Green, 2002)

> "[we] compare [search] task completion times for two search algorithms [...] we had a number of paid participants describe a difficult [search] task which they had recently attempted. [...] Once 100 tasks were obtained in this manner, a separate group of 200 paid participants [acted] as users to attempt these tasks. [...] The resulting task times [...]" (Xu and Mease, 2009)

There are considerable differences between the two evaluation experiments: 1 system, 8 users in Q&G vs. 2 systems, 200 users in X&M; researcher-composed narrow tasks in Q&G vs. user-generated open tasks in X&M, etc. However, none of these differences change the answer to the question:

> *Q: What does this evaluation consider a better system to be?*

In the above evaluations, a better system is not one that is found to be better in an experiment with 8 users that access an airline schedule database to book a specific flight while talking to the system and take less time to do it. User numbers, interaction mode, location of information, etc., are not part of the aspect of system quality assessed, but of experiment design or system implementation. Instead, once we disregard such factors, we are left with the following answer to the above question:

> *A: A better system is one that enables the user to complete a given task more quickly.*

At this level we can see that both evaluations above assess the same aspect of quality. We would expect two well-designed evaluations that assess the same pair of systems in terms of this aspect of quality, using the same task and data, to come to the same conclusion about which is better.[1]

The above process is in fact how we derive standard QC definitions from three surveys of NLP evaluations. The first (Howcroft et al., 2020) proposed 71 standardised QCs for human evaluation of text-generating systems which we reviewed and revised for inclusion in QCET. We conducted two new surveys (Belz et al., to appear) of 60 papers each, published in 2022–2024 in ACL main proceedings. Here our scope was all of NLP, not just NLG as was the case in the 20Years survey. We identified altogether 455 individual occurrences of QCs, among which were 19 new ones which we added to the taxonomy. Based on the three surveys we ended up with 114 QCs for QCET.

For QC names, we follow the convention that we name QCs after the 'good end' of the correspond-

[1] Or more precisely, we would expect a majority of evaluations of this type to come to the same conclusion (because each is associated with some probability of a wrong result).

ing quality spectrum, or both ends where there isn't one (see Section 3), also aiming to closely reflect the definition. E.g. the QC assessed in the above evaluations is Task Completion Speed.

As a taxonomy of quality criteria, QCET is agnostic about, and therefore covers equally, automatic and human evaluations, all types of NLP systems and output modalities (speech, structured representations, labels, data), with the exception of certain individual terminal nodes (e.g. Spelling Accuracy is defined only for textual output).

3 Taxonomy Structure and QC Nodes

Figure 1 provides a high-level view of the structure of the QCET taxonomy. The second, third and fourth levels (after the root) correspond to the main three branching factors frame of reference (Section 3.2), type of quality (Section 3.3), and aspect of quality (Section 3.4). Below these in the actual taxonomy are levels of specific quality criteria (for space reasons not shown in the figure, except for single, abbreviated examples), mostly at the one terminal-node level shown in the figure. However, in a small number of cases there are more specific QCs sitting on two further levels. We start below with conventions for node IDs and content, before describing each of the three branching factors.

3.1 Node IDs and content

In the taxonomy, in addition to the QC names shown in Figure 1, each node also has (i) the QC definition, (ii) attestations in the literature, (iii) explanatory notes, and (iv) example questions to put to evaluators for the example evaluation modes absolute and relative (Belz et al., 2020). We have included the full list of QCs, with *i–iii* above in the appendix (Section C).

Each node has a unique ID which traces its path from the root via the first letters in the intervening node labels. E.g. Grammaticality (top right in Figure 1) has the ID QOC-f-1 (*Q*uality, *O*utput in its own right, *C*orrectness, *f* orm only, leaf node *1*).

In the remainder of the paper, we use node ID plus node name to refer to QCs. These can be quite long, so we use light grey highlight to indicate the span of the node ID/name, as in [QOC-f-1] Grammaticality. Note that all nodes represent a subclass and most also a QC. Some nodes only function as subclasses and not as QCs in their own right, e.g. the internal nodes in the **Features** subclass. IDs for such nodes are marked with an asterisk.

3.2 Frame-of-Reference Branches

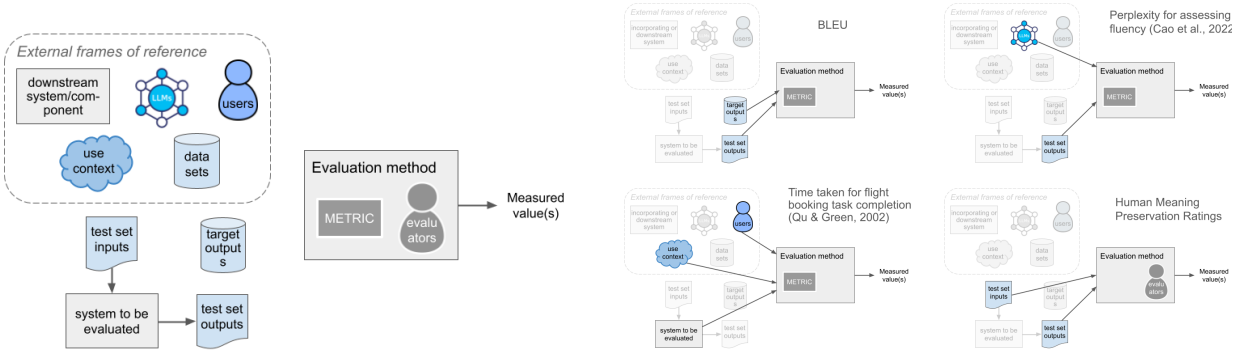
The root node corresponds to the single most general QC class, **Quality of outputs**. The next taxonomic level relates to the **Frame of Reference** that is used in an evaluation. For example, in order to assess Fluency with human raters, we don't need to look at anything other than a sample of system outputs; to assess Meaning Preservation e.g. in paraphrasing, we need to look only at a sample of system inputs and outputs; to assess BLEU, we only need a sample of outputs and corresponding target outputs; and in order to assess Task Completion Speed by users of the system, we need to get users to interact with the system and complete a specified task (here, users and use context form part of a system-external frame of reference that the QC is defined relative to).

Figure 2a provides a diagrammatic overview of these possible inputs to evaluation methods, divided into external frames of reference in the dashed box, and the system, and samples of inputs, outputs and target outputs at the bottom left. By looking at which of these a given evaluation method uses, we can tell which of the frame-of-reference branches of the taxonomy the QC being evaluated in the evaluation method belongs to:

1. **Output in its own right**: QCs that are defined relative to just the output (capture the quality of the output in its own right); an evaluation method of this type uses only the system outputs, as in e.g. Human Fluency Ratings.
2. **Output relative to input**: QCs that are defined relative to both and only the output and the input (capture the quality of the output relative to the input and nothing else); an evaluation method of this type uses only the system inputs and outputs, as in e.g. Meaning Preservation Ratings in Figure 2.
3. **Output relative to in-distribution target outputs (+/- input)**: QCs that are defined relative to target outputs sampled from the same distribution as the training data, and optionally also to the input (capture the quality of the output relative to given target outputs, optionally also taking the input into account). E.g. BLEU in Figure 2 uses (just) system outputs and target outputs.
4. **Output relative to a system-external frame of reference (FoR) (+/- input)**: QCs that are defined relative to a system-external FoR, and optionally also the input (capture the quality of the output relative to an external frame of reference, optionally also taking the output and/or input into account). An evaluation method of this type uses an explicit external FoR, as is the case when measuring time taken for task completion (here the FoR is user interaction with the system), see Figure 2.

3.3 Type-of-Quality Branches

The next taxonomic level captures the type of quality that a QC relates to (Belz et al., 2020). **Correctness** and **Goodness** QCs align with (i) what



(a) Possible inputs to evaluation methods.

(b) Four example evaluation methods and their inputs.

Figure 2: The different frames of reference, system-external and internal that are typically used, in different combinations, in evaluation methods (a); and example evaluation methods and the frames of reference they use (b).

we train systems to be good at, and/or (ii) commonsense notions of desirable properties in NLP systems. E.g. [QEG-w-6] User Satisfaction as Affected by Outputs and [QIC-w-1] Translation Accuracy: it is hard to conceive of circumstances under which we would want to consider a system better if it has *lower* user satisfaction or *lower* translation accuracy. Another way to look at it is that there is a preferred end to the scale independently of evaluation context.

This is not the case for the **Features** subclass where either end of the scale can be the preferred end depending on evaluation context. E.g. in the case of [QOF-w-3] Complexity/Non-complexity (outputs as a whole), sometimes better systems are those with simpler outputs (e.g. text simplification; Angrosh and Siddharthan, 2014), and sometimes those with less simple outputs (e.g. story generation; Purdy et al., 2018). Or, alternatively, neither end of the scale is better, and a better system is one that is better at generating texts at target levels of simplicity/complexity (graph summarisation at different reading levels; e.g. Moraes et al., 2016).

In more general terms:

1. **Correctness** QCs are based on (a finite set of) countable errors (e.g. spelling errors in [QOC-f-2] Spelling Accuracy). This makes it possible to state formally and precisely the conditions under which an output is of maximal quality, namely when it is free of errors.
2. For **Goodness** QCs, the conditions under which an output is of maximal quality cannot normally be stated generally. Goodness QCs are not primarily based on countable errors, typically taking multiple factors into account without distinguishing them. E.g. in [QOG-w-4] Humanlikeness, many different factors play into what makes an output more human-like.
3. **Features**: For a feature-type QC +X/-X, outputs are not generally better either if they are more +X, or if they are more -X. Depending on evaluation context, either more +X may be better, or more -X may be better, worse, or neither is associated with a notion of better/worse. E.g. in the case of [QEF-w-3] Effect on User Emotion, a better system produces outputs that affect the user's emotions (a) more, (b) less, or (c) as specified in the input, in terms of a given range of possible emotions.

3.4 Aspect-of-Quality Branches

The third taxonomic level captures which aspect of an output is being assessed:

1. **Form**: The form of outputs (in contrast to its content) is assessed, e.g. [QOC-f-1] Grammaticality: a sentence can be grammatical yet wrong or nonsensical in content.
2. **Content**: The content/meaning of outputs alone is assessed, e.g. [QIC-c-4] Coverage of Topics (given in input) – two sentences can have the same meaning, but differ in form.
3. **Outputs as a whole**: Outputs are assessed without distinguishing between form and content. E.g. [QOG-c-3] Coherence is a property of outputs as a whole, either form or meaning can detract from it.

Except for a few edge cases, we have found it straightforward to distinguish **Form** QCs from **Content** QCs. The former refers to how something is said, whereas the latter refers to what is said. Style, level of formality, choice between near-synonyms, syntax, word order, typography, etc. are part of **Form**. Sentiment, topic, factual truth, entailment, consistency, coherence, etc. are part of **Content**. However, when comparing the meaning of two representations (input, output, reference), the line between Form and Content has to be drawn explicitly. In the case of metrics, the algorithmic definition draws the line, while in human evaluation evaluators need to be instructed at what level of granularity to assess sameness of meaning.

4 Using QCET

Below we discuss three envisaged use cases for QCET, with illustrative examples. For fully worked

examples, please see the extended version of this paper at https://github.com/DCU-NLG/qcet_code.

Identifying the QC in an existing experiment and mapping it to the right QCET node: The first step is to locate all resources shared about a given experiment, then to identify (i) QC name, (ii) QC definition and (iii) the question and/or instructions put to evaluators. In many cases, *i–iii* are not completely aligned in which case *iii* takes priority as expressing what was actually evaluated.

A complicating factor is that in the effort of explaining one QC, researchers often introduce terms associated with other QCs, e.g. in the second Fluency definition at the start of the paper, Fluency is explained (only) in terms of grammaticality which introduces another QC. To identify the correct QC node, the taxonomy is perused top down, armed with the information in *i–iii* above, until the correct node is reached that corresponds to the specific individual quality criterion being assessed (possibly multiple QCs).

Selecting quality criteria for a new evaluation: A good starting point for evaluation in system development (in both academic and industry contexts) is the following question: *suppose we have two candidate systems, how do we know which one produces better outputs?* A useful answer is unlikely to be *the one with the higher BLEU score,* because that just means its outputs are more similar to the given sample of target outputs. Instead, the answer is likely involve multiple dimensions of quality.

Taking accommodation highlights generation in an accommodation booking context as an example (Kamath et al., 2024), a company developing such a system might conceivably decide that the better accommodation description summariser of two candidate systems would produce summaries that have *better quality of language*, *cover the main selling points*, contain *fewer mistakes or misrepresentations*, while also being closer *to some given target length range*. These broad dimensions each correspond to one or more specific QCET *branches*, but can each be covered by multiple QCET *nodes* within those branches (except for output length).

Kamath et al. went for [QOC-w-1] Grammaticality for the first aspect; a combination of [QOG-c-2] Informativeness and [QOG-w-5.1] Clarity for the second; [QIC-c-2] Absence of Additions (relative to input) and [QIC-c-3] Consistency with Input for the third; and controlled the fourth without assessing it as part of the reported evaluation.

AI regulation compliance assessment: AI regulation is under active development in several countries, and implemented in some, with interpretation and application in practice is underway. E.g. the EU's AI Act[2] came into force on 1 August 2024, with provisions coming into operation within 6 to 36 months. The Act requires high-impact general-purpose AI models that might pose systemic risk, such as GPT-4 to undergo thorough evaluations under transparency and accuracy requirements. What such evaluations will consist of, and what type of evaluations must be carried out, is currently not clear. The European Commission has requested standards to be developed by ISO/IEC that can be used to assess and enforce compliance with the provisions of the AI Act, and these include assessment for different aspects of overall system 'accuracy.'[3]

The standard used by oversight bodies could directly incorporate standardised QCET QCs in which case assessment boils down to applying the above steps for identifying QCs to the evaluations carried out by the developer, then comparing the resulting QCET QCs with those identified in the standard. Note this is also the approach that would be used by a *deployer* of technology developed by a third party to ensure they are compliant with regulations in using the technology. The developer would use the same standard to identify the QCs that need to be covered (possibly among others) by their evaluations, otherwise following the steps for selecting QCs above.

5 Conclusion

We have presented QCET, a taxonomy of 114 standard quality criteria complete with definitions and attestations from the literature, derived from a combined total of 933 existing NLP evaluation experiments. QCET is designed to support (i) assessments of the comparability of existing evaluations, (ii) guiding the design of new evaluations that are comparable by design, and (iii) assessment of regulation compliance. Rather than attempting to achieve complete coverage of QCs currently in use, QCET is designed to be extensible by adding new QCs to the appropriate branches of the taxonomy.

[2] https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence

[3] ISO/IEC AWI 23282 Artificial Intelligence: Evaluation methods for accurate natural language processing systems: https://www.iso.org/standard/87387.html

Ethical Considerations and Risks

The main components in the research presented are (i) systematic surveys of peer-reviewed NLP papers, and (ii) construction of a resource taxonomising the quality criteria assessed in evaluations in the papers. As such, the ethical implications and risks associated with the work is minimal. However, we have vetted the evaluation methods and quality criteria asses in them for appropriateness. We would have excluded any inappropriate evaluation methods and quality criteria, had we found any.

Limitations

A limitation of the work is the time window and the sampling method via which we obtained the set papers we analysed in the systematic survey described in Section A. It is likely that there are evaluation methods and quality criteria in papers outside of this sample that are not covered by our taxonomy. We have addressed this limitation by making the taxonomy arbitrarily expandable as and when new evaluation methods and quality criteria come to light that are not yet covered.

Acknowledgements

Mille's contribution was funded by the European Union under the Marie Skłodowska-Curie grant agreement No 101062572 (M-FleNS), and by the Irish Department of Tourism, Culture, Arts, Gaeltacht, Sport and Media via the eSTÓR project. Thomson's contribution was funded by the ADAPT SFI Centre for Digital Media Technology. Our work has also benefited more generally from being carried out within the research environment of the ADAPT SFI Centre, funded by Science Foundation Ireland through the SFI Research Centres Programme and co-funded under the European Regional Development Fund (ERDF) through Grant 13/RC/2106.

References

Fadi Abu Sheikha and Diana Inkpen. 2011. Generation of formal and informal sentences. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 187–193, Nancy, France. Association for Computational Linguistics.

C.P Afsal and K S Kuppusamy. 2024. Assessing the readability and coherence in gemini's triple draft generation: A multi-metric approach. In *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–7.

Sweta Agrawal. 2023. *Complexity Controlled Natural Language Generation*. Ph.D. thesis, University of Maryland, College Park.

Khalid Alnajjar and Mika Hämäläinen. 2018. A master-apprentice approach to automatic creation of culturally satirical movie titles. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 274–283, Tilburg University, The Netherlands. Association for Computational Linguistics.

Mandya Angrosh and Advaith Siddharthan. 2014. Text simplification using synchronous dependency grammars: Generalising automatically harvested rules. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 16–25, Philadelphia, Pennsylvania, U.S.A. Association for Computational Linguistics.

Somnath Basu Roy Chowdhury, Chao Zhao, and Snigdha Chaturvedi. 2022. Unsupervised extractive opinion summarization using sparse coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1209–1225, Dublin, Ireland. Association for Computational Linguistics.

Anya Belz, Simon Mille, and David M. Howcroft. 2020. Disentangling the properties of human evaluation methods: A classification system to support comparability, meta-evaluation and reproducibility testing. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 183–194, Dublin, Ireland. Association for Computational Linguistics.

Anya Belz, Simon Mille, and Craig Thomson. to appear. Standardising evaluation criterion names and definitions in nlp via systematic surveys. To appear.

Anya Belz, Craig Thomson, Ehud Reiter, Gavin Abercrombie, Jose M. Alonso-Moral, Mohammad Arvan, Anouck Braggaar, Mark Cieliebak, Elizabeth Clark, Kees van Deemter, Tanvi Dinkar, Ondřej Dušek, Steffen Eger, Qixiang Fang, Mingqi Gao, Albert Gatt, Dimitra Gkatzia, Javier González-Corbelle, Dirk Hovy, Manuela Hürlimann, Takumi Ito, John D. Kelleher, Filip Klubicka, Emiel Krahmer, Huiyuan Lai, Chris van der Lee, Yiru Li, Saad Mahamood, Margot Mieskes, Emiel van Miltenburg, Pablo Mosteiro, Malvina Nissim, Natalie Parde, Ondřej Plátek, Verena Rieser, Jie Ruan, Joel Tetreault, Antonio Toral, Xiaojun Wan, Leo Wanner, Lewis Watson, and Diyi Yang. 2023a. Missing information, unresponsive authors, experimental flaws: The impossibility of assessing the reproducibility of previous human evaluations in NLP. In *The Fourth Workshop on Insights from Negative Results in NLP*, pages 1–10, Dubrovnik, Croatia. Association for Computational Linguistics.

Anya Belz, Craig Thomson, Ehud Reiter, and Simon Mille. 2023b. Non-repeatable experiments and non-reproducible results: The reproducibility crisis in human evaluation in NLP. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3676–3687.

Amanda Cercas Curry, Dimitra Gkatzia, and Verena Rieser. 2015. Generating and evaluating landmark-based navigation instructions in virtual environments. In *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*, pages 90–94, Brighton, UK. Association for Computational Linguistics.

Alessandra Cervone, Chandra Khatri, Rahul Goel, Behnam Hedayatnia, Anu Venkatesh, Dilek Hakkani-Tur, and Raefer Gabriel. 2019. Natural language generation at scale: A case study for open domain question answering. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 453–462, Tokyo, Japan. Association for Computational Linguistics.

Chi Chen, Peng Li, Maosong Sun, and Yang Liu. 2023. Weakly supervised vision-and-language pre-training with relative representations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8341–8355, Toronto, Canada. Association for Computational Linguistics.

Wei-Fan Chen, Henning Wachsmuth, Khalid Al-Khatib, and Benno Stein. 2018. Learning to flip the bias of news headlines. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 79–88, Tilburg University, The Netherlands. Association for Computational Linguistics.

Yanran Chen, Hannes Gröner, Sina Zarrieß, and Steffen Eger. 2024. Evaluating diversity in automatic poetry generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19671–19692, Miami, Florida, USA. Association for Computational Linguistics.

Yiruo Cheng, Kelong Mao, and Zhicheng Dou. 2024. Interpreting conversational dense retrieval by rewriting-enhanced inversion of session embedding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2879–2893, Bangkok, Thailand. Association for Computational Linguistics.

Jenny Chim, Julia Ive, and Maria Liakata. 2024. Evaluating synthetic data generation from user generated text. *Computational Linguistics*, pages 1–44.

Francisco Javier Chiyah Garcia, David A. Robb, Xingkun Liu, Atanas Laskov, Pedro Patron, and Helen Hastie. 2018. Explainable autonomy: A study of explanation styles for building clear mental models. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 99–108, Tilburg University, The Netherlands. Association for Computational Linguistics.

James Clarke and Mirella Lapata. 2006. Models for sentence compression: A comparison across domains, training requirements and evaluation measures. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 377–384, Sydney, Australia. Association for Computational Linguistics.

Miruna Clinciu, Arash Eshghi, and Helen Hastie. 2021. A study of automatic metrics for the evaluation of natural language explanations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2376–2387.

K Bretonnel Cohen, Jingbo Xia, Pierre Zweigenbaum, Tiffany J Callahan, Orin Hargraves, Foster Goss, Nancy Ide, Aurélie Névéol, Cyril Grouin, and Lawrence E Hunter. 2018. Three dimensions of reproducibility in natural language processing. *LREC Int Conf Lang Resour Eval*, 2018:156–165.

Nathalie Colineau, Cecile Paris, and Keith Vander Linden. 2002. An evaluation of procedural instructional text. In *Proceedings of the International Natural Language Generation Conference*, pages 128–135, Harriman, New York, USA. Association for Computational Linguistics.

Liam Cripwell, Anya Belz, Claire Gardent, Albert Gatt, Claudia Borg, Marthese Borg, John Judge, Michela Lorandi, Anna Nikiforovskaya, William Soto-Martinez, et al. 2023. The 2023 webnlg shared task on low resource languages overview and evaluation results (webnlg 2023). In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*.

Hoa Trang Dang. 2005. Overview of duc 2005. In *Proceedings of the document understanding conference*, volume 2005, pages 1–12. Citeseer.

Courtney R Davis, Karen J Murphy, Rachel G Curtis, and Carol A Maher. 2020. A process evaluation examining the performance, adherence, and acceptability of a physical activity and diet artificial intelligence virtual health assistant. *International journal of environmental research and public health*, 17(23):9137.

Wietse de Vries, Martijn Wieling, and Malvina Nissim. 2022. Make the best of cross-lingual transfer: Evidence from POS tagging with over 100 languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7676–7685, Dublin, Ireland. Association for Computational Linguistics.

Giuseppe Di Fabbrizio, Amanda Stent, and Robert Gaizauskas. 2014. A hybrid approach to multi-document summarization of opinions in reviews. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 54–63, Philadelphia, Pennsylvania, U.S.A. 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.

Roy Eisenstadt and Michael Elhadad. 2020. Neural micro-planning for data to text generation produces more cohesive text. In *Proceedings of the Workshop on Discourse Theories for Text Planning*, pages 6–9, Dublin, Ireland. Association for Computational Linguistics.

Mireia Farrús Cabeceran, Marta Ruiz Costa-Jussà, José Bernardo Mariño Acebal, and José Adrián Rodríguez Fonollosa. 2010. Linguistic-based evaluation criteria to identify statistical machine translation errors. In *14th Annual Conference of the European Association for Machine Translation*, pages 167–173.

Jason Fong. 2024. *Controlling text-to-speech pronunciation using limited linguistic resources*. Ph.D. thesis, The University of Edinburgh.

James Forrest, Somayajulu Sripada, Wei Pang, and George Coghill. 2018. Towards making NLG a voice for interpretable machine learning. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 177–182, 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.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166.

Katy Gero, Chris Kedzie, Jonathan Reeve, and Lydia Chilton. 2019. Low level linguistic controls for style transfer and content preservation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 208–218, Tokyo, Japan. Association for Computational Linguistics.

Demian Ghalandari, Chris Hokamp, and Georgiana Ifrim. 2022. Efficient unsupervised sentence compression by fine-tuning transformers with reinforcement learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1267–1280, Dublin, Ireland. Association for Computational Linguistics.

Nadine Glas and Catherine Pelachaud. 2015. Topic transition strategies for an information-giving agent. In *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*, pages 146–155, Brighton, UK. Association for Computational Linguistics.

Javier González Corbelle, Alberto Bugarín-Diz, Jose Alonso-Moral, and Juan Taboada. 2022. Dealing with hallucination and omission in neural natural language generation: A use case on meteorology. In *Proceedings of the 15th International Conference on Natural Language Generation*, pages 121–130, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Nancy Green. 2006. Generation of biomedical arguments for lay readers. In *Proceedings of the Fourth International Natural Language Generation Conference*, pages 114–121, Sydney, Australia. Association for Computational Linguistics.

Vrindavan Harrison and Marilyn Walker. 2018. Neural generation of diverse questions using answer focus, contextual and linguistic features. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 296–306, Tilburg University, The Netherlands. Association for Computational Linguistics.

Jacob S Hershenhouse, Daniel Mokhtar, Michael B Eppler, Severin Rodler, Lorenzo Storino Ramacciotti, Conner Ganjavi, Brian Hom, Ryan J Davis, John Tran, Giorgio Ivan Russo, et al. 2024. Accuracy, readability, and understandability of large language models for prostate cancer information to the public. *Prostate Cancer and Prostatic Diseases*, pages 1–6.

David Howcroft, Anya Belz, Miruna Clinciu, Dimitra Gkatzia, Sadid Hasan, Saad Mahamood, Simon Mille, Sashank Santhanam, Emiel van Miltenburg, and Verena Rieser. 2020. Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions. In *Proceedings of the 13th International Natural Language Generation Conference*.

Yi Hu and Philipos C Loizou. 2007. Subjective comparison and evaluation of speech enhancement algorithms. *Speech communication*, 49(7-8):588–601.

Zhiqiang Hu, Roy Ka-Wei Lee, Charu C. Aggarwal, and Aston Zhang. 2022. Text style transfer: A review and experimental evaluation. *SIGKDD Explor. Newsl.*, 24(1):14–45.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. 2024. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1416–1428, Bangkok, Thailand. Association for Computational Linguistics.

Kevin Humphreys, Mike Calcagno, and David Weise. 2001. Reusing a statistical language model for generation. In *Proceedings of the ACL 2001 Eighth*

European Workshop on Natural Language Generation (EWNLG), Toulouse, France. Association for Computational Linguistics.

Joseph Marvin Imperial, Gail Forey, and Harish Tayyar Madabushi. 2024. Standardize: Aligning language models with expert-defined standards for content generation. *Preprint*, arXiv:2402.12593.

Svanhvít Lilja Ingólfsdóttir, Petur Ragnarsson, Haukur Jónsson, Haukur Simonarson, Vilhjalmur Thorsteinsson, and Vésteinn Snæbjarnarson. 2023. Byte-level grammatical error correction using synthetic and curated corpora. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7299–7316, Toronto, Canada. Association for Computational Linguistics.

Scott Jarvis, Yves Bestgen, and Steve Pepper. 2013. Maximizing classification accuracy in native language identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 111–118, Atlanta, Georgia. Association for Computational Linguistics.

Yiping Jin and Phu Le. 2016. Selecting domain-specific concepts for question generation with lightly-supervised methods. In *Proceedings of the 9th International Natural Language Generation conference*, pages 133–142, Edinburgh, UK. Association for Computational Linguistics.

Juraj Juraska, Kevin Bowden, and Marilyn Walker. 2019. ViGGO: A video game corpus for data-to-text generation in open-domain conversation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 164–172, Tokyo, Japan. Association for Computational Linguistics.

Srinivas Ramesh Kamath, Fahime Same, and Saad Mahamood. 2024. Generating hotel highlights from unstructured text using LLMs. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 280–288, Tokyo, Japan. Association for Computational Linguistics.

Zdeněk Kasner and Ondrej Dusek. 2022. Neural pipeline for zero-shot data-to-text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3914–3932, Dublin, Ireland. Association for Computational Linguistics.

Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-guided controlled generation of paraphrases. *Transactions of the Association for Computational Linguistics*, 8:330–345.

Puneet Kumar, Sarthak Malik, Balasubramanian Raman, and Xiaobai Li. 2024. Synthesizing sentiment-controlled feedback for multimodal text and image data. *Preprint*, arXiv:2402.07640.

Gaetan Latouche, Marc-André Carbonneau, and Benjamin Swanson. 2024. BinaryAlign: Word alignment as binary sequence labeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10277–10288, Bangkok, Thailand. Association for Computational Linguistics.

Taehyun Lee, Seokhee Hong, Jaewoo Ahn, Ilgee Hong, Hwaran Lee, Sangdoo Yun, Jamin Shin, and Gunhee Kim. 2024. Who wrote this code? watermarking for code generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4890–4911, Bangkok, Thailand. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

David Lindberg, Fred Popowich, John Nesbit, and Phil Winne. 2013. Generating natural language questions to support learning on-line. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 105–114, Sofia, Bulgaria. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2024. The lay person's guide to biomedicine: Orchestrating large language models. *Preprint*, arXiv:2402.13498.

Iain Macdonald and Advaith Siddharthan. 2016. Summarising news stories for children. In *Proceedings of the 9th International Natural Language Generation conference*, pages 1–10, Edinburgh, UK. Association for Computational Linguistics.

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, Nancy, France. Association for Computational Linguistics.

Simon Mille, Anya Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner. 2018. The first multilingual surface realisation shared task (SR'18): Overview and evaluation results. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 1–12, Melbourne, Australia. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

Remi Mir, Bjarke Felbo, Nick Obradovich, and Iyad Rahwan. 2019. Evaluating style transfer for text. *Preprint*, arXiv:1904.02295.

Raghav Mohan. 2021. Modifying visual explanations to improve the user understand-ability of explainable

artificial intelligence systems. *Eindhoven University of Technology; Master's thesis.*

Priscilla Moraes, Kathleen McCoy, and Sandra Carberry. 2016. Enabling text readability awareness during the micro planning phase of NLG applications. In *Proceedings of the 9th International Natural Language Generation conference*, pages 121–131, Edinburgh, UK. Association for Computational Linguistics.

Angelina Patience Mulia, Pirelli Rahelya Piri, and Cuk Tho. 2023. Usability analysis of text generation by chatgpt openai using system usability scale method. *Procedia Comput. Sci.*, 227(C):381–388.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2018. RankME: Reliable human ratings for natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 72–78, New Orleans, Louisiana. Association for Computational Linguistics.

Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, Su Wang, and Jason Baldridge. 2024. Docci: Descriptions of connected and contrasting images. *Preprint*, arXiv:2404.19753.

Juri Opitz and Anette Frank. 2022. Better Smatch = better parser? AMR evaluation is not so simple anymore. In *Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems*, pages 32–43, Online. Association for Computational Linguistics.

Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic graphs for generating deep questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1463–1475, Online. Association for Computational Linguistics.

Sheena Panthaplackel, Adrian Benton, and Mark Dredze. 2022. Updated headline generation: Creating updated summaries for evolving news stories. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6438–6461, Dublin, Ireland. Association for Computational Linguistics.

Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning. *arXiv preprint arXiv:1704.03084*.

Puyuan Peng, Po-Yao Huang, Shang-Wen Li, Abdelrahman Mohamed, and David Harwath. 2024. VoiceCraft: Zero-shot speech editing and text-to-speech in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12442–12462, Bangkok, Thailand. Association for Computational Linguistics.

Maja Popović. 2020. Informative manual evaluation of machine translation output. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5059–5069, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Adithya Pratapa, Antonios Anastasopoulos, Shruti Rijhwani, Aditi Chaudhary, David R. Mortensen, Graham Neubig, and Yulia Tsvetkov. 2021. Evaluating the morphosyntactic well-formedness of generated texts. *Preprint*, arXiv:2103.16590.

Adrien Pupier, Maximin Coavoux, Jérôme Goulian, and Benjamin Lecouteux. 2024. Growing trees on sounds: Assessing strategies for end-to-end dependency parsing of speech. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 225–233, Bangkok, Thailand. Association for Computational Linguistics.

Christopher Purdy, Xinyu Wang, Larry He, and Mark Riedl. 2018. Predicting generated story quality with quantitative measures. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 14(1):95–101.

Yan Qu and Nancy Green. 2002. A constraint-based approach for cooperative information-seeking dialogue. In *Proceedings of the International Natural Language Generation Conference*, pages 136–143, Harriman, New York, USA. Association for Computational Linguistics.

Elaheh Rafatbakhsh, Alireza Ahmadi, Amirsaeid Moloodi, and Saeed Mehrpour. 2021. Development and validation of an automatic item generation system for english idioms. *Educational Measurement: Issues and Practice*, 40(2):49–59.

Sathish Reddy, Dinesh Raghu, Mitesh M. Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using a RNN based question generation model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 376–385, Valencia, Spain. Association for Computational Linguistics.

Yarik Menchaca Resendiz and Roman Klinger. 2025. Mopo: Multi-objective prompt optimization for affective text generation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5588–5606.

Verena Rieser, Simon Keizer, Oliver Lemon, and Xingkun Liu. 2011. Adaptive information presentation for spoken dialogue systems: Evaluation with real users. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 102–109, Nancy, France. Association for Computational Linguistics.

Jie Ruan, Wenqing Wang, and Xiaojun Wan. 2024. Defining and detecting vulnerability in human evaluation guidelines: A preliminary study towards reliable NLG evaluation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7965–7989, Mexico City, Mexico. Association for Computational Linguistics.

Horacio Saggion, Sanja Štajner, Stefan Bott, Simon Mille, Luz Rello, and Biljana Drndarevic. 2015. Making it simplext: Implementation and evaluation of a text simplification system for spanish. *ACM Transactions on Accessible Computing (TACCESS)*, 6(4):1–36.

Sashank Santhanam and Samira Shaikh. 2019. Towards best experiment design for evaluating dialogue system output. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 88–94, Tokyo, Japan. Association for Computational Linguistics.

Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic generation of programming exercises and code explanations using large language models. In *Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 1*, ICER '22, page 27–43, New York, NY, USA. Association for Computing Machinery.

Yuichi Sasazawa, Terufumi Morishita, Hiroaki Ozaki, Osamu Imaichi, and Yasuhiro Sogawa. 2023. Controlling keywords and their positions in text generation. *Preprint*, arXiv:2304.09516.

Patricia Schmidtova, Saad Mahamood, Simone Balloccu, Ondrej Dusek, Albert Gatt, Dimitra Gkatzia, David M. Howcroft, Ondrej Platek, and Adarsa Sivaprasad. 2024. Automatic metrics in natural language generation: A survey of current evaluation practices. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 557–583, Tokyo, Japan. Association for Computational Linguistics.

Siqi Shen, Veronica Perez-Rosas, Charles Welch, Soujanya Poria, and Rada Mihalcea. 2022. Knowledge enhanced reflection generation for counseling dialogues. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3096–3107, Dublin, Ireland. Association for Computational Linguistics.

Sotaro Shibayama, Deyun Yin, and Kuniko Matsumoto. 2021. Measuring novelty in science with word embedding. *PLOS ONE*, 16(7):1–16.

Chang Shu, Yusen Zhang, Xiangyu Dong, Peng Shi, Tao Yu, and Rui Zhang. 2021. Logic-consistency text generation from semantic parses. *Preprint*, arXiv:2108.00577.

Somayajulu Sripada, Neil Burnett, Ross Turner, John Mastin, and Dave Evans. 2014. A case study: NLG meeting weather industry demand for quality and quantity of textual weather forecasts. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 1–5, Philadelphia, Pennsylvania, U.S.A. Association for Computational Linguistics.

Craig Thomson and Ehud Reiter. 2020. A gold standard methodology for evaluating accuracy in data-to-text systems. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 158–168, Dublin, Ireland. Association for Computational Linguistics.

Tim Van de Cruys. 2020. Automatic poetry generation from prosaic text. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2471–2480.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language*, 67:101151.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Chris van der Lee, Emiel Krahmer, and Sander Wubben. 2017. PASS: A Dutch data-to-text system for soccer, targeted towards specific audiences. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 95–104, Santiago de Compostela, Spain. Association for Computational Linguistics.

Ielka van der Sluis and Chris Mellish. 2009. Towards empirical evaluation of affective tactical NLG. In *Proceedings of the 12th European Workshop on Natural Language Generation (ENLG 2009)*, pages 146–153, Athens, Greece. Association for Computational Linguistics.

Ellen M Voorhees, Dawn M Tice, et al. 1999. The trec-8 question answering track evaluation. In *TREC*, volume 1999, page 82.

Weichuan Wang, Zhaoyi Li, Defu Lian, Chen Ma, Linqi Song, and Ying Wei. 2024a. Mitigating the language mismatch and repetition issues in llm-based machine translation via model editing. *arXiv preprint arXiv:2410.07054*.

Yikun Wang, Rui Zheng, Liang Ding, Qi Zhang, Dahua Lin, and Dacheng Tao. 2024b. Uncertainty aware learning for language model alignment. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11087–11099, Bangkok, Thailand. Association for Computational Linguistics.

Nick Webb, David Benyon, Preben Hansen, and Oil Mival. 2010. Evaluating human-machine conversation for appropriateness. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Chao-Chung Wu, Ruihua Song, Tetsuya Sakai, Wen-Feng Cheng, Xing Xie, and Shou-De Lin. 2019. Evaluating image-inspired poetry generation. In *CCF international conference on natural language processing and chinese computing*, pages 539–551. Springer.

Zhangyi Wu, Tim Draws, Federico Cau, Francesco Barile, Alisa Rieger, and Nava Tintarev. 2023. Explaining search result stances to opinionated people. In *World Conference on Explainable Artificial Intelligence*, pages 573–596. Springer.

Stanley Xie, Ruchir Rastogi, and Max Chang. 2017. Deep poetry: Word-level and character-level language models for shakespearean sonnet generation. *Natural Lang. Process. Deep Learn*.

Jun Xu, Zeyang Lei, Haifeng Wang, Zheng-Yu Niu, Hua Wu, and Wanxiang Che. 2021. Enhancing dialog coherence with event graph grounded content planning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3941–3947.

Ya Xu and David Mease. 2009. Evaluating web search using task completion time. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 676–677.

Kevin Yang and Dan Klein. 2021. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Aditya Yedetore, Tal Linzen, Robert Frank, and R Thomas McCoy. 2023. How poor is the stimulus? evaluating hierarchical generalization in neural networks trained on child-directed speech. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Pengcheng Yin, John Wieting, Avirup Sil, and Graham Neubig. 2022. On the ingredients of an effective zero-shot semantic parser. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1455–1474, Dublin, Ireland. Association for Computational Linguistics.

Qian Yu, Lidong Bing, Qiong Zhang, Wai Lam, and Luo Si. 2020. Review-based question generation with adaptive instance transfer and augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 280–290, Online. Association for Computational Linguistics.

Xiang Yue, Ziyu Yao, and Huan Sun. 2022. Synthetic question value estimation for domain adaptation of question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1340–1351, Dublin, Ireland. Association for Computational Linguistics.

Zejun Zhang, Zhenchang Xing, Xin Xia, Xiwei Xu, and Liming Zhu. 2022. Making python code idiomatic by automatic refactoring non-idiomatic python code with pythonic idioms. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2022, page 696–708, New York, NY, USA. Association for Computing Machinery.

Zhihao Zhang, Tomas Goldsack, Carolina Scarton, and Chenghua Lin. 2024. Atlas: Improving lay summarisation with attribute-based control. *Preprint*, arXiv:2406.05625.

Jonathan Zheng, Alan Ritter, and Wei Xu. 2024. NEOBENCH: Evaluating robustness of large language models with neologisms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13885–13906, Bangkok, Thailand. Association for Computational Linguistics.

Kai Zheng, VG Vinod Vydiswaran, Yang Liu, Yue Wang, Amber Stubbs, Özlem Uzuner, Anupama E Gururaj, Samuel Bayer, John Aberdeen, Anna Rumshisky, et al. 2015. Ease of adoption of clinical natural language processing software: an evaluation of five systems. *Journal of biomedical informatics*, 58:S189–S196.

Tianqi Zhong, Zhaoyi Li, Quan Wang, Linqi Song, Ying Wei, Defu Lian, and Zhendong Mao. 2024. Benchmarking and improving compositional generalization of multi-aspect controllable text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6486–6517, Bangkok, Thailand. Association for Computational Linguistics.

Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. Don't make your llm an evaluation benchmark cheater. *Preprint*, arXiv:2311.01964.

A Derivation and Expansion of Quality Criteria

We had three sources of quality criteria: (i) an existing survey of human evaluations of language-generating systems (Howcroft et al., 2020) which we reviewed and updated, (ii) a round of expanding the taxonomy based on our knowledge of metric and human evaluation methods, and (iii) two new systematic surveys of NLP papers.

After i and ii, we searched for exact attestations in the literature for each QC, i.e. where exactly the QC had been assessed in an evaluation method, retaining only those for which we found at least one attestation.

We then conducted the two surveys described next, where we found just 19 unique new QCs in the 455 evaluation experiments analysed; these were then added to the QCET taxonomy. This serves as an indication that QCET has reasonably good coverage.

New surveys

To better cover automatic metrics and non-text generating NLP systems in QCET, we carried out two additional surveys for this paper of 2×60 randomly selected NLP papers published in ACL main proceedings over the last three years (2022–2024), collecting provided information for the following properties (in addition to bibliographic information) for every evaluation method found in the paper:

1. Metric vs. human evaluation, or 'none found'
2. Location in paper or elsewhere of evidence
3. Verbatim QC name in paper, or 'none given'
4. Verbatim QC definition (or closest thing to it) in paper
5. QCET node a QC belongs to, or 'not found'

If QCET did not already cover the QC, we created a new QC node for it. Nearly all new QC names and definitions were derived from evaluation methods using automatic metrics. In this way, we increased the total number of QC terminal nodes in QCET to 114.

B Diagrammatic View of QCET Taxonomy

Figure 3 shows the whole of the QCET taxonomy in diagrammatic overview, displaying node IDs and QC names only.

C List of Quality Criteria with Definitions and Notes

C.1 QCs that define quality in terms of outputs only

Figure 3 shows a simplified view of the **Quality of outputs in their own right** branch of the QCET Taxonomy (only node IDs and Names are shown for each node). The top three levels were explained in the paper; below we list each QC and definition, with some additional explanatory notes in some cases, grouped according to subtrees (see Figure 3).

C.1.1 Correctness

Form

[QOC-f-1] Grammaticality: A better system produces texts with fewer grammatical errors.

Example: Humphreys et al. (2001) informally evaluate 200 outputs manually for Grammaticality reporting 4% of outputs with grammatical errors for their combined parser/generator.

Notes: [QOC-f-1] is Grammaticality as judged by native speakers, i.e. it's a human-assessable only QC. Cf. [QIC-f-1] Matching Syntactic Structure (given in input), and [QEC-f-2] Adherence to Syntactic Rules which can be assessed either with metrics or humans.

[QOC-f-2] Spelling Accuracy: A better system produces texts with fewer spelling errors.

Example: Farrús Cabeceran et al. (2010) manually compare Google Translate and the language-pair specific N-II system, e.g. reporting 169 orthographic errors in 711 Spanish-Catalan translations for Google compared to 62 for N-II.

Notes: –

[QOC-f-3] Pronunciation Accuracy: A better system produces speech with fewer pronunciation errors.

Example: Fong (2024) manually compare US and Scottish speech code inputs for their text-to-speech system, finding e.g. that the former lead to mispronunciations in 15% of words, and the latter in 24% (Table 7.1).

Notes: –

Content

[QOC-c-1] Semantic Correctness: A better system produces outputs with fewer semantic errors.

Example: Lindberg et al. (2013) ask an education specialist to assess questions generated by their system in terms of semantic validity, finding e.g. that 66% of them "made sense."

Notes: Semantic correctness is about the output being logically sound, but also obeying commonsense knowledge about the real world and events occurring in the right temporal order.

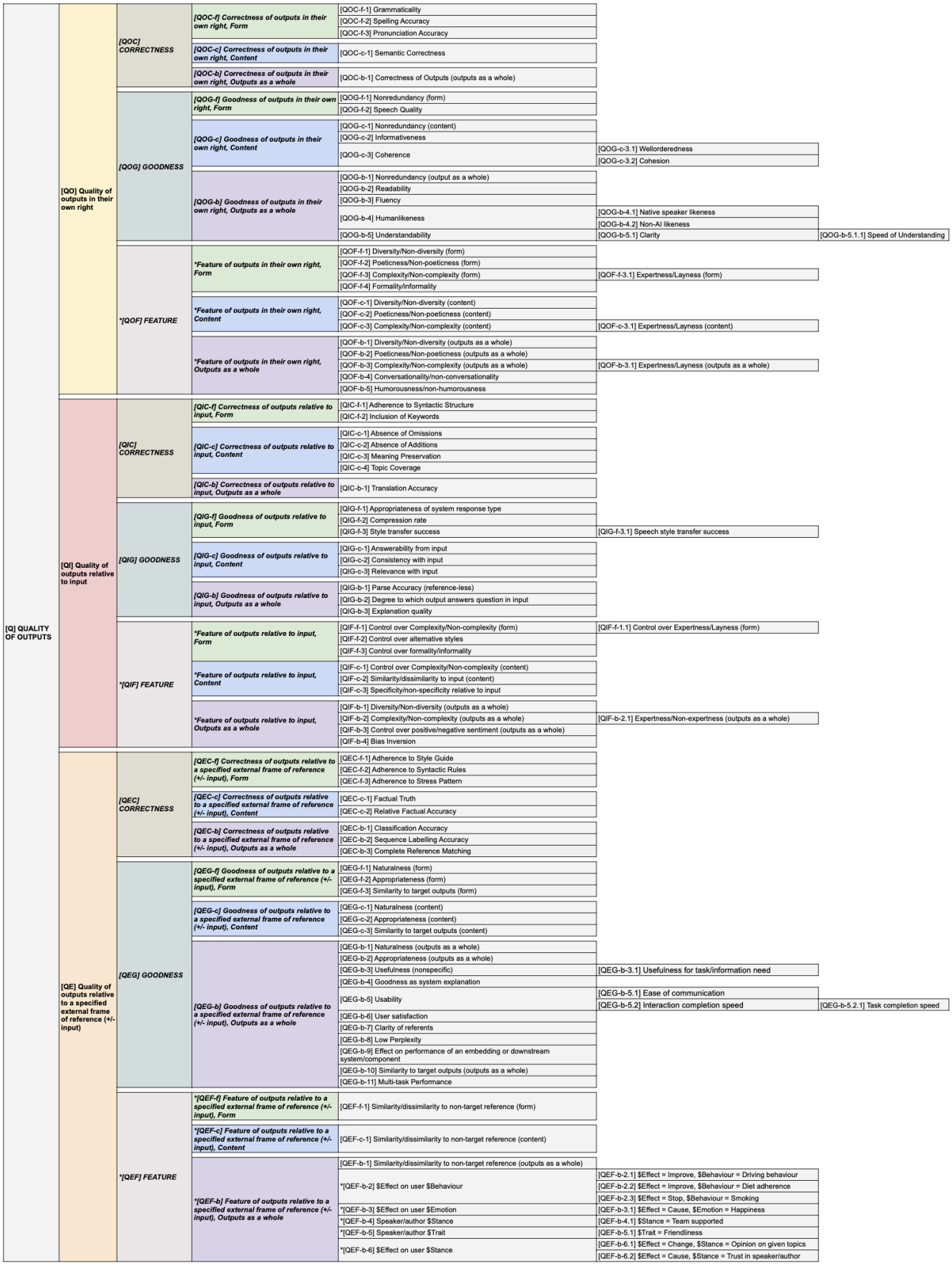


Figure 3: Diagrammatic overview of the QCET taxonomy, showing node IDs and quality criterion (QCs) names only. * = node is a class of QCs, but not a QC in its own right.

Outputs as a whole

[QOC-w-1] Correctness of Outputs (outputs as a whole): A better system produces outputs with fewer overall errors.

Example: Rafatbakhsh et al. (2021) manually determine the proportion of acceptable multiple-choice language learning items generated by their system; the assessment involved checking that question and answers matched, and that there were the right number of answers.

Notes: Correctness of Outputs is often about whether the output is of the correct type, e.g. in a question generator, *is the output a question*, or if an LLM is prompted for class labels, *is the output a class label*. The notion of correctness often derives simply from the system task (as in both these examples).

C.1.2 Goodness

Form

[QOG-f-1] Nonredundancy (form): A better system produces outputs with less redundancy in their form.

Example: Wang et al. (2024a) assess outputs from LLM-based MT with a 'repetition ratio' metric defined as the percentage of translations that have repetitions of substrings at the end.

Notes: Examples of redundancies of form include unnecessary repetitions of word or character strings, and extraneous brackets in code or mathematical expressions.

[QOG-f-2] Speech Quality: A better system produces speech that is of better quality.

Example: Hu and Loizou (2007) ask evaluators to rate speech enhancement outputs in terms of the level of distortion of the speech signal on a 5-point scale ranging from "very natural, no degradation" to "very unnatural, very degraded."

Notes: –

Content

[QOG-c-1] Nonredundancy (content/meaning): A better system produces outputs with less redundancy in their content/meaning.

Example: Di Fabbrizio et al. (2014) evaluate the 'compactness' of review summaries on a 5-point scale via human evaluation on Amazon Mechanical Turk, where a compact summary is one that "does not repeat information."

Notes: Examples of redundancies of content/meaning include the same meaning being expressed more than once in different ways, use of full names when pronouns would suffice, and overly explanatory details (e.g. *They closed the door behind them using the doorhandle which was affixed to the door.*).

[QOG-c-2] Informativeness: A better system produces outputs that are more informative.

Example: Green (2006) evaluates the outputs of a discourse generator that produces lay-oriented genetic counseling texts, by asking students to edit the outputs to ensure their contents provide the right level of information. The amount of editing done is viewed as indicative of informativeness.

Notes: Informativeness can be about information density and/or information sufficiency. E.g. a text that conveys a lot of information concisely, without using more words than necessary, is information dense. A text that provides the right level of information for a given scenario provides sufficient information.

[QOG-c-3] Coherence: A better system produces outputs whose contents/meaning hang(s) together better.

Example: In the 2005 DUC shared task on summarisation (Dang, 2005), outputs are assessed in terms of the degree to which they meet the following statement: "The summary should be well-structured and well-organized. The summary should not just be a heap of related information, but should build from sentence to sentence to a coherent body of information about a topic."

Notes: –

[QOG-c-3.1] Wellorderedness: A better system produces outputs whose content/meaning is ordered better.

Example: In their content-planning enhanced approach to dialogue, Xu et al. (2021) evaluate content ordering by first manually segmenting a dialogue by topic, and then rating each segment 1 if the ordering is appropriate, otherwise 0, and reporting the average.

Notes: [QOG-c-3.1] Wellorderedness captures whether content is ordered in a way that makes sense, e.g. that events are presented in the right order, that related points are made in the right place(s), etc. It is a necessary, but not sufficient, condition for [QOG-c-3] Coherence.

[QOG-c-3.2] Cohesiveness: A better system pro-

duces outputs whose content/meaning elements are linked better.

Example: Eisenstadt and Elhadad (2020) compare WebNLG outputs generated by (i) T5, and (ii) T5 plus a neural micro-planner, with the human target output texts in terms of the counts of cohesive devices they contain, finding that texts generated with a micro-planner are much more similar in this respect to human-written texts.

Notes: [QOG-c-3.2] Cohesiveness is all about linkage between content elements at every level of granularity, typically a property of text, it involves discourse connectives, anaphoric referencing, lexical congruity, etc. It is a necessary, but not sufficient, condition for [QOG-c-3] Coherence.

[QOG-c-4] Internal Consistency of Outputs: A better system produces outputs that are more consistent in their content/meaning.

Example: In their work on automatic alignment of images and text snippets from different sources, Chen et al. (2023) evaluate output image-text pairs with the CLIPScore text-to-image similarity metric that computes the cosine similarity between the embeddings of the image and the text produced by a multimodal model.

Outputs as a whole

[QOG-w-1] Nonredundancy (output as a whole): A better system produces outputs with less overall redundancy.

Example: In the 2005 DUC shared task on summarisation (Dang, 2005), outputs are assessed in terms of the degree to which they meet the following statement: "There should be no unnecessary repetition in the summary. Unnecessary repetition might take the form of whole sentences that are repeated, or repeated facts, or the repeated use of a noun or noun phrase (e.g., 'Bill Clinton') when a pronoun ('he') would suffice."

Notes: Nonredundancy of outputs as a whole captures redundancies of form and content both. The explanation from DUC 2005 quoted in the attestation for this node provides a good explanation of this QC.

[QOG-w-2] Readability: A better system produces outputs that are more readable.

Example: Afsal and Kuppusamy (2024) compare the readability of texts generated with the Gemini LLM via different prompts with the Flesch Reading Ease (FRE) and Flesch-Kincaid Grade Level (FKGL) metrics.

Notes: Readability captures 'reading ease' in the sense of text measures like Flesch and Flesch-Kincaid which aim to capture ability to easily read at different reading ages. Better readability is associated e.g. with more common words, shorter words, shorter sentences, and simpler sentence structure. Cf. [QOG-w-3] Fluency: very short sentences with repetitive structure would score highly on Readability, but not on Fluency.

[QOG-w-3] Fluency: A better system produces outputs that are more fluent.

Example: Resendiz and Klinger (2025) evaluate the fluency of affective text generation systems via LLM-prompting with the following prompt: "Assess the text's fluency, assigning a score from 1 to 5, with 5 representing the highest level of fluency. Do not give an explanation of the selection."

Notes: Fluency captures how well text or speech flows, being absorbed readily without bringing the reader or listener up short, and without in the case of speech, hesitations, filler, or overly long pauses. For high fluency, language does not necessarily need to be simple, cf. [QOG-w-2] Readability.

[QOG-w-4] Humanlikeness: A better system produces outputs that are more human-like.

Example: Cercas Curry et al. (2015) assess the humanlikeness of navigation instructions by asking evaluators to rate the extent to which an instruction "could have been produced by a human" on a 4-point scale.

Notes: –

[QOG-w-4.1] Native Speaker Likeness: A better system produces speech or text that is more like that of a native speaker.

Example: [QGO-f-2-1] Novikova et al. (2018) assess system outputs e.g. by asking evaluators the question: "Could the utterance have been produced by a native speaker?"

Notes: –

[QOG-w-4.2] Non-AI Likeness: A better system produces outputs that are less like those produced by AI.

Example: Juraska et al. (2019) ask human judges to assess model outputs in terms of "how much one would expect to encounter [this] utterance in a conversation with a human, as opposed to sounding robotic'.'

Notes: –

[QOG-w-5] Understandability: A better system produces outputs that are more understandable.

Example: Hershenhouse et al. (2024) assess LLMs in terms of their ability to communicate medical information to the public by asking crowdworkers to demonstrate their understanding of generated texts through multiple-choice questions.

Notes: Understandability captures whether an output can be understood and is commonly evaluated in terms of whether it has been understood (via comphrehension questions). Cf. sub-QC [QOG-w-5.1] Clarity for which an output is assessed in terms of the higher-threshold criterion whether it can be *easily* understood.

[QOG-w-5.1] Clarity: A better system produces outputs that are clearer.

Example: Clinciu et al. (2021) evaluate a system that generates explanations of Bayesian network graphs, e.g. in terms of clarity where evaluators are asked to indicate "[h]ow clear the meaning of an explanation is" on a 7-point scale, where 1 = unclear and 7 = very clear.

Notes: –

[QOG-w-5.1.1] Speed of Understanding: A better system produces outputs that are faster to understand.

Example: Mohan (2021) assessed system explanations in terms of the time it took people to select a response when asked how well they understood the explanations generated by the system.

Notes: –

C.1.3 Feature

Form

[QOF-f-1] Diversity/Non-diversity (form): A better system produces outputs that are in their form either (a) more diverse, or (b) less diverse.

Example: Chen et al. (2024) compare the average type-token ratio (ATTR) of poems generated with a range of models, finding e.g. that humans, poetry-specific models and general models have broadly similar ATTR scores.

Notes: Diversity of form captures the variedness of the form of outputs either at the level of individual outputs where those are longer than a sentence, or at the level of a sample of outputs. Varied form could be diverse sentence structures, format, or speech patterns, etc.

[QOF-f-2] Poeticness/Non-poeticness (form): A better system produces outputs that are in their form either (a) more poetic, or (b) less poetic.

Example: Xie et al. (2017) manually compare four LSTM variants in terms of the quality of rhyme and metre, finding that gated and CNN-based LSTMs perform better than word and character LSTMs.

Notes: (Non)poeticness of form captures the degree to which outputs have the formal characteristics of a poem, including alliteration, rhythm, rhyming and specific syllable/metre patterns. For [QOF-f-2], this is assessed without taking the input or anything external to the system into account.

[QOF-f-3] Complexity/Non-complexity (form): A better system produces outputs that are in their form either (a) more complex, or (b) less complex.

Example: In their evaluation of a text simplification system for Spanish, Saggion et al. (2015) measure the syntactic complexity of a sentence as the maximum distance between root and leaf nodes in the dependency tree of the sentence.

Notes: Complexity of form captures aspects of output complexity irrespective of meaning, e.g. longer word and sentence length, nested syntactic structure, and low frequency words can all be indicative of higher complexity of form.

[QOF-f-4] Formality/Informality: A better system produces outputs that are either (a) more formal, or (b) less formal.

Example: Abu Sheikha and Inkpen (2011) asked human annotators to label system outputs from their template-based, formality-controlled text generator as either formal or informal.

Notes: (In)formality captures how relaxed the language of outputs is: in a conversation with friends, or on social media, language tends to be more informal, whereas in academic articles or legal contexts, it tends to be much more formal.

[QOF-f-5] Output Length: A better system produces outputs that are either (a) longer, or (b) shorter.

Example: Ghalandari et al. (2022) report the mean length of the outputs from their unsupervised sentence compression system.

Content

[QOF-c-1] Diversity/Non-diversity (content/meaning): A better system produces outputs that are in their content/meaning either (a) more diverse, or (b) less diverse.

Example: Chen et al. (2024) use mean cosine similarity between SBERT embeddings to assess the semantic diversity of generated poems, finding e.g. that their German poems are on the whole substan-

tially more diverse than the English ones.

Notes: (Non)diversity of content/meaning captures the variedness of the content/meaning of outputs either at the level of individual outputs where those are longer than a sentence, or at the level of a sample of outputs. Varied content/meaning could be diverse topics, information, or events, etc.

[QOF-c-2] Poeticness/Non-poeticness (content/meaning): A better system produces outputs that are in their content/meaning either (a) more poetic, or (b) less poetic.

Example: Wu et al. (2019) ask evaluators to assess "whether some part of the poem is imaginative and/or moving" in a 4-step assessment procedure which is then mapped to a single score.

Notes: (Non)poeticness of form captures the degree to which outputs have the semantic characteristics of a poem, including metaphor, topic and narrative structure. For [QOF-c-2], this is assessed without taking the input or anything external to the system into account.

[QOF-c-3] Complexity/Non-complexity (content/meaning): A better system produces outputs that are in their content/meaning either (a) more complex, or (b) less complex.

Example: Onoe et al. (2024) measure semantic complexity as the number of nodes in the 'scene graph' corresponding to an image description.

As part of the composite LLM Rater metric, Luo et al. (2024) ask evaluators to assess the extent to which a summary "avoid[s] the use of technical details that would be difficult for non-expert readers to understand[, ... and] contains sufficient explanations of any complex terms and abbreviations."

Notes: Complexity of content/meaning captures aspects of output complexity irrespective of form, e.g. complex logical structure or technical details can be indicative of higher semantic complexity.

Outputs as a whole

[QOF-w-1] Diversity/Non-diversity (outputs as a whole): A better system produces outputs that are either (a) more diverse, or (b) less diverse.

Example: Jin and Le (2016) ask evaluators to assess the overall diversity of a set of questions generated by a system from a given input text.

Li et al. (2015) assess the diversity of the responses produced by their conversational system with the distinct-1 and distinct-2 metrics computed as the number of distinct unigrams (bigrams) over the total number of generated tokens.

Notes: (Non)diversity of outputs as a whole captures diversity of form and content both, either at the level of individual outputs where those are longer than a sentence, or at the level of a sample of outputs. E.g. a diverse set of questions and answers generated for a given text passage would have little overlap in coverage of the text between them.

[QOF-w-2] Poeticness/Non-poeticness (outputs as a whole): A better system produces outputs that are either (a) more poetic, or (b) less poetic.

Example: Xie et al. (2017) ask evaluators to assess the overall poeticness of generated poems, finding e.g. that poems generated by more complex LSTMs are perceived as more poetic than those generated by simple LSTMs.

Notes: (Non)poeticness of outputs captures the degree to which outputs have the characteristics of a poem, including alliteration, rhythm, rhyming and specific syllable/metre patterns, metaphor, topic and narrative structure. For [QOF-w-2], this is assessed without taking the input or anything external to the system into account.

[QOF-w-3] Complexity/Non-complexity (outputs as a whole): A better system produces outputs that are either (a) more complex, or (b) less complex.

Example: Angrosh and Siddharthan (2014) compare their rule-based simplification system (which combines handcrafted with automatically acquired rules) to a learned system and a human topline in terms of human-assessed text simplicity, finding that their system outperforms the learned system.

Notes: Complexity of outputs captures aspects of output complexity both of form and content/meaning, e.g. longer word and sentence length, nested syntactic structure, and low frequency words, complex logical structure or technical details can all be indicative of higher complexity.

Zhang et al. (2024) assess the overall layness of generated lay summaries via human evaluation ("to what extent is the content of the model output comprehensible (or readable) to a non-expert, in terms of both structure and language?"), and three automatic metrics.

[QOF-w-4] Conversationality/Non-conversationality: A better system produces outputs that are either (a) more conversational, or (b) less conversational.

Example: Cervone et al. (2019) ask evaluators to assess how conversational the turns generated by different versions of a conversational agent are on a scale of 1–6.

Notes: Conversationality is typically assessed in dialogue scenarios where it captures the degree to which a series of turns between user and system resemble conversations between people, usually in the same situation. While it's difficult to conceive of situations where Nonconversationality would need to be assessed explicitly, it is certainly the case that in many situations user-facing generated text should not be conversational, in particular in single-turn scenarios such as question-answering.

[QOF-w-5] Humorousness/Non-humorousness: A better system produces outputs that are either (a) more humorous, or (b) less humorous.

Example: Alnajjar and Hämäläinen (2018) assess whether generated film titles are humorous, on a 1–5 agreement scale.

Notes: –

C.2 QCs that define quality relative to inputs

Figure 3 shows a simplified view of the **Quality of outputs relative to input** branch of the QCET Taxonomy (only node IDs and Names are shown for each node). Below we list each QC and definition, with some additional explanatory notes in some cases, grouped according to subtrees (see Figure 3).

C.2.1 Correctness

Form

[QIC-f-1] Conformance to Syntactic Structure (given in input): A better system produces texts with fewer deviations from the target syntactic structure provided in the input.

Example: Kumar et al. (2020) compute the tree-edit distance between the syntax exemplar given in the input and the output generated by their syntax-controlled paraphrasing system.

Notes: –

[QIC-f-2] Inclusion of Keywords (given in input): A better system produces texts that lack fewer of the keywords provided in the input.

Example: Sasazawa et al. (2023) compute the proportion of outputs that contain all the keywords given in the input in several keyword position controlled text generation systems.

Notes: –

Content

[QIC-c-1] Absence of Omissions (relative to input): A better system produces outputs that lack fewer of the content/meaning units provided in the input.

Example: González Corbelle et al. (2022) assess their transformer-based weather forecast generator e.g. with a metric that computes the number of output texts with omissions (input elements not literally contained in the output), finding omissions in 160 out of 272 texts (58%).

Notes: Absence of Omissions is defined for cases where Absence of Additions, as its inverse, is also defined. This will typically hold in those cases where the output is supposed to contain/cover/verbalise all and only content/items/formal representations provided in the input. Cf. [QIC-f-2] Inclusion of Keywords where addition is not defined.

[QIC-c-2] Absence of Additions (relative to input): A better system produces outputs that contain fewer content/meaning units not provided in the input.

Example: Cripwell et al. (2023) evaluate the semantic accuracy of the outputs of data-to-text systems in under-resourced languages using several criteria, one of them being the absence of additions. Human evaluators are shown the structured data input and an output text and are asked "Looking at the Text, is all of its content expressed in the Data expression? (Allow duplication of content.)"

Notes: Absence of Additions is defined for cases where Absence of Omissions, as its inverse, is also defined. This will typically hold in those cases where the output is supposed to contain/cover/verbalise all and only content/items/formal representations provided in the input. Cf. the three Nonreduncy QGs, [QOG-f-1], [QOG-c-1], and [QOG-w-1], where omission is not defined.

[QIC-c-3] Consistency with Input: A better system produces outputs that have fewer inconsistencies with a given aspect of the input.

Example: Shu et al. (2021) propose a metric called bi-directional logic evaluation of consistency (BLEC) for evaluating the consistency between database query logic inputs and textual questions in the output.

Notes: Cf. Similarity/Dissimilarity to Input (content/meaning); Consistency with Input is not about meaning similarity, but consistency in a task-specific sense, for which absence of contradictions may be sufficient.

[QIC-c-4] Coverage of Topics (given in input): A better system produces outputs that lack fewer of the topics provided in the input.

Example: Basu Roy Chowdhury et al. (2022) assess summarisation systems in terms of the average number of distinct aspects (fine-grained topics) covered in generated summaries, finding e.g. that the aspect-aware variant of their system increases aspect coverage by about 1 topic on average.

Notes: –

Outputs as a whole

[QIC-w-1] Translation Accuracy: A better system produces translations of the input with fewer translation errors.

Example: Popović (2020) asked annotators to identify and mark up word spans in machine-translated text whose meaning differed from the input text, then investigated different ways of numerically aggregating the annotations.

Notes: When assessed by human evaluators, Translation Accuracy is often broken down into different error types which are assessed (and sometimes reported) separately. However, such error taxonomies differ too widely in granularity and meaning between papers to be incorporated here as sub-QCs. Among metrics, BLEU (created for MT) works particularly well, especially when assessed against multiple target output translations per input.

C.2.2 Goodness

Form

[QIG-f-1] Appropriateness of System Response Type: A better system produces outputs that are of a more appropriate type relative to the input.

Example: Webb et al. (2010) evaluate whether or not it was appropriate (a) for the system to give a response when it responded, and (b) for the stystem not to respond when it didn't, regardless of the content of the response.

Notes: Appropriateness of System Response Type captures, at the level of entire responses, whether the system response was of the right type. Often used in dialogue, evaluation methods assessing this QC might take into account whether the system took initiative or handed over to a human when it should have done. In a question-answering scenario, whether the system response constituted an attempt to answer the question (as distinct from whether that answer was correct or informative) might be assessed.

[QIG-f-2] Success of Style Transfer from Sample: A better system produces outputs that are more like the style sample provided in the input.

Example: In their survey of text style transfer research, Hu et al. (2022) identify two main ways in which previous work has measured style transfer success: transfer acccuracy measured as the accuracy achieved by style classifiers compared to the intended style; and earth mover distance between the style distributions of the input text and the transferred text.

Notes: Cf. Control over Alternative Styles; in Success of Style Transfer from Sample, content and style samples are provided in the input, whereas in 'Control over' QCs, the task definition includes the controlled attribute and its finite set of possible values one of which is added to the input.

[QIG-f-3.1] Success of Speech Style Transfer from Sample: A better system produces speech that sounds more like the speech sample specified in the input.

Example: Peng et al. (2024) evaluate speech LLMs on speech editing tasks where the input is a speech recording, corresponding transcript and an edited version of the transcript, and the output is a spoken rendition of the modified transcript that is intended to sound like the speech recording. The evaluation assesses if the speech sample and the spoken system output sound the same, via metric (source and target sound file comparion) and human evaluation (rate input/output similarity on a scale from 1 to 5).

Notes: As is the case for the parent node, [QIG-f-2] Success of Style Transfer from Sample, here the target style is provided in the form of a sample in the input. In contrast, in [QIF-f-2] Control over Style, typically the system is trained on a given number of different styles which are selected via control attributes in the input.

Content

[QIG-c-1] Answerability from Input: A better system produces questions that are more answerable on the basis of information provided in the input.

Example: In their evaluation of a question generation system, Harrison and Walker (2018) ask human assessors to evaluate on a 4-point scale how much of the information required to correctly answer the generated question is contained within the [input] text passage.

Notes: –

[QIG-c-3] Relevance to Input: A better system produces outputs that are in a given sense more relevant to the input.

Example: Shen et al. (2022) evaluate systems for

the automatic generation of counseling dialogue turns by asking evaluators to indicate on a 3-point scale "whether the response is on topic and relevant to the dialogue history" (the latter being provided in the input).

Notes: Relevance relative to the input is often assessed in a question-answering, instruction-generation or dialogue context, where it captures the relevance of the answer to the user's question, of the instruction to the user's need, and of the dialogue to the dialogue context. 'The given sense' in which relevance is assessed is often specified in human evaluations, e.g. whether the answer responds to the question.

Outputs as a whole

[QIG-w-1] Parse Accuracy (reference-less): A better system produces outputs that are more complete and accurate parses of the input.

Example: Opitz and Frank (2022) evaluate parsers by asking evaluators to asess the 'parse acceptability' of parser outputs (graphs) given the input sentence.

Notes: Parse Accuracy is sometimes assessed by asking parsing experts whether e.g. output phrase-structure or depency parses represent complete and correct analyses of the input sequence.

[QIG-w-2] Degree to which Output Answers Question in Input: A better system produces outputs that are more complete and accurate answers to questions provided in the input.

Example: In the TREC-8 Question Answering shared task Voorhees et al. (1999) ask human assessors whether answers generated by participating systems contain words that answer the input question.

Notes: –

[QIG-w-3] Quality as Explanation of Input: A better system produces outputs that are more complete and accurate explanations of the inputs.

Example: Sarsa et al. (2022) evaluate automatically generated code explanations by answering the question "Are all parts of the code explained?" (Yes / No) and computing the proportion of correctly explained lines out of all the generated explanation lines.

Notes: Cf. [QEG-w-4] Goodness as Explanation of System Behaviour; Quality as Explanation of Input only takes the input and output into account, assessing how well the latter explains the former.

C.2.3 Feature

Form

[QIF-f-1] Control over Complexity/Non-complexity (form): A better system produces outputs that are in their form more at the target level of complexity provided in the input.

Example: Agrawal (2023) assesses the degree to which complexity-controlled machine translation systems succeed in generating translations at the target level of complexity by computing the correlation between the Automatic Readability Index (ARI) scores of the actual and target translations.

Notes: Complexity of form captures aspects of output complexity irrespective of meaning, e.g. longer word and sentence length, nested syntactic structure, and low frequency words can all be indicative of higher complexity of form. [QIF-f-1] Control over Complexity/Non-complexity (form) is for systems where target levels of (non)complexity of form are defined and the system is trained to achieve them.

[QIF-f-2] Control over Style: A better system produces outputs that are in their form more in the target style provided in the input.

Example: Gero et al. (2019) evaluate whether automatically generated sentences exhibit the style specifed in the input, by asking human annotators to label output sentences with style labels and then calculating the accuracy compared to the input labels.

Notes: Style captures aspects of form, as opposed to meaning. It relates to the way something is said, rather than what is said. Examples include formal/informal, literary/non-fiction, and more fine-grained distinctions such as newspaper house style and personal writing style. [QIF-f-2] Control over Style typically is used to evaluate systems that are trained on a specific set of alternative styles, with a control attribute in the input indicating the style that outputs are supposed to be generated in.

[QIF-f-2.1] Control over Formality/Informality: A better system produces outputs that are in their form more at the target level of formality provided in the input.

Example: Yang and Klein (2021) assess the outputs of an informal-to-formal MT system in terms of the mean likelihood that the output is indeed formal according to a formality classifier trained on separate data.

Notes: (In)formality captures how relaxed the lan-

guage of outputs is: in a conversation with friends, or on social media, language tends to be more informal, whereas in academic articles or legal contexts, it tends to be more formal. [QIF-f-2.1] Control over Formality/Informality typically is used to evaluate systems that are trained on data with different levels of formality, with a control attribute in the input indicating the level of formality that outputs are supposed to be generated in. Cf. [QOF-f-4] Formality/Informality which captures the (in)formality of the output when generating outputs with given levels of formality is not part of the system task.

[QIF-f-3] Output Size Relative to Input: A better system produces outputs that achieve, relative to the input, (a) a greater reduction in size, (b) a greater increase in size, or (c) size change at the target level given in the input.

Example: Clarke and Lapata (2006) compare human and machine-produced sentence-level summaries via their respective compression rates (number of tokens in output sentences over number of tokens in input sentences), finding that humans are more conservative than machines.

Notes: [QIG-f-3] Output Size Relative to Input is commonly assessed as the ratio of output size over input size, where size is e.g. measured as number of words or characters. Typical NLP tasks where this QC is assessed include sentence summarisation and simplification.

[QIF-f-4] Similarity/Dissimilarity to Input (form): A better system produces outputs that in their form are (a) more similar to the input, (b) less similar to the input, or (c) more at the target level of similarity to the input, where that target level is provided in the input.

Example: Yin et al. (2022) measure the 'syntactic divergence' between text input and paraphrased output using Kendall's tau.

Content

[QIF-c-1] Control over Complexity/Non-complexity (content/meaning): A better system produces outputs that are in their content/meaning more at the target level of complexity provided in the input.

Example: Imperial et al. (2024) assess content generation which uses English language standards (CEFR, CCS) to control complexity by asking expert assessors to assign English standard levels to generated stories, then measuring the accuracy relative to target levels specified in inputs.

Notes: Complexity of content/meaning captures aspects of output complexity irrespective of form, e.g. complex logical structure or technical details can be indicative of higher semantic complexity. [QIF-c-1] Control over Complexity/Non-complexity (content/meaning) is for systems where target levels of (non)complexity of content/meaning are defined and the system is trained to achieve them.

[QIF-c-2] Similarity/Dissimilarity to Input (content/meaning): A better system produces outputs that in their content/meaning are (a) more similar to the input, (b) less similar to the input, or (c) more at the target level of similarity to the input, where that target level is provided in the input.

Example: In their survey of text style transfer research, Hu et al. (2022) identify some of the main ways in which previous work has measured the meaning (dis)similarity between source text and transferred text: cosine similarity between embeddings, word overlap (excluding style-related words), and human assessment of meaning (dis)similarity.

Notes: [QIF-c-2] Similarity/Dissimilarity to Input (content/meaning) is the same as [QEF-c-1] Similarity/Dissimilarity to Non-target Reference (content/meaning), but here the comparison is against the input, rather than a system-external reference. Typical NLP tasks where this QC is assessed include paraphrasing and style transfer.

[QIF-c-3] Specificity/Non-specificity (relative to input): A better system produces outputs that, relative to some aspect of the input, are (a) more specific, (b) less specific, or (c) more at the target level of specificity provided in the input.

Example: In an evaluation of a content selection method for question generation, Jin and Le (2016) ask human judges to assess in a binary fashion whether or not a question is specific enough considering the input text.

Notes: [QIF-c-3] Specificity/Non-specificity (relative to input) is about the level of specificity with which the output addrsses a given aspect of the input. E.g. where the output answers an input question, does it do so with enough specificity, or is it too vague? Similarly if the output is instructions for a given task, are the instructions specific enough to solve the task?

Outputs as a whole

[QIF-w-1] Control over Complexity/Non-complexity (outputs as a whole): A better system produces outputs that are more at the target level of complex-

ity provided in the input.

Example: Moraes et al. (2016) assess their system's ability to generate graph summaries at the target level of complexity by first asking evaluators to rank systems in terms of complexity (=suitability for grade levels), then computing the proportion of pairwise ranks that match those of the target complexity levels, finding a match in 72% of cases.

Notes: Complexity of outputs captures aspects of output complexity both of form and content/meaning, e.g. longer word and sentence length, nested syntactic structure, and low frequency words, complex logical structure or technical details can all be indicative of higher complexity. [QIF-w-1] Control over Complexity/Non-complexity (outputs as a whole) is for systems where target levels of (non)complexity are defined and the system is trained to achieve them.

Luo et al. (2024) prompt an LLM to summarise articles "for a lay audience," asking evaluators to assess summaries in terms of the following statement:"Layness: Compared to the original article, the summary should decrease the linguistic complexity, omit content that is too technical, and include sufficient background explanation of technical terms."

[QIF-w-2] Control over Sentiment: A better system produces outputs that are more at the target level of positivity/negativity provided in the input.

Example: Kumar et al. (2024) assess the effectiveness of sentiment control (negative, positive, uncontrolled) in a feedback generator as the accuracy according to majority voting by four sentiment classifiers.

Notes: [QIF-w-2] Control over Positive/Negative Sentiment captures, usually at the whole-text level, the overall tone of a text reflecting positive/negative disposition either to its topic(s) overall, or towards a specific aspect. This QC is for systems where target levels of positivity/negativity are defined (commonly positive, neutral, negative) and the system is trained to achieve them (possibly among other things including in LLMs).

[QIF-w-3] Bias Inversion: A better system produces outputs that are more of the inverse bias relative to the input.

Example: Chen et al. (2018) evaluate a system that inverts the political bias of a news article (between left and right) by asking human evaluators to assess whether input/output pairs have fully or partially opposite bias.

Notes: Here, target bias is usually implicit in the system task, e.g. the system would be trained on input-output pairs where the output has the inverse bias of the input. Alternatively, both input and output could have (inverse) bias labels.

[QIF-w-4] Similarity/Dissimilarity to Input (outputs as a whole): A better system produces outputs that overall are (a) more similar to the input, (b) less similar to the input, or (c) more at the target level of similarity to the input, where that target level is provided in the input.

Example: Panthaplackel et al. (2022) ask evaluators to assess whether an updated headline makes only minimal edits to the original headline, i.e. makes changes only to parts that warrant it.

Notes: [QIF-w-4] Similarity/Dissimilarity to Input (outputs as a whole) is the same as [QEF-w-1] Similarity/Dissimilarity to Non-target Reference (outputs as a whole), but here the comparison is against the input, rather than a system-external reference. Typical NLP tasks where this QC is assessed include paraphrasing and style transfer.

[QIF-w-5] Control over Multiple Attributes: A better system produces outputs that are more at the target levels of multiple attributes provided in the input.

Example: Zhong et al. (2024) report results for 'controllability' defined as the average classifier identification success for the controlled attributes.

C.3 QCs that define quality in terms of one or more target outputs

C.3.1 Correctness

Form

[QTC-f-1] Form Accuracy: A better system produces outputs that in their form less often differ from the given target outputs .

Example: Kasner and Dusek (2022) report the accuracy score of their ordering model on WebNLG against the human-generated plans from Ferreira et al. (2018).

Content

[QTC-c-1] Meaning Accuracy: A better system produces outputs that in their content/meaning less often differ from the given target outputs .

Example: Zheng et al. (2024) evaluate free definition generation in terms of the percentage of correct definitions generated, where a generated definition

is correct if it is judged to be semantically equivalent to the given target output by GPT-4.

Outputs as a whole

[QTC-w-1] Classification Accuracy: A better system produces output classes that less often differ from the given target output class (from a given finite set of classes) .

Example: Jarvis et al. (2013) evaluate a classifier that predicts the native language of participants by computing class Recall on 11,000 texts (in 11 languages).

Notes: The notion of accuracy in this QC is wider than just the Accuracy metric, encompassing also e.g. Recall, Precision, F-score, and other combinations of true/false positives/negatives.

[QTC-w-2] Sequence Labelling Accuracy: A better system produces sequences of output labels that less often differ from the given target output label (from a given finite set of labels).

Example: de Vries et al. (2022) compute the part-of-speech (POS) tagging accuracy achieved by a task-tuned model on pairs of languages where one was seen during task-tuning and the other was not. They report POS tagging accuracy for a large number of pairs of languages some of which were seen during model pretraining, some were not.

[QTC-w-3] Complete Target Output Matching: A better system produces outputs that less often differ from a given target output (where the set of possible outputs is not given, and is not necessarily finite).

Example: Yue et al. (2022) report the exact match (EM) rate for question-answer generation systems, where an exact match is a question that appears in the set of target output questions verbatim.

Notes: In contrast to Classification Accuracy, this QC is defined for cases where the system does *not* choose between a set of posible outputs that is known a priori. Instead the output is typically generated in some way from the input, and is often quantified as the exact match rate.

[QTC-w-3.1] Complete Word Matching: A better system produces a words that less often differ from corresponding given target words (where the set of possible output words is not given, and is not necessarily finite).

Example: Pupier et al. (2024) report the word error rate (WER) of systems in a speech recognition task.

[QTC-w-3.2] Character Matching: A better system produces a words that less often differ from corresponding given target words (where the set of possible output characters is not given, and is not necessarily finite).

Example: Pupier et al. (2024) report the character error rate of systems in a speech recognition task.

[QTC-w-4] Retrieval Accuracy: A better system produces query results that less often differ from those in a given set of target query results.

Example: Cheng et al. (2024) report Mean Reciprocal Rank for their conversational retrieval system.

[QTC-w-5] Sequence Alignment Accuracy: A better system produces alignments between two input sequences that have fewer errors compared to a given target alignment.

Example: Latouche et al. (2024) report two word alignment metrics to assess their approach to word sequence alignment: Alignment Error Rate and the percentage of correctly aligned words.

[QTC-w-6] Parse Accuracy (with references): A better system produces parses for input texts that have fewer errors compared to a given target parse.

Example: Pupier et al. (2024) report two parse accuracy metrics for their approach to end-to-end dependency parsing of speech: labelled attachment score (LAS) and unlabelled attachment score (UAS).

C.3.2 Goodness

Form

[QTG-f-1] Similarity to Target Outputs (form): A better system produces outputs that are in their form more similar to given target outputs.

Example: Gero et al. (2019) use BLEU to compute the similarity between (a) reconstructions of test sentences from content and feature representations, and (b) the original test sentences. Because the content is constrained to be the same, this assesses similarity of form.

Notes: Similarity to target outputs is a very common form of evaluation in NLP where one or more target outputs (often called gold outputs or references) are provided as part of a test set, and the degree of similarity between actual system output and target output(s) is measured. Note this is different from binary same/not same assessments made e.g. in Classification Accuracy. [QTG-f-1] Similarity to Target Outputs (form) assesses similarity in terms of form, covering aspects such as morphology, syntax, document structure, style, etc.

Content

[QTG-c-1] Similarity to Target Outputs (content/meaning): A better system produces outputs that are in their content/meaning more similar to given target outputs.

Example: In their study about experiment design for the evaluation of dialogue system ouptuts, Santhanam and Shaikh (2019) compute the cosine similarity between embeddings of (a) system responses and (b) target system responses'.

Mille et al. (2018) evaluate multilingual surface realisers that take syntactic or semantic trees as input by asking raters to assess the meaning similarity between system outputs and the target outputs (i.e. the original sentences previously parsed to get the inputs).

Notes: Similarity to target outputs is a very common form of evaluation in NLP where one or more target outputs (often called gold outputs or references) are provided as part of a test set, and the degree of similarity between actual system output and target output(s) is measured. Note this is different from binary same/not same assessments made e.g. in Classification Accuracy. [QTG-c-1] Similarity to Target Outputs (content/meaning) assesses similarity in terms of content units or semantic representations.

Outputs as a whole

[QTG-w-1] Similarity to Target Outputs (outputs as a whole): A better system produces outputs that are overall more similar to given target outputs.

Example: In the WebNLG shared tasks Gardent et al. (2017), the similarity between outputs of data-to-text generators and target system outputs is evaluated using BLEU (strict n-gram matching) and METEOR (allowing synonyms and morphological variation).

Notes: Similarity to target outputs is a very common form of evaluation in NLP where one or more target outputs (often called gold outputs or references) are provided as part of a test set, and the degree of similarity between actual system output and target output(s) is measured. Note this is different from binary same/not same assessments made e.g. in Classification Accuracy. [QTG-w-1] Similarity to Target Outputs (outputs as a whole) assesses overall similarity, not distinguishing form or content.

[QTG-w-2] Similarity to Inputs and Target Outputs Combined (outputs as a whole): A better system produces outputs that are overall more similar to given target outputs and more similar to the input.

Example: Ingólfsdóttir et al. (2023) use the GLEU metric in their work on grammatical error correction to asses that their system's outputs both make the right corrections (similarity to target outputs), and do so in a way that minimally changes the text (similarity to input texts).

[QTG-w-3] Cross-Dataset Generalisation: A better system produces outputs that obtain higher scores on a given out-of-distribution task dataset .

Example: Huang et al. (2024) report work where a system is trained on a series of tasks, and after each new task training round, the system's performance (i) on the last task it was trained on, and (ii) on the next task it will be trained on, is assessed with ROUGE-L.

Notes: [QTG-w-3] captures the extent to which a system generalises beyond the data distribution on a sample from which it was trained (and in rare cases, created manually). It can be used e.g. to assess transfer learning (without any additional training).

C.4 QCs that define quality in terms of one or more external frames of reference

C.4.1 Correctness

Form

[QEC-f-1] Adherence to Style Guide: A better system produces texts with fewer deviations from a given style guide.

Example: Zhang et al. (2022) report an automatic code refactoring system which transforms Python code into more python-idiomatic, functionally equivalent code on the basis of a given set of idioms. They assess each refactored output e.g. by manually checking that it conforms with Pythonic idiom, reporting an accuracy of 0.998.

Notes: When assessing [QEC-f-1] Adherence to Style Guide, outputs are normally compared with a system-external style guide onn the basis of which style errors can be identified in outputs, and then counted and aggregated.

[QEC-f-2] Adherence to Syntactic Rules: A better system produces outputs with fewer syntactic errors as defined by a given grammar.

Example: Pratapa et al. (2021) present a metric to evaluate the morphosyntactic well-formedness of generated text using dependency parsing and morphosyntactic rule checking.

Notes: [QEC-f-2] Adherence to Syntactic Rules captures grammaticality, or perhaps more accurately parsability, as defined by a given formal grammar, either computational or described in a text resource. In the former case, automatic assessment can establish whether a text can be parsed with a given grammar; in the latter case, evaluators can be asked if a text conforms with the rules described in the grammar. Cf. [QOC-f-1] Grammaticality

Content

[QEC-c-1] Factual Truth: A better system produces texts with fewer real-world untruths.

Example: Thomson and Reiter (2020) evaluate the outputs of their sports summarisation system by asking participants (a) to mark up factual errors as determined by open web search as non-overlapping word spans, then (b) to categorise the word spans. They report an average of 19 erorrs per summary.

Notes: In assessing [QEC-c-1] Factual Truth, the aim is to establish the real-world truth or untruth of output content. In contrast to [QEC-c-2] Relative Factual Accuracy, specific information sources (not expected to contain contradictory information) are not normally provided in evaluation. More typically, a process is described whereby truth is to be established for the purposes of the evaluation which may involve resolving any amount of contradictory information.

[QEC-c-2] Relative Factual Accuracy: A better system produces texts with fewer untruths according to a given source of information.

Example: Min et al. (2023) present FACTSCORE, an estimator that decomposes generated text into atomic facts, then validates each based on a given knowledge source.

Notes: Assessing [QEC-c-2] Relative Factual Accuracy involves consulting a given source of information that is not expected to contain contradictory information (a website, a work of fiction, a database etc.), and checking if outputs are factually accurate relative to the closed world of the information source. I.e. unlike in [QEC-c-1] Factual Truth, there is no attempt to get at real-world truth of output content; whatever the information source states is taken as fact.

Outputs as a whole

[QEC-w-1] Functional Correctness: A better system produces outputs that result in functional behaviour that less often differs from the target behaviour when applied to a given set of tests.

Example: Lee et al. (2024) evaluate the functional quality of generated source code by executing the code and calculating the proportion of times generated code performs correctly on a set of unit tests, reported as pass@1 in tables.

C.4.2 Goodness

Form

[QEG-f-1] Naturalness (form): A better system produces texts that are in their form more natural in a given context and/or for a given subset of speakers.

Example: Mir et al. (2019) evaluate the naturalness of style-transferred texts via classifiers that have been adversarially trained to distinguish transferred from non-transferred texts in the same style domain. The idea is that outputs that are natural texts in the given style will be classified as non-transferred by the classifier. As the meaning is constrained to be the same, this assesses the naturalness of the form of (non-)transferred texts.

Notes: If an output is not natural in form, then it isn't likely to be encountered in this form in the given scenario. Cf. [QEG-f-2] Appropriateness: if a text is not appropriate in form then it shouldn't be in this form in the scenario.

[QEG-f-2] Appropriateness (form): A better system produces texts that are in their form more appropriate in a given context and/or for a given subset of speakers or audience.

Example: Sripada et al. (2014) asked experts from the UK national weather agency to assess whether individual generated forecast texts are of appropriate length.

Notes: If an output is not appropriate in form, then it shouldn't be in this form in the given scenario. Cf. [QEG-f-1] Naturalness: if a text is not natural in form then it isn't likely to be encountered in this form in the scenario, but it's not the case that it shouldn't be.

Content

[QEG-c-1] Naturalness (content/meaning): A better system produces texts that are in their content/meaning more natural in a given context and/or for a given subset of speakers.

Example: Xu et al. (2021) extract event chains from narrative texts and connect them as a graph. To evaluate the quality of the graph, they randomly sample 500 edges and calculate the proportion of

edges which are “suitable for chatting," finding that to be the case for 73.6

Notes: If an output is not natural in content, then it isn't likely to be encountered with this content in the given scenario. Cf. [QEG-f-2] Appropriateness: if a text is not appropriate in content then it shouldn't have this content in the scenario.

[QEG-c-2] Appropriateness (content/meaning): A better system produces texts that are in their content/meaning more appropriate in a given context and/or for a given subset of speakers or audience.

Example: Mahamood and Reiter (2011) evaluate the impact of adding reassurance statements to automatically generated texts giving medical information to parents of pre-term newborns. Parent users rate on a scale of 1–5 the extent to which the text “appropriately considers the parents' emotional state in the given scenario”.

Notes: If an output is not appropriate in content, then it shouldn't have this content in the given scenario. Cf. [QEG-f-1] Naturalness: if a text is not natural in content then it isn't likely to be encountered with this content in the scenario, but it's not the case that it shouldn't be.

Outputs as a whole

[QEG-w-1] Naturalness (outputs as a whole): A better system produces texts that are more natural in a given context and/or for a given subset of speakers.

Example: In the E2E shared task (Dušek et al., 2018), systems generated short restaurant descriptions from a meaning representation; these were manually evaluated on a quasi-continuous scale in terms of naturalness.

Notes: If an output is not natural, then it isn't likely to be encountered in the given scenario. Cf. [QEG-f-2] Appropriateness: if an output is not appropriate then it shouldn't be used in the scenario.

[QEG-w-2] Appropriateness (outputs as a whole): A better system produces texts that are more appropriate in a given context and/or for a given subset of speakers or audience.

Example: Macdonald and Siddharthan (2016) compare the outputs of two news summarisers, one producing basic summaries and the other producing summaries suitable for children. They ask human participants the following question: “Overall, which of these summaries do you believe is more suitable for a child?”

Notes: If an output is not appropriate, then it shouldn't be used in the given scenario. Cf. [QEG-f-1] Naturalness: if an output is not natural then it isn't likely to be encountered in the scenario, but it's not the case that it shouldn't be.

[QEG-w-3] Usefulness (nonspecific): A better system produces outputs that more useful to the user.

Example: Colineau et al. (2002) evaluate a system that generates interactive instructions for technical writers using a text editor by asking users to indicate how they would “evaluate the usefulness of the help” on a 6-point scale.

Notes: [QEG-w-3] Usefulness (nonspecific) is often assessed simply by asking users how useful they find the system. Cf. User Satisfaction as Affected by Outputs: users can be satisfied with a system (e.g. one whose primary purpose is entertainment) without also finding it useful. See also the more specific sub-QC [QEG-w-3.1] Usefulness for Task/Information Need.

[QEG-w-3.1] Usefulness for Task/Information Need: A better system produces outputs that are more useful for the user's task and/or information need.

Example: Qu and Green (2002) assess a cooperative mixed-initiative dialogue system for information-seeking dialogue via a user study where they measure the agreement between “the user's recorded solution for each task” and “the user's original information need” with the kappa statistic (Carletta, 1992).

Notes: [QEG-w-3.1] Usefulness for Task/Information Need shares the same characteristics of its parent QC[QEG-w-3] Usefulness (nonspecific), but is more specific, assessing usefulness for a given task, such as following generated instructions to trouble-shoot a malfunctioning app, or for a given information need, e.g. are the accommodation descriptions on a website useful in selecting a holiday rental.

[QEG-w-4] Goodness as Explanation of System Behaviour: A better system produces explanations that better help the user understand its behaviour .

Example: Chiyah Garcia et al. (2018) evaluate a system that generates natural language explanations for an autonomous underwater vehicle by asking evaluators to indicate (dis)agreement with the following statement on a 7-point scale: “'Worth it' question: It would be worth reading the explanations to understand how the system is behaving.”

Notes: Cf. [QIG-w-3] Quality as Explanation of Input; Quality as Explanation of System Behaviour assesses how well the system can explain its own outputs or other aspects of its behaviour, e.g. what the evidence was on the basis of which it rejects an application for job or a loan.

[QEG-w-5] System Usability as Affected by Outputs: A better system is more usable by the user, where compared systems differ only in their outputs.

Example: Zheng et al. (2015) evaluate five varied NLP systems using a software usability tool called TURF which records interaction streams and user audio; they also evaluate ease of use on a scale from -1 to 2.

Notes: In software development, Usability is commonly defined in terms of the effort required to use a system. The scope of the QCET taxonomy is evaluation of NLP systems, so usability evaluation in this context would vary just the NLP components, not e.g. the user interface or animation design.

[QEG-w-5.1] Ease of Communication: A better system produces outputs that make communication with the user easier.

Example: In their evaluation of ease of communication with a new dialogue system, Rieser et al. (2011) ask evaluators for a rating on a scale from 1 to 6 for how well users perceived they were understood by the system.

Notes: [QEG-w-5.1] Ease of Communication captures the perceived or measured ease with which users (i) convey their communicative goals to the system, and (ii) understand what the system tells them.

[QEG-w-5.2] Task Completion Speed: A better system produces outputs that result in faster task completion by the user.

Example: Qu and Green (2002) give users flight reservation tasks to evaluate two versions of a dialogue system in terms of task completion time, finding that the system-initiative version allows users to solve tasks more quickly.

Notes: This QC is assessed with a system, a user and a task that the user has to complete with the system. The speed with which the user completes the task is measured. Task Completion Speed is independ o overall interaction duration. Cf. [QEF-w-8] Interaction Completion Speed.

[QEG-w-6] User Satisfaction as Affected by Outputs: A better system is one that users are more satisfied with, where compared systems differ only in their outputs.

Example: Mulia et al. (2023) evaluate the satisfaction of users with ChatGPT via (dis)agreement with the statement "I think that I would like to use this system frequently," finding a mean agreement score of 3.95 out of 5.

Notes: User Satisfaction is about how happy users are with their use of a system, and is commonly assessed by asking users directly how satisfied they are, or via tracking repeat use. In the present context of NLP system evaluation, evaluations would take into account variations of the NLP component(s), but not other aspects like other functionality, user interface, etc.

[QEG-w-7] Clarity of Referents: A better system produces referring expressions that more clearly identify their referents.

Example: In the 2005 DUC shared task on summarisation, Dang (2005) manually assess systems in terms of 'Referential clarity,' defined as follows: "It should be easy to identify who or what the pronouns and noun phrases in the summary are referring to. If a person or other entity is mentioned, it should be clear what their role in the story is. So, a reference would be unclear if an entity is referenced but its identity or relation to the story remains unclear."

Notes: Clarity of referents is about how readily intended referents can be identified from referring expressions in texts or other representations. The explanation included in the DUC 2005 attestation provides a good explanation for the case of texts.

[QEG-w-8] Performance of an Embedding/Downstream System/Component: A better system produces outputs that result in outputs with better performance when used by another system or component.

Example: Reddy et al. (2017) evaluate a system that generates question-answer pairs from keywords by assessing whether its outputs can improve the performance of a semantic parser when added to its training data. Comparing performance when (a) training on manual question-answer pairs only with (b) training on the manual data plus the system's question-answer pairs, they find a 5.5% improvement for the augmented training data.

Notes: Assessing a system in terms of the impact its outputs have on the performance of the bigger system it is part of, or in terms of another system

that uses its outputs, is often called an extrinsic form of evaluation. Extrinsic evaluation is especially suitable for NLP components embedded in a larger system, such as a TTS component that is part of an interactive system.

[QEG-w-9] Multi-task Performance: A better system produces outputs that obtain higher aggregated scores on a given set of task datasets and metrics .

Example: Zhou et al. (2023) explore data leakage in LLM assessment, evaluating different models on the MMLU benchmark of 57 different tasks that require real-world knowledge and problem-solving abilities.

Notes: Multi-task benchmarks have become increasingly common in NLP, particularly in LLM evaluation. [QEG-w-10] Multi-task Performance covers any case where aggregated results expressing performance at multiple tasks is reported.

[QEG-w-10] Win Rate: A better system produces outputs that more often beat a given comparitor system or system stand-in.

Example: Wang et al. (2024b) evaluate a model by measuring the fraction of times a powerful LLM (e.g. GPT-4) prefers the outputs from that model over outputs from a reference model.

C.4.3 Feature

Form

[QEF-f-1] Similarity/Dissimilarity to Non-target Reference (form): A better system produces outputs that are in their form (a) more similar, (b) less similar, or (c) more at the target level of similarity, compared to given references that are not target system outputs.

Example: Chim et al. (2024) assess various types of similarity of synthetic user-generated text with 'known source data' that was included in training data for the generating model. One type of similarity they look at is idiolect preservation where they measure the cosine similarity between idiolect embeddings that reflect stylistic idiosyncrasies to capture style preservation relative to the source data.

Notes: We use the term 'non-target references' here to refer to texts, speech, or structured representations which system outputs are compared against, but which do not have the status of a target system output for the given input. In the case of [QEF-f-1] Similarity/Dissimilarity to Non-target Reference (form), such comparisons assess form similarity. This is the case e.g. when the similarity of outputs to a sample of multiple texts, speech, or structured representations (that are not target outputs) is measured as an indication of stylistic or other form alignment.

Content

[QEF-c-1] Similarity/Dissimilarity to Non-target Reference (content/meaning): A better system produces outputs that are in their content/meaning (a) more similar, (b) less similar, or (c) more at the target level of similarity, compared to given references that are not target system outputs.

Example: Shibayama et al. (2021) propose a new metric to assess the degree of novelty of scientific articles, which uses the distance relative to embeddings of cited articles, and show that a larger average distance correlates with a higher level of novelty.

Notes: We use the term 'non-target references' here to refer to texts, speech, or structured representations which system outputs are compared against, but which do not have the status of a target system output for the given input. In the case of [QEF-f-1] Similarity/Dissimilarity to Non-target Reference (content/meaning), such comparisons assess content/meaning similarity. This is the case e.g. when the similarity of outputs to a sample of multiple texts, speech, or structured representations (that are not target outputs) is measured as an indication of topic or other semantic alignment.

Outputs as a whole

[QEF-w-1] Similarity/Dissimilarity to Non-target Reference (outputs as a whole): A better system produces outputs that are overall (a) more similar, (b) less similar, or (c) more at the target level of similarity, compared to given references that are not target system outputs.

Example: Chim et al. (2024) assess various types of similarity of synthetic user-generated text with 'known source data' that was included in training data for the generating model. One type of similarity they look at is divergence (text overlap) as an intrinsic proxy for privacy preservation, for which they compute the BLEU score between source-data text and synthetic text, reporting divergence per data point as 1–BLEU(s, t).

Notes: We use the term 'non-target references' here to refer to texts, speech, or structured representations which system outputs are compared against, but which do not have the status of a target system

output for the given input. In the case of [QEF-f-1] Similarity/Dissimilarity to Non-target Reference (outputs as a whole), such comparisons assess overall output similarity. This is the case e.g. when the similarity of outputs to a sample of multiple texts, speech, or structured representations (that are not target outputs) is measured as an indication of overall alignment.

[QEF-w-2] Effect on User Behaviour: A better system produces outputs that affect the user's behaviour (a) more, (b) less, or (c) as specified in the input, in terms of a given range of possible behaviours.

Example: Davis et al. (2020) evaluate a virtual health assistant *inter alia* in terms of user diet adherence and overall goal achievement, finding that mean dietary adherence was 91% and was lowest for discretionary foods, grains, red meat, and vegetables. Participants met their step goal 59% of the time.

Notes: [QEF-w-2] Effect on User Behaviour captures changes in the behaviour of users as a result of using the system. Cf. [QEF-w-3] Effect on User Emotion; [QEF-w-6] Effect on User Opinion; and [QEF-w-7] Effect on User Stance. Examples of behaviours include driving behaviour, diet adherence, smoking, and exercise.

[QEF-w-3] Effect on User Emotion: A better system produces outputs that affect the user's emotions (a) more, (b) less, or (c) as specified in the input, in terms of a given range of possible emotions.

Example: van der Sluis and Mellish (2009) evaluate text generation strategies aimed at emphasising positive feedback in mixed-feedback texts via lexical and syntactic choice in terms of the effect on users' emotions. One type of measure asks users to rate the strength with which they are feeling different emotions before and after receiving feedback (in the form of human-written stand-ins); they find that the system version using positive emphasis strategies has a distinct positive effect, increasing mean ratings for all tested emotions.

Notes: [QEF-w-3] Effect on User Emotion captures changes in the emotional state of users as a result of using the system. Cf. [QEF-w-2] Effect on User Behaviour; [QEF-w-6] Effect on User Opinion; and [QEF-w-7] Effect on User Stance. An example of a set of emotion classes is Ekman et al.'s six cross-cultural basic emotions: anger, disgust, fear, happiness, sadness and surprise.

[QEF-w-4] Detectability of Speaker/Author Stance: A better system produces outputs that make the entity producing the output come across to an observer as having one of a range of given stances (a) more, (b) less, or (c) to the degree specified in the input.

Example: van der Lee et al. (2017) evaluate a system that generates football game summaries for (a) supporters of the home team, and (b) supporters of the away team, by asking evaluators who they they think the summaries are intended for, finding that evaluators identified the correct team in 91% of all cases.

Notes: [QEF-w-4] Detectability of Speaker/Author Stance is about the degree to which the user perceives the entity producing the outputs (which may be perceived as an interlocutor) as having a given stance towards a given object. In contrast to trait ([QEF-w-5] Detectability of Speaker/Author Trait), a stance cannot be expressed in a single adjective. An example is (strength of) support for a sports team, as in the example attestation.

[QEF-w-5] Detectability of Speaker/Author Trait: A better system produces outputs that make the entity producing the output come across to an observer as having one of a range of given traits (a) more, (b) less, or (c) to the degree specified in the input.

Example: Glas and Pelachaud (2015) evaluate different strategies for dialogue agents to introduce new topics into conversation, assessing which makes the user perceive the agent as more (a) competent, (b) friendly, (c) fun, and (d) informed, i.e. four different traits.

Notes: [QEF-w-5] Detectability of Speaker/Author Trait is about the degree to which the user perceives the entity producing the outputs (which may be perceived as an interlocutor) as having a given trait. A trait in this context is usually something that can be captured in a single adjective, as in the example attestation.

[QEF-w-6] Effect on User Opinion: A better system produces outputs that affect the user's opinions (a) more, (b) less, or (c) as specified in the input, in terms of a given range of possible opinions.

Example: Wu et al. (2023) evaluated six versions of a web search system in terms of the extent to which user opinions change on several topics after system use. Opinions for and against are captured at the start and the end on a +6 to -6 scale and opinion change is computed as the difference between the

value at the start and the value at the end.

Notes: [QEF-w-6] Effect on User Opinion captures changes in user opinion about a given topic as a result of using the system. Cf. [QEF-w-2] Effect on User Behaviour; [QEF-w-3] Effect on User Emotion; and [QEF-w-7] Effect on User Stance. Opinion is more fine-grained than stance, and while an opinion can be typical of a stance, it would take more than one opinion to identify a stance. E.g. favouring the introduction of a universal income is a political opinion, not an overall political stance.

[QEF-w-7] Effect on User Stance: A better system produces outputs that affect the user's stance (a) more, (b) less, or (c) as specified in the input, in terms of a given range of possible stances.

Example: Forrest et al. (2018) compare explainability components for use with machine learning tools where components use either (i) language generation, or (ii) more numbers-based forms of explanation, by asking users if they "would trust a decision with this explanation," finding that the components that use language-generation inspire more trust in users.

Notes: [QEF-w-7] Effect on User Stance captures changes in user stance as a result of using the system. Cf. [QEF-w-2] Effect on User Behaviour; [QEF-w-3] Effect on User Emotion; and [QEF-w-6] Effect on User Opinion. Stance is more coarse-grained than opinion, typically comprising multiple related opinions. E.g. Left-leaning and right-leaning are political stances, not in themselves political opinions.

[QEF-w-8] Interaction Completion Speed: A better system produces outputs that result in interactions with the user completing in (a) more time, (b) less time, or (c) as much time as specified in the input.

Example: Peng et al. (2017) evaluate four dialogue systems in terms of the average number of turns in simulated user interactions with three user types.

Notes: [QEF-w-8] Interaction Completion Speed captures how quickly user-system interactions are completed. This can be measured in different ways, including number of turns, number of system turns, length of turns altogether, or clock time. Note that in many cases, a better system will have faster interaction completion speed, but in other cases the aim may be to keep the user interacting for as long as possible.

[QEF-w-9] Likelihood According to External Model: A better system produces outputs that are estimated to be more likely by a given external model.

Example: Yedetore et al. (2023) train various models (5-gram model, LSTMs, Transformers) on child-directed language data, and use perplexity (a standard formulation as well as the word-frequency normalised SLOR metric) to evaluate how well each model captures the basic structure of the training domain, finding that Transformers have the lowest perplexity, the 5-gram model the highest.

Notes: The more common use of perplexity is in evaluations where *low* perplexity as computed with a given model is desirable, where it's seen as indicative of 'natural' output. However, it can equally be desirable for outputs to have high perplexity, e.g. in situation where a different style from that encapsulated by the model is intended. [QEF-w-9] Model Perplexity is a Feature-type QC, hence captures both possibilities. Note that various metrics exist for measuring model perplexity including normalised ones such as SLOR.