Data-Based Natural Language Generation : Evaluation and Explainability

Juliette Faille

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Data-Based Natural Language Generation : Evaluation and Explainability

THÈSE

présentée et soutenue publiquement le 17 novembre 2023

pour l'obtention du

Doctorat de l'Université de Lorraine

(mention informatique)

par

Juliette Faille

Composition du jury

Présidente du jury :	Chloé Clavel	Professeure, LTCI, Telecom-Paris, Institut Polytechnique de Paris
Rapporteurs :	Cyril Labbé	Professeur, Université de Grenoble-Alpes
	Benjamin Piwowarski	Chargé de Recherche (HdR), CNRS, ISIR, Sorbonne Université
Examinatrices :	Chloé Clavel	Professeure, LTCI, Telecom-Paris, Institut Polytechnique de Paris
	Laure Soulier	Maîtresse de conférences (HdR), ISIR, Sorbonne Université
Directrice de thèse :	Claire Gardent	Directrice de recherche, CNRS, LORIA
Co-Directeur de thèse:	Albert Gatt	Professeur, Université d'Utrecht

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thesul.

NL4XAI

Interactive *Natural Language*
Technology for eXplainable
Artificial Intelligence

Disclaimer & acknowledgment This thesis was carried out in the context of the NL4XAI research project. The NL4XAI project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860621. This document reflects the views of the author(s) and does not necessarily reflect the views or policy of the European Commission. The REA cannot be held responsible for any use that may be made of the information this document contains.

Acknowledgements

I would like to thank my PhD advisors, Claire Gardent and Albert Gatt. I am really grateful for their guidance, which was always very considerate and of high scientific level. I enjoyed learning how to do research with both of them.

I also would like to thank all the team members of the research groups I had the chance to visit during my PhD project, in particular the members of NLP groups at the Universities of Malta and Utrecht, and the NADIA team at Orange Labs. In particular, thank you to Michele Cafagna and Ettore Mariotti with whom I collaborated during the different secondments, I really enjoyed our discussions. I also thank Lina Rojas, Quentin Brabant, and Gwénolé Lecorvé with whom I worked at Orange Labs.

I would like to thank the members, researchers, and ESRs of the NL4XAI project. I really appreciated meeting with them during the project training events.

I would like to thank the members of the Synalp Team at Loria and in particular Kelvin, Liam, William, Anna L, Anna N, Joe, Barbara, Teven, Vadim, Yannick, Christophe, Gaël, Angela, Anastasia, Paul, Guillaume, Maxime, Duy, and Karolin. They made my time at Loria really enjoyable. (And in particular, thank you to Liam for proofreading parts of this thesis !)

I am really grateful for all the people I got to meet during the past three years.

Finally, I would like to thank the people who supported me during this PhD project and during all my graduate studies, my parents, my grandmother, my grandfather, my sister, and of course for his support and encouragement each time I needed them, Alexandre.

À ma famille.

Abstract

Recent Natural Language Generation (NLG) models achieve very high average performance. Their output texts are generally grammatically and syntactically correct which makes them sound natural. Though the semantics of the texts are right in most cases, even the state-of-the-art NLG models still produce texts with partially incorrect meanings. In this thesis, we propose evaluating and analyzing content-related issues of models used in the NLG tasks of Resource Description Framework (RDF) graphs verbalization and conversational question generation.

First, we focus on the task of RDF verbalization and the omissions and hallucinations of RDF entities, i.e. when an automatically generated text does not mention all the input RDF entities or mentions other entities than those in the input. We evaluate 25 RDF verbalization models on the WebNLG dataset. We develop a method to automatically detect omissions and hallucinations of RDF entities in the outputs of these models. We propose a metric based on omissions or hallucination counts to quantify the semantic adequacy of the NLG models. We find that this metric correlates well with what human annotators consider to be semantically correct and show that even state-of-the-art models are subject to omissions and hallucinations.

Following this observation about the tendency of RDF verbalization models to generate texts with content-related issues, we propose to analyze the encoder of two such state-of-the-art models, BART and T5. We use the probing explainability method and introduce two probing classifiers (one parametric and one non-parametric) to detect omissions and distortions of RDF input entities in the embeddings of the encoder-decoder models. We find that such probing classifiers are able to detect these mistakes in the encodings, suggesting that the encoder of the models is responsible for some loss of information about omitted and distorted entities.

Finally, we propose a T5-based conversational question generation model that in addition to generating a question based on an input RDF graph and a conversational context, generates both a question and its corresponding RDF triples. This setting allows us to introduce a fine-grained evaluation procedure automatically assessing coherence with the conversation context and the semantic adequacy with the input RDF.

Our contributions belong to the fields of NLG evaluation and explainability and use techniques and methodologies from these two research fields in order to work towards providing more reliable NLG models.

Keywords: NLG, Evaluation, Explainability, RDF-to-Text, Conversation Question Generation, Semantic Adequacy, Probing, Omissions, Hallucinations, Distortions, Dialog Coherence

Résumé

Les modèles de génération de langage naturel (NLG) ont récemment atteint de très hautes performances. Les textes qu'ils produisent sont généralement corrects sur le plan grammatical et syntaxique, ce qui les rend naturels. Bien que leur sens soit correct dans la grande majorité des cas, même les modèles de NLG les plus avancés produisent encore des textes avec des significations partiellement inexactes. Dans cette thèse, en nous concentrons sur le cas particulier des problèmes liés au contenu des textes générés, nous proposons d'évaluer et d'analyser les modèles utilisés dans les tâches de verbalisation de graphes RDF (Resource Description Framework) et de génération de questions conversationnelles.

Tout d'abord, nous étudions la tâche de verbalisation des graphes RDF et en particulier les omissions et hallucinations d'entités RDF, c'est-à-dire lorsqu'un texte généré automatiquement ne mentionne pas toutes les entités du graphe RDF d'entrée ou mentionne d'autres entités que celles du graphe d'entrée. Nous évaluons 25 modèles de verbalisation de graphes RDF sur les données WebNLG. Nous développons une méthode pour détecter automatiquement les omissions et les hallucinations d'entités RDF dans les sorties de ces modèles. Nous proposons une métrique basée sur le nombre d'omissions ou d'hallucinations pour quantifier l'adéquation sémantique des modèles NLG avec l'entrée. Nous constatons que cette métrique est corrélée avec ce que les annotateurs humains considèrent comme sémantiquement correct et nous montrons que même les modèles les plus globalement performants sont sujets à des omissions et à des hallucinations.

Suite à cette observation sur la tendance des modèles de verbalisation RDF à générer des textes avec des problèmes liés au contenu, nous proposons d'analyser l'encodeur de deux de ces modèles, BART et T5. Nous utilisons une méthode d'explicabilité par sondage et introduisons deux sondes de classification, l'une paramétrique et l'autre non paramétrique, afin de détecter les omissions et les déformations des entités RDF dans les plongements lexicaux des modèles encodeur-décodeur. Nous constatons que ces classifieurs sont capables de détecter ces erreurs dans les encodages, ce qui suggère que l'encodeur des modèles est responsable d'une certaine perte d'informations sur les entités omises et déformées.

Enfin, nous proposons un modèle de génération de questions conversationnelles basé sur T5 qui, en plus de générer une question basée sur un graphe RDF d'entrée et un contexte conversationnel, génère à la fois une question et le triplet RDF correspondant. Ce modèle nous permet d'introduire une procédure d'évaluation fine évaluant automatiquement la cohérence avec le contexte de la conversation et l'adéquation sémantique avec le graphe RDF d'entrée.

Nos contributions s'inscrivent dans les domaines de l'évaluation en NLG et de l'explicabilité. Nous empruntons des techniques et des méthodologies à ces deux domaines de recherche afin d'améliorer la fiabilité des modèles de génération de texte.

Mots-clés: Génération automatique de texte, Evaluation, Explicabilité, RDF-to-Text, Génération de Questions Conversationnelles, Adéquation sémantique, Sondage, Omissions, Hallucinations, Déformations, Cohérence des dialogues

Evaluation et Analyse des Modèles Neuronaux de Génération Automatique de Texte

Les modèles de génération automatique de langage naturel (en anglais Natural Language Generation, ou NLG) sont désormais largement utilisés, y compris par le grand public. En effet, les applications de la génération automatique de texte touchent désormais de nombreux domaines tels que les services à la clientèle (avec l'utilisation de chatbots dans des domaines variés, par exemple les planificateurs de voyages, le commerce en ligne ou les assurances), les aides à la création de contenu avec des outils performants de résumé et de paraphrase. Certaines applications Image-to-Text et Speech-to-Text utilisent et incluent également des modules NLG.

Disposer de modèles NLG fiables et comprendre leurs limites est donc un enjeu important. L'évaluation et l'explicabilité sont deux approches visant à améliorer la qualité et la fiabilité des modèles. Dans les paragraphes suivants, nous présentons les domaines de l'explicabilité, de l'évaluation des modèles de NLG ainsi que le cadre de cette thèse.

Évaluation des modèles NLG

Les modèles récents pour la génération automatique de texte sont des modèles d'apprentissage profond qui peuvent être considérés comme des "boîtes noires" ("black-box" en anglais), c'est-à-dire des modèles dont le fonctionnement interne n'est pas accessible ou compréhensible pour les humains. Le manque d'explicabilité (XAI) est souvent considéré comme le principal obstacle au déploiement pratique de l'intelligence artificielle (IA) (Barredo Arrieta et al., 2019). L'explication des modèles est susceptible d'accroître les performances en aidant à créer des modèles plus robustes et d'améliorer la confiance des utilisateurs en rendant les modèles plus interprétables. D'un point de vue juridique et suite à la loi européenne sur l'IA ("AI Act »), l'interprétabilité sera requise pour le déploiement de systèmes d'IA dans certaines applications, en particulier celles à haut risque. Dans cette même optique, l'étude de (Madsen et al., 2022) sur l'explicabilité post-hoc des modèles de NLP mentionne quatre motivations principales pour l'explicabilité : l'éthique, la sécurité, la responsabilité et la compréhension scientifique. Dans cette thèse, nous utilisons la définition de l'explicabilité et de l'interprétabilité de Barredo Arrieta et al. (2019). Un *modèle interprétable* est un modèle directement compréhensible par un être humain. Un *modèle explicable* est un modèle qui peut être expliqué à un humain en utilisant une interface entre le modèle et l'humain. Comme les modèles de NLG les plus récents sont des boîtes noires et ne peuvent pas être directement compris par les humains, nous nous concentrons dans cette thèse sur l'explicabilité.

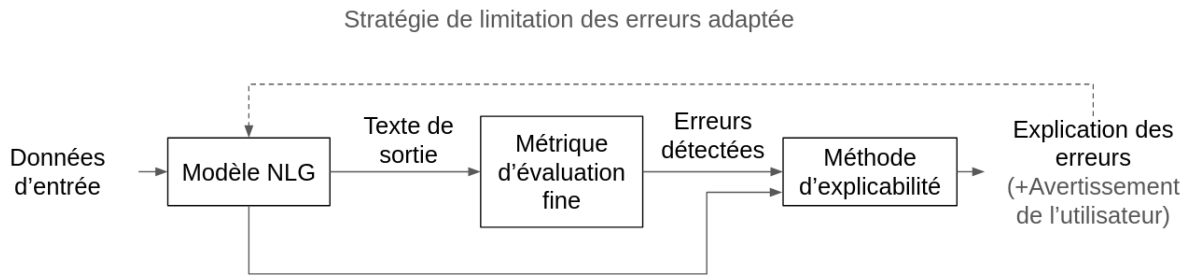


FIGURE 1 – Diagramme fonctionnel montrant comment les métriques d'évaluation et les méthodes d'explicabilité peuvent être combinées afin d'améliorer les modèles existants de génération de texte

Évaluation des modèles NLG

Le NLG consiste à générer automatiquement un texte. Dans la plupart des applications de NLG, pour être considéré comme de bonne qualité, un texte généré automatiquement doit paraître naturel ou fluide et avoir un sens correct. L'évaluation est le sous-domaine du NLG qui traite de l'appréciation de la qualité des textes générés. Ce sous-domaine de recherche a été très actif ces dernières années et de nombreuses métriques d'évaluation ont été proposées, comme le montrent les articles d'étude bibliographique tels que (Gatt and Krahmer, 2018), (Sai et al., 2022) ou (Celikyilmaz et al., 2020), ainsi que le récent benchmark GEM (Gehrmann et al., 2021). Alors que la fluidité, le naturel ou la lisibilité permettent de déterminer si le texte est correct d'un point de vue linguistique, l'évaluation liée au contenu est liée à la sémantique et permet de déterminer si le sens du texte est correct. Les modèles de NLG en anglais récents sont des modèles préentraînés. Ils obtiennent de très bons résultats pour les critères de fluidité. Toutefois, des problèmes liés au contenu sont encore présents dans certains textes générés y compris par les modèles les plus performants. Les problèmes liés au contenu peuvent être analysés à différents niveaux, tels que le mot, la phrase ou le discours, et englobent de nombreux aspects différents d'un texte, tels que la cohérence, la fidélité ou l'utilité dans le contexte d'une application particulière. La qualité générale des modèles récents étant très élevée, des méthodes d'évaluation et de détection fines sont nécessaires pour identifier les erreurs.

Lien entre l'explicabilité et l'évaluation des modèles NLG

Dans le contexte d'un besoin croissant de modèles d'IA fiables, la recherche visant à rendre les modèles de boîte noire compréhensibles par les utilisateurs a prospéré ces dernières années, avec de nombreux articles publiés dans les domaines de l'explicabilité et de l'interprétabilité. Cependant, très peu de travaux se sont concentrés sur les modèles de génération de texte. D'autres approches visant à améliorer la fiabilité des modèles de NLG ont été plus largement étudiées, à savoir l'évaluation et la contrôlabilité des modèles de NLG. Dans cette thèse, nous visons à créer des liens entre les domaines de l'explicabilité et de l'évaluation des modèles de NLG. En effet, l'évaluation et l'explicabilité peuvent être considérées comme des approches complémentaires pour obtenir des modèles plus robustes et compréhensibles. L'évaluation peut être utilisée pour détecter et quantifier les erreurs et l'explicabilité pour les analyser et les comprendre. La figure 1 illustre la façon dont les méthodes d'évaluation et d'explicabilité peuvent interagir.

Graphe RDF d'entrée	(*Lady_Anne_Monson*, *birthPlace*, *Darlington*) (*Lady_Anne_Monson*, *birthDate*, *1726-01-01*) (*Lady_Anne_Monson*, *birthPlace*, *Kingdom_of_England*) (*Lady_Anne_Monson*, *residence*, *India*)
Texte Généré	Lady Anne Monson was born in England on January, 1st 1826, and lives in Germany.

TABLE 1 – Exemples de verbalisation d'un graphe RDF contenant des hallucinations, omissions et déformations.

Champ d'application de la thèse

Dans cette thèse, nous considérons une sous-tâche de la génération de texte, la tâche de Data-to-Text dans laquelle un texte de sortie est automatiquement généré à partir de certaines données d'entrée. Nous étudions deux applications particulières de la génération de texte à partir de données en anglais : la verbalisation de graphes RDF (Resource Description Framework) et la génération de questions conversationnelles utilisant également des graphes RDF. Dans le contexte de la verbalisation RDF, les erreurs sémantiques incluent les *hallucinations* (lorsque le modèle génère des informations qui ne sont pas dans l'entrée) et les *omissions* (lorsque certaines informations sont manquantes dans le texte de sortie). Nous définissons également les *déformations* qui peuvent être considérées comme intermédiaires entre les hallucinations et les omissions, i.e. lorsque certaines informations d'entrée sont partiellement manquantes ou incorrectes. Des exemples de ces erreurs sont présentés dans la table 1. Dans le contexte de la génération de questions conversationnelles, la *cohérence du dialogue* (i.e. la cohérence de la question générée avec le contexte de la conversation) et l'*adéquation sémantique* avec le RDF d'entrée doivent également être vérifiées. Les problèmes liés au contenu comprennent de ce fait également les répétitions ou les contradictions entre la question générée et le contexte de la conversation. Des exemples d'erreurs sont présentés dans la table 2.

Questions de recherche et contributions

Notre objectif de recherche est d'étudier les problèmes liés au contenu dans les modèles de génération de textes à partir de graphes RDF (RDF-to-Text) avec comme but final de résoudre ou d'atténuer ces problèmes. Nous abordons les questions de recherche suivantes :

RQ1 : Dans quelle mesure les modèles RDF-to-Text de pointe présentent-ils des problèmes liés au contenu, en particulier des omissions et des hallucinations ?

Pour répondre à cette première question de recherche, nous étudions dans le chapitre 2 les modèles de verbalisation en anglais de graphes RDF soumis aux WebNLG Challenges 2017 et 2020 (Gardent et al., 2017; Castro Ferreira et al., 2020a).

Dans le cas des verbalisateurs RDF, une condition nécessaire pour que le texte généré soit sémantiquement adéquat est que (i) toutes les entités présentes dans l'entrée soient mentionnées au moins une fois dans la sortie, et (ii) qu'aucune autre entité que les entités RDF d'entrée ne soit mentionnée dans la sortie. Nous désignons ce critère par le terme d'adéquation sémantique basée sur les entités (ESA en abrégé) et proposons une métrique reposant sur les nombre d'omissions ou d'hallucinations détectées automatiquement dans les sorties des différents modèles. À l'aide de cette mesure, nous quantifions les omissions et les hallucinations faites par les différents modèles et nous les comparons.

Notamment, en moyenne, pour les modèles les plus récents, nous avons détecté que 17% des textes avaient au moins une entité manquante. Grâce à une vérification manuelle sur environ 500 textes, nous

Graphe RDF d'entrée	(*Anna-Karin Stromstedt*, *country of citizen- ship*, *Sweden*) (*Anna-Karin Stromstedt*, *place of birth*, *Vansbro Municipality*) (*Sweden*, *capital*, *Stockholm*) (*Vansbro Municipality*, *located in the administrative territorial entity*, *Dalarna County*) (*Vansbro Municipality*, *country*, *Sweden*) (*Anna-Karin Stromstedt*, *sport*, *biathlon*) (*Anna-Karin Stromstedt*, *participant in*, *2006 Winter Olympics*)
Context du dialogue	In what city was Anna-Karin Strömstedt born ? Vansbro Municipality Which sport did she play ? Biathlon Which political territory is Vansbro Municipality located in ? Dalarna County To which country does Strömstedt belong as its citizen ? Sweden
Possibilités de question générée	What country is she from ? **(Répétition)** Which competition did he participate in ? **(Mauvais pronom)**

TABLE 2 – Exemples d'erreurs lors de la génération de questions conversationnelles

avons pu constater que si notre détection d'entités échoue parfois à détecter des entités dans les textes, et signale donc trop d'entités non détectées, elle est raisonnablement performante pour les modèles qui ont le plus d'entités manquantes et peut donc servir à identifier de graves erreurs dans la verbalisation des entités.

Nous calculons également la corrélation de cette métrique avec les autres métriques automatiques et les résultats des évaluations humaines du challenge WebNLG. Nous montrons que si notre mesure est corrélée avec les résultats des évaluations humaines, le degré de corrélation varie en fonction des spécificités de la configuration de l'évaluation humaine. Cela suggère que pour mesurer l'adéquation des textes générés en fonction des entités, il peut être préférable d'utiliser une évaluation automatique telle que celle que nous proposons. Celle-ci pourrait en effet être plus fiable, car moins subjective et plus spécifique au problème de verbalisation des entités, que les mesures d'évaluation humaine.

Dans le chapitre 2, grâce à une métrique d'évaluation que nous développons et évaluons elle même de manière approfondie, nous observons que les modèles de verbalization de graphes RDF de WebNLG produisent des textes contenant des omissions et des hallucinations d'entités d'entrées. Nous remarquons en particulier que tous les modèles neuronaux et mêmes des modèles parmi les plus performants et récents comme ceux basés sur BART (Lewis et al., 2020) et T5 (Raffel et al., 2019) produisent ce type d'erreur.

RQ2 : Pouvons-nous détecter des problèmes liés au contenu, à savoir des omissions et des déformations dans les plongements des modèles encodeur-décodeur ? En particulier, pouvons-nous détecter ces problèmes dans les encodages des graphes d'entrée RDF ?
Dans le chapitre 3, nous proposons l'utilisation de deux sondes de classification afin de détecter les omissions et les déformations dans les encodages des graphes RDF de BART et T5. La première est non paramétrique, basée sur le calcul de similarité cosinus entre les encodages de graphes RDF. La seconde est quand à elle paramétrique utilisant des classifieurs binaires neuronaux.

D'abord, nous ajustons (en anglais "fine-tune") un modèle BART pré-entraîné sur les données WebNLG et la tâche de verbalisation de graphes RDF. Ensuite nous utilisons ce modèle afin de générer des verbalisations des graphes RDF de WebNLG et des graphes RDF de KELM. En utilisant l'outil d'annotation que nous avons développé au chapitre 2, nous annotons automatiquement les omissions dans les textes générés. Parallèlement nous recrutons trois annotateurs qui examinent manuellement les omissions et les déformations dans un sous-ensemble des textes annotés automatiquement. Enfin nous avons utlisé les deux types de sondes afin de à savoir si les omissions et les déformations des entités RDF peuvent être détectées dans leurs encodages, c'est-à-dire dans la sortie l'encodeur de BART. Le classifieur non paramétrique a obtenu une F-mesure de 0,68 et le classificateur paramétrique une F-mesure de 0,82. Nous observons que les résultats diffèrent pour les déformations et les omissions, ce qui suggère que les entités déformées et omises ne sont pas encodées de la même manière. Nous avons également montré que nos résultats s'étendent aux omissions dans un modèle T5, ce qui confirme notre affirmation selon laquelle notre méthodologie d'exploration est agnostique par rapport au modèle de génération encodeur-decodeur choisi.

En résumé dans ce chapitre 3, nous montrons que la plupart des omissions et des déformations peuvent être détectées dans les encodages de graphes RDF, ce qui suggère que l'encodeur est au moins partiellement responsable de l'apparition de ces erreurs. D'un point de vue méthodologique, nos résultats indiquent également que les sondes de classification peuvent être utilisées pour analyser les problèmes liés au contenu dans les encodeurs des modèles notamment ceux utilisant les transformeurs.

RQ3 : Comment pouvons-nous adapter les modèles NLG pour faciliter l'évaluation des problèmes liés au contenu ?

Dans le chapitre 4, nous examinons notre troisième question de recherche dans le contexte de la génération de questions conversationnelles pour les dialogues de tutorat à partir de données RDF. Inspirés par Narayan et al. (2021), nous proposons un modèle de génération de questions basé sur T5 qui génère à la fois un plan de la question à générer sous la forme d'un triplet RDF et la prochaine question du dialogue elle-même.

La génération d'un triple et de la question correspondante permet un processus d'évaluation en deux étapes. Tout d'abord, nous évaluons le triple généré en le comparant aux triples RDF d'entrée et au contexte de la conversation. Cette étape permet de vérifier à la fois l'adéquation sémantique avec l'entrée et la cohérence avec le contexte. Ensuite, nous évaluons la question en nous assurant qu'elle correspond au triple généré. La génération d'un triplet en plus de la question n'entrave pas le modèle NLG puisque sa performance par rapport à un modèle de base générant uniquement les questions ne diminue pas. Cette configuration permet cependant d'évaluer le contenu de la sortie de manière automatique et détaillée.

Listes de publications et rapports

Des parties de cette thèse se trouvent dans les articles suivants :

— Juliette Faille, Albert Gatt, and Claire Gardent. 2020. The Natural Language Pipeline, Neural Text Generation and Explainability. In 2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence, pages 16–21, Dublin, Ireland. Association for Computational Linguistics.

— Juliette Faille, Albert Gatt and Claire Gardent. 2021. Entity-Based Semantic Adequacy for Data-to-Text Generation. In Findings of the Association for Computational Linguistics : EMNLP 2021, pages 1530–1540, Punta Cana, Dominican Republic. Association for Computational Linguistics.

— Juliette Faille, Albert Gatt and Claire Gardent. 2023. Probing Omissions in Transformer-based RDF-to-Text Models, Juliette Faille, Albert Gatt and Claire Gardent, publication dans TACL.

— Juliette Faille, Quentin Brabant, Gwénolé Lecorvé, Lina Rojas Barahona and Claire Gardent. 2023. Question Generation in Knowledge-Driven Tutoring Dialog, en cours de soumission

En outre, nous avons contribué aux livrables suivants du projet européen NL4XAI :

— D2.1 Technical report on state-of-the-art end-to-end NLG systems, Michele Cafagna, Juliette Faille, Claire Gardent, Albert Gatt, 2020

— D2.3 Technical report on explainable NLG, Juliette Faille, Albert Gatt, Claire Gardent, 2021

— D2.4 Guidelines for explainable NLG evaluation, Michele Cafagna, Juliette Faille, Claire Gardent, Albert Gatt, 2022

— D2.6 Explainable NLG use case technical report, Juliette Faille, Claire Gardent, 2023

Contents

List of Figures

List of Tables

Introduction

Natural Language Generation (NLG) models are widely used, including by the general public, and daily interaction with these models is already happening on a large scale. Indeed the applications of NLG now touch upon many different fields such as customer services (with the use of chatbots in varied areas, e.g. travel, e-commerce, or insurance), aids for content creation with high-performance summarization and paraphrasing tools. Some Image-to-Text and Speech-to-Text applications also use and include NLG modules.

Having reliable NLG models and understanding their limitations is therefore a high-stakes challenge. Evaluation and Explainability are two approaches aiming at improving the quality and reliability of the models. In the following paragraphs, we introduce Explainability, Evaluation of NLG models, and the scope of this thesis.

Explainability of NLG models

The state-of-the-art models for NLG are deep learning models which can be characterized as "black boxes", i.e. models whose inner workings are not accessible or comprehensible to humans. Lack of explainability (XAI) is often seen as the main barrier to the practical deployment of Artificial intelligence (AI) (Barredo Arrieta et al., 2019). Explaining models is likely to increase performance by helping create more robust models and improve trust by making models more interpretable for users. From a legal perspective and following the EU AI Act (European Comission, 2021; EP), interpretability will be required for the deployment of AI systems in some applications, in particular, high-risk ones. Along these lines, (Madsen et al., 2022)'s survey on post-hoc explainability of neural NLP mentions four main motivations for explainability : ethics, safety, accountability, and scientific understanding. In this thesis, we use the definition of explainability and interpretability from Barredo Arrieta et al. (2019). An *interpretable model* is a model that is directly understandable by a human. An *explainable model* is a model which can be explained to a human using some kind of interface between the model and the human. As state-of-the-art NLG models are black-boxes and cannot be directly understood by humans, we focus in this thesis on explainability.

Evaluation of NLG models

NLG is the task of automatically generating a text. In most NLG applications, to be considered of good quality, an automatically generated text has to sound natural or fluent and have the intended meaning. Evaluation is the subfield of NLG that deals with judging the quality of generated texts. This research subfield has been very active in the last years and a lot of different evaluation metrics have been proposed as shown by the survey papers such as Gatt and Krahmer (2018), Sai et al. (2022) or Celikyilmaz et al. (2020) as well as by the recent GEM benchmark Gehrmann et al. (2021). Whereas fluency, naturalness, or readability measure whether the text is linguistically correct, content-related evaluation is connected with

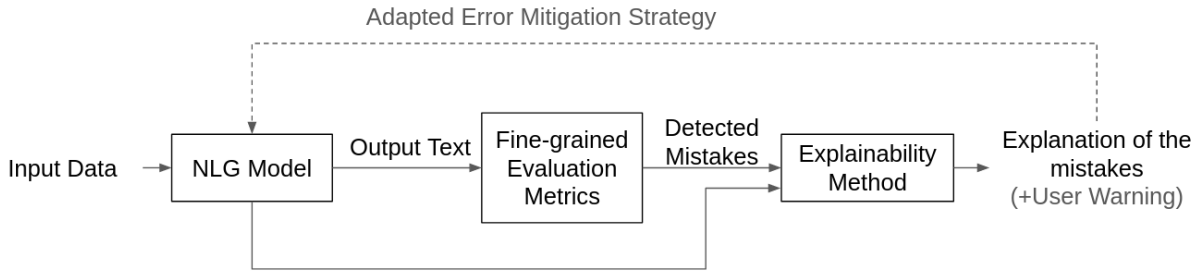


FIGURE 1 – Functional Diagram showing how evaluation metrics and explainability methods can be combined to improve the existing NLG models

semantics and assesses whether the meaning of the text is the intended one. Recent English pre-trained models have achieved very good performance for the fluency criteria. However, content-related issues are still present in some of the texts generated by state-of-the-art models. Content-related issues can be analyzed at different levels such as word, sentence, or discourse level, and encompass many different aspects of a text such as coherence, faithfulness to the input or usefulness in the context of a particular application. As the average quality of state-of-the-art models is very high, fine-grained evaluation and detection methods are required to identify the mistakes.

Link between Explainability and Evaluation of NLG models

In the context of an increasing need for trustworthy AI models, research towards making black box models understandable by users has flourished in the past years with many papers being published in the fields of explainability and interpretability. However, very little work has focused on NLG models. Other approaches aiming at improving the reliability of NLG models more broadly have been studied, namely the evaluation and the controllability of NLG models. In this thesis, we aim at creating links between the fields of XAI and the evaluation of NLG models. Indeed evaluation and explainability can be seen as complementary approaches to get more robust and understandable models. Evaluation can be used to detect and quantify mistakes and explainability to analyze and understand them. An example of how evaluation and explainability methods can interact is illustrated by Figure 1.

Scope of the thesis

In this thesis, we consider one subtype of NLG, the Data-to-Text task in which an output text is automatically generated from some input data. We will study two particular applications within Data-to-Text generation in English : the verbalization of Resource Description Framework (RDF) graphs and conversational question generation also using RDF graphs.

In the context of RDF verbalization, semantic inaccuracies include *hallucinations* (when the model generates information that is not in the input) and *omissions* (when some information is missing in the output text). We also define *distortions* which can be seen as an intermediary between hallucinations and omissions, when some input information is partially missing or incorrect. Examples of such mistakes are shown in Table 1.

In the context of conversational question generation, *dialog coherence* (coherence of the generated

question with the conversation context) and *semantic adequacy* with the input RDF also have to be verified. Content-related issues also include repetitions or contradictions between the question generated and the conversation context. Examples of mistakes are shown in Table 2.

Input RDF	(*Lady_Anne_Monson*, *birthPlace*, *Darlington*) (*Lady_Anne_Monson*, *birthDate*, *1726-01-01*) (*Lady_Anne_Monson*, *birthPlace*, *Kingdom_of_England*) (*Lady_Anne_Monson*, *residence*, *India*)
Output Text	Lady Anne Monson was born in England on January, 1st 1826, and lives in Germany.

TABLE 1 – Example of an RDF verbalization with hallucinations, omissions and distortions.

Research questions and Contributions

Our research goal is to investigate content-related issues in RDF-to-Text models in order to eventually solve or mitigate these issues. We address the following main research questions :

RQ1 : To what extent do state-of-the-art RDF-to-Text models have content-related issues, in particular omissions and hallucinations ?
To answer this first research question, we study the English RDF verbalizers submitted to the WebNLG Challenges 2017 and 2020 (Gardent et al., 2017; Castro Ferreira et al., 2020a). In the case of RDF verbalizers, one necessary condition for the generated text to be semantically adequate is that (i) all entities present in the input should be mentioned at least once in the output, and (ii) no other entity apart from those in the input is mentioned in the output. We refer to this requirement as entity-based semantic adequacy (ESA for short) and propose a metric based on automatically detected omissions and hallucinations in the outputs of the different models. Using this metric we quantify omissions and hallucinations made by the different models and compare them.

RQ2 : Can we detect content-related issues, namely omissions, and distortions in the embeddings of encoder-decoder models ? In particular, can we detect these issues in the encodings of RDF input graphs ?
We introduce two probing classifiers, a non-parametric and a parametric one, to detect omissions and distortions in the embeddings of RDF graphs in BART (Lewis et al., 2020) and T5 (Raffel et al., 2019). We show that indeed omissions and distortions can be detected in the encodings of RDF graphs, which suggests that the encoder is at least partially responsible for omissions and distortions. From a methodological perspective, our results also indicate that probing classifiers can be used to analyze content-related issues in the encoders of transformer-based models.

RQ3 : How can we adapt state-of-the-art NLG models to facilitate the identification and quantification of content-related issues ?
We examine this third research question in the context of conversational question generation for RDF-based tutoring dialogs. Inspired by Narayan et al. (2021), we propose a T5-based question generation model that generates a content plan in the form of an RDF triple together with the next question of the dialog. This setup allows for the evaluation of output content in an automatic and fine-grained manner.

Input RDF	(*Anna-Karin Stromstedt*, *country of citizen- ship*, *Sweden*) (*Anna-Karin Stromstedt*, *place of birth*, *Vansbro Municipa- lity*) (*Sweden*, *capital*, *Stockholm*) (*Vansbro Municipality*, *located in the administrative territorial entity*, *Dalarna County*) (*Vansbro Municipality*, *country*, *Sweden*) (*Anna-Karin Stromstedt*, *sport*, *biathlon*) (*Anna-Karin Stromstedt*, *participant in*, *2006 Winter Olympics*)
Dialog Context	In what city was Anna-Karin Strömstedt born ? Vansbro Municipality Which sport did she play ? Biathlon Which political territory is Vansbro Municipality located in ? Dalarna County To which country does Strömstedt belong as its citizen ? Sweden
Possibilities for Generated Question	What country is she from ? **(Repetition)** Which competition did he participate in ? **(Wrong pronoun)**

TABLE 2 – Example of possible mistakes in the conversational question generation

Thesis Outline

In Chapter 1, we introduce and define some of the background concepts we use in the thesis. We first present the two RDF-to-Text tasks we are working on, giving examples of datasets and models commonly used. We then explain how evaluation is done for these tasks in the literature, going from global metrics to fine-grained approaches. Finally, we introduce methods used to explain NLG models as well as a few commonly used methods in XAI for NLP models that could potentially be applied to NLG models.

In Chapter 2, we address our first research question and evaluate RDF verbalizers which took part in the WebNLG Challenges 2017 and 2020. We propose a fine-grained evaluation metric ESA to evaluate the semantic adequacy between the input RDF graph and the output verbalization.

In Chapter 3, we focus on our second research question and study how probing classifiers can detect omissions and distortions in the encodings of T5 and BART models fine-tuned for RDF verbalization.

In Chapter 4, we study our third research question and propose to generate both a RDF triple and its corresponding question to enable fine-grained automatic evaluation of the conversation coherence.

List of publications and reports

Parts of this thesis can be found in the following articles :

— Juliette Faille, Albert Gatt, and Claire Gardent. 2020. The Natural Language Pipeline, Neural Text Generation and Explainability. In 2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence, pages 16–21, Dublin, Ireland. Association for Computational Linguistics.

— Juliette Faille, Albert Gatt and Claire Gardent. 2021. Entity-Based Semantic Adequacy for Data-to-Text Generation. In Findings of the Association for Computational Linguistics : EMNLP 2021,

pages 1530–1540, Punta Cana, Dominican Republic. Association for Computational Linguistics.

— Juliette Faille, Albert Gatt and Claire Gardent. 2023. Probing Omissions in Transformer-based RDF-to-Text Models, to appear in TACL.

— Juliette Faille, Quentin Brabant, Gwénolé Lecorvé, Lina Rojas Barahona and Claire Gardent. 2023. Question Generation in Knowledge-Driven Tutoring Dialog, to be submitted.

Additionally, we contributed to the following NL4XAI project deliverables :

— D2.1 Technical report on state-of-the-art end-to-end NLG systems, Michele Cafagna, Juliette Faille, Claire Gardent, Albert Gatt, 2020

— D2.3 Technical report on explainable NLG, Juliette Faille, Albert Gatt, Claire Gardent, 2021

— D2.4 Guidelines for explainable NLG evaluation, Michele Cafagna, Juliette Faille, Claire Gardent, Albert Gatt, 2022

— D2.6 Explainable NLG use case technical report, Juliette Faille, Claire Gardent, 2023

1

Background

Contents

In this chapter, we introduce the concepts we use in chapters 2, 3 and 4 and mention some of the works related to this thesis.

In Section 1.1, we first define the RDF-to-Text task and the two subtasks we study, i.e. RDF graph verbalization and conversational question generation from RDF graphs. In Section 1.2, we describe (1) the three types of content-related issues we are studying for RDF graph verbalization : hallucinations, omissions, and distortions, and (2) issues we focus on in our conversational question generation experiments, such as repetitions and irrelevant information in a dialog context. We then present some commonly used metrics in Data-to-Text evaluation, from the most global scores to the most fine-grained evaluation processes. In Section 1.3, we summarize related work on explainability applied to NLG models and reflect upon the link between evaluation and explainability in NLG.

1.1 RDF-to-Text Generation

1.1.1 RDF data

RDF stands for Resource Description Framework (Schreiber and Raimond, 2014). It's a data representation coming from the semantic web, where data is organized in triples. Each triple contains two

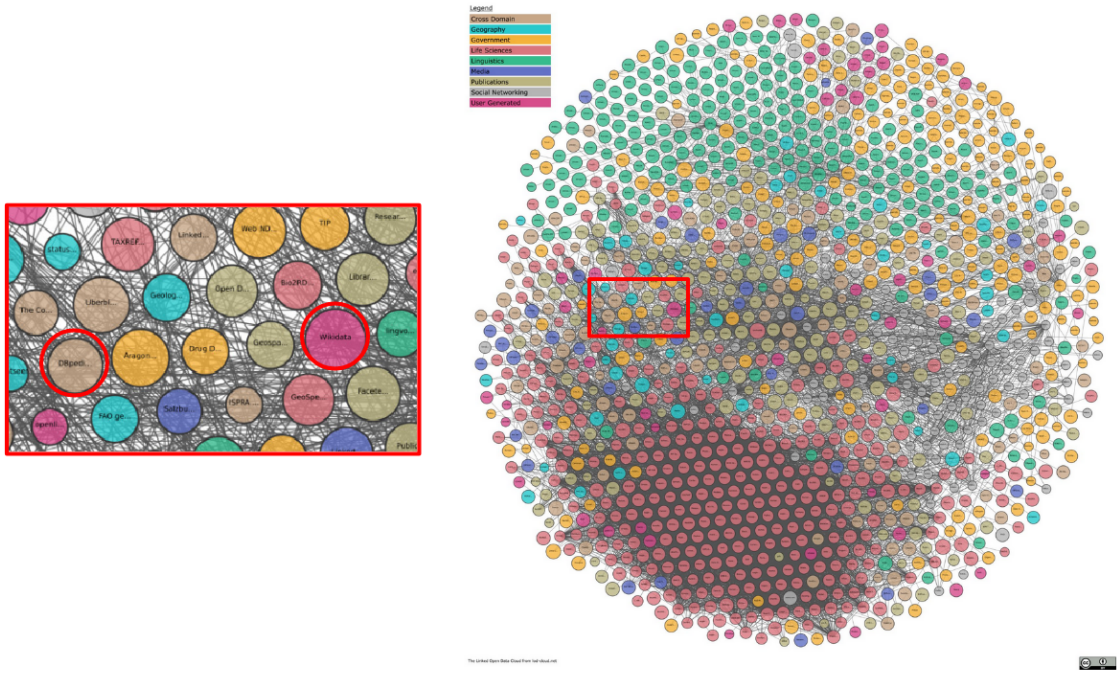


FIGURE 1.1 – Representation of the Linked Open Data Cloud (figure from http://cas.lod-cloud.net/clouds/lod-cloud-sm.jpg, accessed on July 10th, 2023) and zoom on Wikidata and DBPedia Datasets

entities, a subject s and an object o, which are linked by a property or predicate p. We denote such a triple (s, p, o). A triple can be seen as a component of a graph, where entities are nodes and the property is the edge connecting them. An RDF graph (or RDF tripleset) consists of a set of triples, which we denote $((s_i, p_i, o_i))_{i \in \mathbb{N}}$.

Examples of RDF databases are DBPedia (Lehmann et al., 2015) (in 2014, DBPedia contained 3B triples out of which 600K were extracted from the English Wikipedia) and Wikidata (Vrandečić and Krötzsch, 2014) (which currently contains 1.2B RDF triples). In this thesis, we mostly use data coming from Wikidata and DBPedia. However, there are also multiple other RDF stores. Famous examples of RDF stores are FOAF (Friend of a Friend) (Brickley and Miller, 2000) which contains relations between people and activity and models social networks data, or GeoNames (Wick, 2015) which is composed of geographical data. In total, RDF stores contain a huge quantity of data. These RDF knowledge stores are interlinked and constitute together the so-called Linked data. Thanks to its common structure, this extensive quantity of data can be browsed and queried efficiently. Figure 1.1 situates DBPedia and Wikidata in the context of the Linked Data and gives an idea of the quantity of available data. Creating models to automatically verbalize this data opens perspectives on making this data easily accessible to users.

1.1.2 A subtask of Natural Language Generation

Natural Language Generation (NLG) is the task of automatically generating natural language. We can distinguish between two main categories of tasks : Text-to-Text and Data-to-Text tasks. Text-to-Text applications are for instance Summarization, Paraphrase, Discourse simplification, or Story generation. Examples of Data-to-Text subtasks are generation from numerical data (such as from weather, or health

KELM tripleset of length 5	(*Than Tun*, *field of work*, *History of Myanmar*) (*Than Tun*, *instance of*, *Human*) (*Than Tun*, *employer*, *University of Michigan*) (*Than Tun*, *sex or gender*, *male*) (*Than Tun*, *educated at*, *University of Yangon*)
WebNLG tripleset of length 2	(*Amatriciana_sauce*, *region*, *Lazio*) (*Amatriciana_sauce*, *ingredient*, *Pecorino_Romano*)

TABLE 1.1 – Example RDF triplesets from KELM and WebNLG

data), from tabular data, from images (for example in the image captioning task), or generation from semantic representations such as Meaning Representations (MR) or RDF graphs.

There are multiple datasets for English Data-to-Text generation. Some of them are of relatively small size and focused on specific domains such as the E2E dataset (Novikova et al., 2017b) which is crowdsourced and contains 50K (MR, Text) pairs describing restaurants.The RotoWire dataset (Wiseman et al., 2017) and its corrected version Sportsett :Basketball (Thomson et al., 2020) contain 5k samples of NBA basketball games records and their summaries in natural language. The Weathergov dataset (Liang et al., 2009) has 22k pairs of tables with weather information and texts of weather forecasts. The CACAPO dataset (van der Lee et al., 2020) contains 10k English (MR, Text) pairs with sports, weather, stocks, and incidents topics. Contrary to the previously mentioned datasets, in the CACAPO dataset, texts were first collected from press articles, which were then annotated with sets of attribute-value pairs. Authors claim that thanks to this data collection process they can ensure more text diversity and naturalness compared to datasets where texts are written to verbalize pre-collected data. The WITA dataset (Fu et al., 2020) is composed of 55k pairs of Wikipedia sentences and automatically parsed Wikidata triples.

Some larger-scale datasets are based on Wikipedia. Wikibio (Lebret et al., 2016) contains 700k instances pairing Wikipedia infoboxes and the first paragraph of the corresponding article. Wikigen (Perez-Beltrachini and Lapata, 2018) is based on Wikibio and contains 200k pairs of list of property-value pairs and Wikipedia abstracts. GenWiki (Jin et al., 2020) contains 1.3M (Knowledge graphs, Text) pairs automatically collected from Wikipedia. The ToTTo dataset (Parikh et al., 2020) has 120k examples and provides pairs of Wikidata tables with some highlighted cells together with their one-sentence natural language description.

In this thesis, we work with RDF data and consider two applications within RDF-to-Text : the verbalization of an input graph and knowledge-grounded conversational question generation. We also limit ourselves to the generation of texts in English.

Verbalization of RDF graphs RDF verbalization is the task of creating a text that expresses the meaning represented in an RDF graph. It involves determining the order in which the different triples of the graph will be articulated. It potentially requires the use of referring expressions, coreferences, anaphora, ellipses, etc. To create an NLG RDF verbalization model, these transformations must be either explicitly listed (e.g. in rule-based or template-based models) or implicitly learned (e.g. in statistical or neural models). Recent NLG RDF verbalization models are mostly neural models which are typically trained on datasets of (RDF, text) pairs.

In this thesis, we are working with two datasets of (RDF, text) pairs : the WebNLG dataset (Gardent et al., 2017; Castro Ferreira et al., 2020a) and the KELM dataset (Agarwal et al., 2021). KELM is a large synthetic dataset of around 8M (RDF, text) pairs. The WebNLG dataset is much smaller, with 47k (RDF-Text) pairs, but also less noisy as the texts were manually written to match the input graph.

Tripleset	(*Albennie_Jones*, *activeYearsStartYear*, *1930*) (*Albennie_Jones*, *genres*, *Rhythm_and_blues*) (*Rhythm_and_blues*, *stylisticOrigins*, *Blues*) (*Rhythm_and_blues*, *derivatives*, *Disco*) (*Albennie_Jones*, *birthPlaces*, *United_States*)
Verbalization	Albennie Jones was born in the United States and started performing rhythm and blues in 1930. Rhythm and blues originated from blues and disco music is one of its derivative.

TABLE 1.2 – Example of verbalization of an RDF triple (from the WebNLG dataset)

Examples of triplesets from WebNLG and KELM are shown in Table 1.1. The choice of the WebNLG dataset is quite natural as it is one of the main benchmarks for RDF-to-Text generation. This dataset was created for a shared task, the WebNLG Challenge (Gardent et al., 2017; Castro Ferreira et al., 2020a), which happened in 2017 and 2020. One task of the challenge is English RDF-to-Text. As the results of the WebNLG Challenges 2017 and 2020 are publicly available we could access all of the participant's results on the test set of the WebNLG data and compare them (in Chapter 2). The fact that this dataset is clean, given that RDF verbalizations were manually written, is a decisive factor for our study in chapter 3. We want indeed to detect the omissions and distortions of the model, instead of mistakes coming from finetuning on potentially misaligned data. GenWiki and Wikibio would also have been interesting datasets to use but are known to contain some misalignment (as shown for instance by Perez-Beltrachini and Gardent (2017)) as they were automatically collected from Wikipedia. To add more diversity and out-of-domain data, we choose to use the KELM dataset which is diverse in terms of domains and entities.

Conversational Question Generation from RDF graphs Conversational question generation from RDF graphs is the task of generating a question based on a dialog context and some RDF graph. The generated question has to be fluent, grammatical, and coherent with the dialog context and with information from the RDF input graph. This task can be seen as a verbalization task (as described in the previous paragraph) with two additional challenges : (i) content selection and (ii) coherence with the dialog context. Indeed, as a question corresponds to a single RDF triple and questions are generated one at a time, the NLG model needs to select one triple from the RDF input graph ; and then turn it into a question. The question needs to be coherent with the dialog context, i.e. the previously generated questions and corresponding answers. Coherence includes the absence of repetitions of previously generated questions and a logical progression from the previously generated questions, but also generating anaphora such as correct pronouns. Examples of datasets for conversational question generation from RDF data include CSQA (Saha et al., 2018) (200k dialogs), and the KGConv dataset (Brabant et al., 2023) (70k dialogs). In Chapter 4, we consider a particular application of conversational question generation, the generation of tutoring dialogs. An example of such a dialog in given in Table 1.3.

Note that an NLG common and related task is Conversational Question Generation from a text. Instead of using an RDF graph as a source of information about which questions should be asked, some approaches use text as input (Shen et al., 2021; Gu et al., 2021).

Conversational Question Generation (either from input structured data or text) has multiple real-world applications. As mentioned by (Gu et al., 2021) it can be used in goal-oriented dialog generation tasks such as automatic personal assistants or customer services. It can also be used in the education domain for creating learning tutoring systems, which is the application our work in chapter 4 relates to. Du et al. (2017) introduce a question generation model to automatically generate reading comprehension quizzes.

Triple :	(*Battle of Dornach*, *country*, *Switzerland*)
Question :	In what country is Battle of Dornach located ?
Answer :	Switzerland
Triple :	(*Battle of Dornach*, *part of*, *Swabian War*)
Question :	What process was this part of ?
Answer :	Swabian War
Triple :	(*Switzerland*, *named after*, *Schwyz*)
Question :	What is Switzerland named after ?
Answer :	Schwyz
Triple :	(*Switzerland*, *driving side*, *right*)
Question :	Which direction is the driving side in Switzerland ?
Answer :	right
Triple :	(*Switzerland*, *located in*, *Central Europe*)
Question :	In what geographic region is Switzerland located ?
Answer :	Central Europe
Triple :	(*Switzerland*, *lowest point*, *Lake Maggiore*)
Question :	What is the lowest geographic point of Switzerland ?
Answer :	Lake Maggiore

TABLE 1.3 – Example of a dialog from the KGConv dataset

In their survey paper Kasneci et al. (2023) mention the automatic generation of problems and quizzes as one of the main opportunities of modern dialog NLG systems to help students to better understand and retain what they are trying to learn. They also mention the possibility to create more personalized questions and therefore match better the needs and learning goals of each student or group of students.

1.1.3 Models

In this subsection we first describe models used in state-of-the-art NLG. We distinguish between two main types of model architectures, encoder-decoder and decoder-only models. We briefly present the pretraining and finetuning optimization strategy. We then introduce in more detail two of the models we use in our experiments in chapters 3 and 4, BART (Lewis et al., 2020) and T5 (Raffel et al., 2019). We finally briefly review commonly used models for the RDF verbalization and Conversational Question Generation tasks.

1.1.3.1 State-of-the-art NLG Models

Encoder-decoder vs Decoder-only Models State-of-the-art models for NLG tasks use the Transformer model architecture from Vaswani et al. (2017). They have either decoder-only (for causal models like for instance models from the GPT family, Radford et al. (2018, 2019b); Brown et al. (2020)) or encoder-decoder architecture. Both generate the output text autoregressively. The main difference between decoder-only and encoder-decoders is the way the input is processed. In decoder-only models, the input and output

Decoder Only		Encoder-Decoder	
GPT 2 (Radford et al., 2019a)	1.5B	T5	60M-11B
LLaMA (Touvron et al., 2023)	7-65 B	FLAN-T5 (Chung et al., 2022)	80M-11B
GPT 3 (Brown et al., 2020)	175B	mT5 (Xue et al., 2021)	300M-13B
BLOOM (Workshop et al., 2023)	176B	BART	406M
PaLM (Chowdhery et al., 2022)	540B	UL2 (Tay et al., 2023)	20B

TABLE 1.4 – Number of parameters for different decoder-only and encoder decoder models

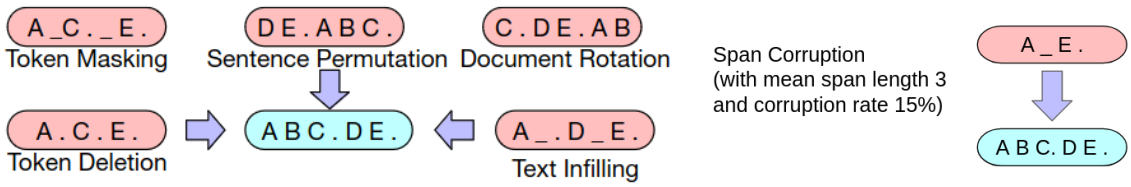
are concatenated and passed through the same layers, such that parameters are shared for the processing of the input and output. In encoder-decoders, the input first goes through the encoder, then the output is generated based on the encoding of the input and the previously generated output tokens. The decoder attends to the full encoded input thanks to the cross-attention mechanism. As the encoder and decoder do not share their parameters, the parameters of the encoder can be more specific to the input. In theory, this would also imply that encoder-decoders have two times more parameters than the decoders-only. However, in practice, state-of-the-art decoder-only models are much larger (see some examples in Table 1.4).

In the literature, decoder-only models tend to be preferred for prompt-based and open-ended text generation and encoder-decoders for other sequence-to-sequence tasks (Tay et al., 2023). For the specific task of RDF-to-Text, there is currently a debate in the research community about which of the two architectures performs best. In their experiments, Raffel et al. (2019) found that the encoder-decoder worked best. Fu et al. (2023) also claim that decoder-only models can be subject to the so-called "Attention Generation Problem", i.e. for longer generated outputs, the attention to the input becomes relatively lower. This would mean that the decoder-only models' outputs are less strongly conditioned on the input than encoder-decoders' outputs. Their experiments on the WebNLG, E2E, and WITA (Fu et al., 2020) datasets show that their encoder-decoder model slightly outperforms their decoder-only baseline on automatic evaluation metrics. Andreas et al. (2021) also show that T5 and BART outperform GPT-based models on the restaurant recommendation dialog generation task.

Pretraining and Finetuning Data-to-Text transformer-based models are typically trained in two steps using pretraining and then finetuning [1]. Pretraining consists in training the model in an unsupervised way on a large quantity of textual data and aims at building a representation of the language. Fine-tuning is done on a much smaller quantity of data and consists in training the model to perform a particular downstream task, in our case RDF-to-Text generation.

Multiple pretraining objectives have been explored. The language modeling objective, i.e. predicting the next token of a sequence, has been used first (e.g. by Dai and Le (2015) or Radford et al. (2018)). Devlin et al. (2019) introduces the masked language modeling (MLM) objective for the pretraining of BERT. MLM consists in randomly masking input tokens and learning to predict them. They also use the Next Sentence Prediction objective, which trains the model to predict if two sentences follow each other (by replacing half of the time the second sentence with a random sentence). Whereas language modeling and masked language modeling aim to learn linguistic representations at word- or token-level, next sentence prediction intends at making the model learn sentence- or discourse-level linguistic representations. Other examples of token-level objectives are token order permutations (Yang et al., 2019) or replaced token detection (Clark et al., 2020). In SpanBERT, Joshi et al. (2020) extend the BERT MLM objective to mask entire spans of tokens and learn to reconstruct them. At sentence level, Lan et al. (2020) use sentence-order prediction, i.e. they train the model to predict if two consecutive sentences are in the correct order or have

1. For very large models, usually decoder-only, there is in general no finetuning, since zero-shot setups seem to perform well.



(a) BART (Image from (Lewis et al., 2020) (b) T5

FIGURE 1.2 – Pretraining objectives of BART and T5

been swapped. (Iter et al., 2020) train their model to predict sentences that are k sentences apart from an anchor sentence.

1.1.3.2 Models used in the RDF Verbalization task

In the 2020 WebNLG challenge (Castro Ferreira et al., 2021) most of the participants (11 out of 14) used either BART, mBART or T5. One participant used the LASER [2] model (Artetxe and Schwenk, 2019) and two participants used template-based models. In the 2017 WebNLG Challenge (Gardent et al., 2017), three submissions used templates and grammar-based systems, four submissions used attention-based encoder-decoders, and one submission used a statistical machine translation system.

Remark. *Although some approaches, such as (Ribeiro et al., 2019), use graph encoders to keep the structure of the RDF graph in the encoding, in most of the state-of-the-art approaches, the RDF input is linearized, which allows the use of sequence-to-sequence models. In practice, RDF-to-Text is therefore very similar to a Text-to-Text task, even though it should be noted that the linearization may change the distance between semantically related parts of the RDF input.*

1.1.3.3 Models used in the Conversational Question Generation task

Dialog generation models can be divided into two broad categories : task-oriented and open-domain. Open-domain dialog usually focuses on chitchat and aims at generating a natural and engaging conversation. Task-oriented dialogs aim to achieve a given goal, such as providing the user with specific information or helping them to take a decision on a particular topic. For instance, Budzianowski et al. (2018) include seven tasks such as restaurant, hotel, or train booking. In this thesis (Chapter 4), we consider a particular subtask of task-oriented dialog generation : conversational question generation.

The most popular approach for task-oriented dialog generation used to decompose the task into 3 submodules (Young et al., 2010) : (1) a Natural Language Understanding module (to understand the dialogue context and the user request), (2) a Dialog management module (to decide which answer to give), and (3) a Natural Language Generation module (to verbalize the answer and communicate it to the user). Recently state-of-the-art approaches tend to use end-to-end pretrained models (Bordes et al., 2017) such as DialoGPT (Zhang et al., 2020), or T5 and BART (Andreas et al., 2021).

1.1.3.4 BART and T5 models

We use BART and T5 for experiments in Chapters 3 and 4. They both use the transformer architecture and are encoder-decoder models. The pretraining of BART is done using reconstruction of corrupted

2. The LASER model is a BiLSTM encoder-LSTM decoder model trained to learn multilingual sentence embeddings in 93 languages.

texts, e.g. token masking, deletion, sentence permutation or document rotation (denoising autoencoder). The pretraining of T5 is done through span corruption. Raffel et al. (2019) find that the best results are obtained by randomly masking 15% of the input sequence and having masked spans of an average length of 3 tokens. The pretraining objectives of BART and T5 are illustrated on Figures 1.2a and 1.2b. The BART and T5 models are both available in different sizes. The BART-base version has 140M parameters and 6 layers in each of the encoder and decoder, whereas BART-large has 400M parameters and 12 layers in each of the encoder and decoder. T5-small (60M parameters) has 6 layers in encoders and decoders, T5-base (220M parameters) has 12, and T5-large (770M parameters) 24. BART was pretrained on the same dataset as the RoBERTa model (Zhuang et al., 2021), using a 160 GB corpus of books, Common crawl news and stories, web texts and the English Wikipedia. For T5, Raffel et al. (2019) created a 750 GB cleaned version of the Common crawl dataset, the C4 dataset (Colossal Clean Crawled Corpus). T5 was pretrained over $2^{19}(\approx 524k)$ steps and with a $2^{16}(\approx 66k)$ batch size, i.e. on $2^{35}(\approx 34B)$ tokens, whereas BART was pretrained over 500k steps with a batch size of 8k, i.e. on 4B tokens.

1.2 Content Evaluation

State-of-the-art NLG models are generally considered very good at producing fluent outputs. However these outputs still sometimes contain content-related mistakes. In subsection 1.2.1, we introduce content-related issues in the tasks of RDF verbalization and RDF conversational question generation. In subsection 1.2.2, we introduce the research field of NLG evaluation and some of its main concepts and subcategories. In subsection 1.2.3 we then review evaluation techniques allowing to assess the semantics of a text. We organize the different evaluation approaches in three main categories : global evaluations of the overall quality of a text (1.2.3.1), evaluation approaches giving one global score for the semantic quality of a text (1.2.3.2) and finally more fine-grained evaluations of a text's content (1.2.3.3). Finally, we situate our contributions with respect to these different categories (1.2.4).

1.2.1 Content related Issues

In Verbalization The state-of-the-art models for RDF-to-Text (or more generally Data-to-Text) have very high performance in terms of fluency. The main mistakes these models make are content-related. By definition of the task, the text content is supposed to be the same as the input graph. Two main types of mistakes have been found to occur : *omissions* (some input information is not mentioned in the output text) and *hallucinations* (some information not present in the input graph is mentioned in the text). In our experiments in Chapter 3, we also found another type of mistake which we name *distortion*. Distortions are text spans that are close to correspond to some RDF entities or properties but not exactly matching. Examples of distortions are quantities with wrong units of measurement, proper names with prominent spelling mistakes, or dates that are partly incorrect (e.g. the day and month are correct, but not the year). They can be seen as partial omissions or hallucinations. Examples of hallucinations, omissions and distortions are shown in Table 1.5.

Among hallucinations, omissions and distortions, the most widely studied mistake type in the literature is hallucinations. (Ji et al., 2023)'s survey of hallucinations mentions different potential causes of hallucinations in NLG models. According to this survey, the main cause for hallucinations is in the *training data.* They can indeed result from discrepancies between the source information and the output text found in noisy datasets, or from datasets biases (duplicates or frequent data fragments will be more likely to be generated). The other cause of hallucinations they mention is the *model* itself. Indeed the architecture and the training procedure of the model can make it more prone to hallucinate during inference. A flawed encoder that learns low-quality representations of the input data is a first line of

Input Tripleset	(*San_Francisco*, *areaCode*, *628*)
Output Text	The area code for San Francisco is 657.
Input Tripleset	(*Bedford_Aerodrome*, *cityServed*, *Bedford_Autodrome*)
Output Text	The city of Birmingham is served by the city of bedford autodrome.
Input Tripleset	(*Liselotte_Grschebina*, *birthPlace*, *Karlsruhe*) (*Liselotte_Grschebina*, *nationality*, *Israel*) (*Liselotte_Grschebina*, *training*, *School_of_Applied_Arts_in_Stuttgart*) (*Karlsruhe*, *country*, *Germany*) (*Israel*, *language*, *Modern_Hebrew*)
Output Text	Lieselotte Grassschebina was born in Karlsruhe, Germany and attended the School of Appleton Arts in Stuttgart. She is a national of the country of Tel Aviv where one of the languages spoken is Russian.

TABLE 1.5 – Examples of automatically generated texts containing hallucinations, omissions and distortions

research. They mention also problems in the decoder's attention to the encoding, namely attention to the wrong part of the input. The decoding strategy can also be responsible for unfaithfulness. This is also what has been shown recently by Wan et al. (2023), who claim that nucleus sampling decoding generated more hallucinations in abstractive summarization and propose two new decoding strategies to mitigate hallucinations. Pre-training leads to some memorization of pre-training data (e.g. McCoy et al. (2023)). When the model favors the memorized knowledge over the input this can create hallucinations.

Thus, Ji et al. (2023) survey hallucinations in different NLG tasks. They mention two types of hallucinations : *intrinsic* (i.e. information in the output that contradicts the input) and *extrinsic* (i.e. information in the output that cannot be found in the input). Maynez et al. (2020) also outline that hallucinations can be *factual* (i.e. information that is not found in the input but verified in world knowledge) and therefore distinguish between factuality and faithfulness.

Faithful means correct with respect to the input. *Factual* means correct with respect to world knowledge. In this thesis, we limit ourselves to the study of faithfulness and exclude factuality for three main reasons. First, factuality is not really relevant for the tasks we study. Indeed, by definition of the RDF verbalization task, factual hallucinations are considered mistakes. As verbalization should contain only information from the input RDF graph, if some other information has been added, it is considered incorrect with respect to the task definition, independent of whether it is factual. Second, from a practical perspective, evaluating factuality would require having access to world knowledge and is much more costly if not in some cases unfeasible. Finally, as our main motivation is to improve the transparency and reliability of the NLG models, accepting factual hallucinations would mean relying on world-knowledge acquired by the model during pretraining, even though we have very little control over the pretraining data and on how models use it.

In Conversational Question Generation Huang et al. (2020b) identify three challenges in dialog

generation : *semantics*, *consistency* and *interactiveness*. Interactiveness is rather related to the form and less to the content of the generated dialog. We, therefore, focus on semantics and consistency. What makes generation from RDF graphs in a dialog different from verbalization of an RDF graph is the presence of the dialog context. The dialog turn to be generated has to be coherent with the previous dialog turns, introducing some new information and avoiding repetitions but also connecting the new information to the previous turns in a logical way. Common content-related issues in conversation question generation include lack of specificity of the generated question, repetition of information already mentioned, and lack of coherence with the conversation context or with the input data. Note that the models are also subject to content-related issues that concern all RDF-to-Text models such as in particular to hallucinations, omissions, or distortions of input information.

1.2.2 Evaluation concepts in NLG

In the following paragraphs, we introduce the topic of NLG evaluation and mention some commonly used concepts and categories of evaluation in NLG that we will be referring to in this thesis.

Why is NLG Evaluation a complicated task ? Evaluating the quality of an NLG model requires looking at the texts it produces and determining whether they are linguistically correct and have a correct meaning and style. This requires being able to check linguistic properties, such as grammaticality or syntax but also knowing what the NLG task is and determining whether the text satisfies its requirements. For instance, in summarization, evaluating a model's output requires determining if it is a valid summary of the input text. In Data-to-Text, it requires checking that the text correctly verbalizes the input data. Evaluating a text requires sometimes background or world knowledge and a thorough understanding of the system's final user preferences (e.g. in terms of type or level of language to use). It is inherently a task-dependent process, and for complex tasks for which the requirements are hard to define (for instance in automatic creative writing), it can also be highly subjective.

Automatic vs Human Evaluation of NLG output texts can be done either automatically by computing one or multiple metrics, or manually by asking humans to evaluate the text. *Automatic evaluation* has the advantage of being cheaper and easily scalable to multiple models or more outputs. It can also be used iteratively during a new model's development. However, it also only partially evaluates texts by focusing on single aspects of them and has been sometimes found to correlate poorly with human judgments. On the contrary, *human evaluation* supposedly provides higher quality evaluation and a more comprehensive view of the text's quality. However, human evaluation also has multiple disadvantages. Apart from being expensive and time consuming, it also lacks standardization (Howcroft et al., 2020). Indeed multiple criteria can be used and the recruitment of the annotators can differ greatly (e.g. annotations can be collected via a crowdsourcing platform such as Amazon Mechanical Turk, or field experts can be recruited to annotate the texts). This diversity in how human evaluation is carried out in practice makes models' results difficult to compare.

Global vs Fine-grained Some model evaluations only include global evaluation metrics which give one score supposed to reflect the overall quality of the text. Other approaches use different metrics or criteria to evaluate different aspects of the text. We describe in more detail global and fine-grained evaluations in the following subsection 1.2.3.

Reference-based vs Reference-less In order to automatically evaluate a generated text, it can be compared to a gold standard or reference text. The more similar the generated text is to the reference

text, the better it is considered to be. Whereas *reference-based evaluation* is conceptually simple and can be convenient to compute, it has several disadvantages. First, a dataset of input and reference pairs needs to be created which can be expensive. More importantly, the quality of the resulting evaluation is not necessarily satisfying as a reference text is usually not the only possible output. In other words, a generated text can be also a good and valid output but if it is not similar to the reference, it will receive a low score. Some datasets and approaches have tried to mitigate this issue by providing multiple references instead of a single one. Even though having multiple references can help evaluate more diverse outputs, it does not fully solve the problem as the set of references does not generally account for all possibilities of generations.

A way to avoid this problem is not to use references at all. Indeed a way to evaluate a generated text can be to compare it directly to the input. This is the approach used by Scialom et al. (2021), who use a question-answering model to check if all facts of a summary are present in the input document. Rebuffel et al. (2021b) proposed to extend this approach to data-to-text, by first building a synthetic multimodal corpora for training the question-answering model. Also in data-to-text, Dušek and Kasner (2020) use a natural language inference (NLI) model to check if the input and output entail each other. Input-based evaluation is also the approach we use in Chapter 2 where we compare entities mentioned in the output text to entities of the RDF input graph.

Surface-based vs Embedding-based Traditional NLG evaluation metrics such as BLEU (Papineni et al., 2002) or ROUGE (Lin, 2004) are surface-based and compare n-grams in the generated text to n-grams in the reference texts. A major limitation of these metrics is that they penalize all kind of paraphrase. Instead of comparing n-grams, later approaches were proposed to compare embeddings of the references and the generated texts by computing their similarity in the embedding space. BERTscore (Zhang* et al., 2020a) for instance uses the contextualized embeddings of BERT and computes the cosine similarities between the embeddings of the reference and generated texts. Kane et al. (2020) propose the NUBIA metric which is based on an aggregation of scores - semantic similarity, logical entailment and sentence intelligibility - themselves computed using RoBERTa and GPT-2.

Challenges for the evaluation in NLG Evaluation in NLG is confronted with different challenges. Open research questions include for instance the evaluation of evaluation metrics and the worst-case evaluation of models.

- **Evaluation of evaluation :**

Finding which evaluation metrics or criteria are the most reliable is the topic of multiple studies. A classical way to evaluate a metric is to check whether it correlates well with human judgments. This is a commonly used approach to validate a new metric. Some approaches also propose to reevaluate in a careful and consistent way already existing evaluation metrics. SummEval Fabbri et al. (2021) reevaluates 14 automatic metrics used for text summarization. Bhandari et al. (2020) also reevaluate metrics in the context of summarization and finds that the best metric differs depending on the specificities of each summarization task, in particular the chosen dataset. Sai et al. (2021) propose checking if 25 commonly used NLG evaluation metrics are robust to perturbations of the output text (such as for instance gender changes, or adding repetitions) and correlate with 18 human evaluation criteria. They find that most metrics do not correlate well with the human evaluation criteria and do not detect perturbations.

- **Worst-case vs Average-case evaluation :**

Novikova et al. (2017a); Reiter (2017) mention the fact that NLG evaluation metrics are all about an average-case evaluation and not worst-case evaluation. Indeed NLG models are evaluated by looking at some of their outputs and computing average evaluation scores on this subset of outputs. Instead,

worst-case evaluation would consist in determining the worst possible text a model could output. Whereas worst-case evaluation is in practice much more difficult to perform than average-case evaluation, it is crucial for the industrial deployment of NLG models, in particular for high-risk applications. Even though worst-case evaluation is, for now, impracticable for state-of-the-art NLG models, it is an interesting research direction, as evaluating the worst-case performance of a model and detecting failure cases is important to develop trustworthy models. This is in line with papers advocating for a better report of NLG models' errors and failure cases (e.g. van Miltenburg et al. (2021)).

Having introduced the main concepts related to evaluation we will refer to in this thesis, we will now describe approaches evaluating the content of texts generated by NLG models.

1.2.3 Evaluation approaches : from global to fine-grained

As stated in Sai et al. (2021), the way humans evaluate a text is a complex process involving different decisions and based on criteria such as "fluency, adequacy, coherence, informativeness, engagingness, consistency, etc". This is a difference with how evaluation is usually done in NLG, with most of the automatic metrics giving one global score for an output text, and most of the human evaluation approaches evaluating texts only along a few criteria. Sai et al. (2021) do perturbations on the outputs of different models of different NLG tasks. For Data-to-Text examples of perturbations include perturbation of numbers, of named entities, or deletion of phrases. They show that most of the 25 automatic metrics they study are not sensitive to these perturbations, and none is sensitive to all. An approach to overcome this limitation as advocated by Sai et al. (2021) but also by van Miltenburg et al. (2021), Thomson and Reiter (2020) or Gehrmann et al. (2021) is to move from global evaluation metrics to more fine-grained evaluation procedures.

In this section, we first briefly describe the main global metrics used in Data-to-Text generation. We then introduce approaches focusing on the evaluation of some specific aspects of the NLG outputs, namely the ones related to semantic adequacy and dialog coherence. Finally, we mention some of the more recent approaches aiming at providing fine-grained evaluation both in automatic and human evaluation.

1.2.3.1 Global Evaluation of NLG outputs

Commonly-used metrics use references and are surface-based (either word- or character-based) metrics such as BLEU (Papineni et al., 2002), TER (Snover et al., 2006) or chrF (Popović, 2015). GLEU (Wu et al., 2016), which we use in chapter 4 is similar to the BLEU score but computes the minimum of recall and precision of matching n-grams between the output text and the reference. As mentioned earlier, these methods fail to account for paraphrases, penalizing good generated texts that are worded differently as the references. Therefore alternative metrics have been proposed such as METEOR (Banerjee and Lavie, 2005), which measures n-gram overlap but integrates synonyms and BERTscore, a trained metric based on word-embedding similarity (Zhang* et al., 2020b). Rather than comparing generated and reference text, other work has focused on developing metrics which model human judgment . Thus (Sellam et al., 2020) introduced BLEURT, an automatic metric pre-trained on synthetic and automatically rated data and fine-tuned on human judgments.

In addition to these global metrics, some metrics aiming at measuring specific features of the NLG outputs have been developed. This is detailed in the following subsection.

1.2.3.2 Towards more aspect-oriented metrics

Semantic Evaluation In this paragraph, we describe how the evaluation of the content of a generated text is done in the NLG literature. We first describe automatic evaluation metric and then human evaluation approaches.

Automatic Semantic Evaluation One way to evaluate the semantics of a generated text is to compute the semantic similarity between this text and one or multiple references, i.e. using *Reference-based evaluation*.

In machine translation for instance, the MEANT metric (Lo and Wu, 2011) applies Semantic Role Labelling to generated and reference text and computes similarity by matching the resulting semantic frames. Similarly, in image captioning, SPICE transforms both generated and reference captions into a scene graph encoding the objects and relations present in these captions and computes an F-score over the semantic propositions in the scene graph Anderson et al. (2016). In Data-to-Text generation, Dhingra et al. (2019) use custom entailment models to determine whether an n-gram in the generated text is entailed by the input and the reference and computes an F-score based on these n-grams. The Data-QuestEval metric Rebuffel et al. (2021b) checks using a question-answering model that the output text information matches the input data. In text summarisation, Goodrich et al. (2019) compare relation tuples extracted from a ground-truth summary and a generated one using either a Named Entity Recognition and a Relation Classifier or an end-to-end Transformer model to extract these tuples.

Another way to evaluate the semantics of the generated output is not with respect to the reference or to human judgments, but with respect to the input, i.e. using *Input-based evaluation*.

Wiseman et al. (2017) define Relation Generation score as the precision of input relations found in the output texts (the relation extraction is performed by a neural model). (Reed et al., 2018) define information extraction patterns to measure the occurrence of the input attributes and their values in the outputs and compute semantic adequacy using the Slot Error Rate. Ribeiro et al. (2020); Dušek and Kasner (2020) uses natural language inference (NLI) to detect two-way entailment between the generated text and the input. Sulem et al. (2020) introduce SAMSA, which assesses simplification quality by comparing the predicate/argument structures contained in the input with those contained in the output summary. Rohrbach et al. (2018) propose the CHAIR metric which computes proportions of hallucinated objects in image captions generated on the MSCOCO dataset (Lin et al., 2014).

Human Semantic Evaluation Some approaches evaluate the semantics of the output text using a single criterion. This is for instance the case in Gardent et al. (2017), where annotators were asked if “the text correctly represents the meaning in the data".

Some other approaches break down the evaluation of semantics into different sub-criteria. In their survey paper about metrics used to manually evaluate the output of NLG models, Sai et al. (2022) distinguish in particular between three main categories of criteria to evaluate meaning in Data-to-Text.

(i) Some papers ask human evaluators about the *faithfulness* of the generated text. For example in Dhingra et al. (2019); Tian et al. (2019), authors ask annotators if the output sentence contains only information which is in the input data. We can note that this criterion deals with the precision of the model.

(ii) Some other approaches aim at evaluating the *relevance* of the text. (Castro Ferreira et al., 2020a) ask annotators whether the text describes only the properties of the input RDF triples.

(iii) Finally, other approaches focus on the *coverage* of the input information in the output text, asking annotators if all information from the input data is in the output text (Tian et al., 2019; Castro Ferreira et al., 2020a). This criterion relates to the recall of the model.

Dialog Coherence Evaluation In this paragraph, we review some of the approaches used to evaluate the coherence of automatically generated dialogs.

Automatic Dialog Coherence Evaluation Dziri et al. (2019) measure coherence in dialogs thanks to NLI. They predict whether the dialog context entails, contradicts or is neutral with the next dialog response. Lowe et al. (2017) propose the ADEM metric for response evaluation in dialogs. The metric uses both the context and response references. With the RUBER metric, Tao et al. (2017) blend reference and reference-less evaluation approaches, by using for the evaluation both a reference of the dialog response and checking if the response matches the query. Ye et al. (2021) develop an evaluation approach that quantifies the dialog coherence, which correlates well with human evaluation criteria. The GRADE metric (Huang et al., 2020a) constructs graphs of the topics in the dialog and computes a coherence score based on that graph.

Human Dialog Coherence Evaluation For the human evaluation of their knowledge-grounded conversation generation, Zhou et al. (2021) use two criteria, appropriateness and informativeness (which where originally proposed by Zhou et al. (2018)). Appropriateness measures whether the grammar, topic and logic of the response is correct in the conversation context and informativeness whether it mentions some new information compared to the context. In Wizards of Wikipedia, Dinan et al. (2019) propose a dialog generation model able to retrieve information from a knowledge base. For its human evaluation they use engagingness which they found to increase when the model provides more new knowledge. Engagingness is not directly and exclusively a measure of coherence but can be expected to drop significantly if the model includes a lot of repetitions or jumps illogically from one topic to another. Some papers use a human evaluation criteria which is specific to the coherence of the generated dialog. For instance, Gu et al. (2021) use “answer consistency” and ask annotators if the questions and answers are consistent. Shen et al. (2021) simply use a “coherence" criterion and ask annotators is the generated question is coherent with the input data (in their case a text).

1.2.3.3 Fine-grained metrics and error reporting

As shown by, for instance, Sundararajan et al. (2022); van Miltenburg et al. (2021); Thomson and Reiter (2020), global scores cannot accurately account for the performance of a model. Fine-grained evaluation of NLG is needed and crucial for the development of reliable models.

Automatic fine-grained evaluation Some evaluation metrics are based on fine-grained aspects of the text. This is the case for the PYRAMID metric (Nenkova and Passonneau, 2004) which is a semi-automatic metric that relies on manual labeling to evaluate content selection in document summarization. Zhong et al. (2022) propose a multi-dimensional automatic evaluator for the tasks of summarization and dialog response generation. They train a T5 model to do Boolean Question Answering on four different criteria (Coherence, Consistency, Fluency and Relevance), i.e. answer questions such as “Is this a coherent summary of the document ?”.

Fine-grained Human evaluation Thomson and Reiter (2020) introduce a fine-grained human evaluation of 21 300-word sports stories. They aim at evaluating factuality rather than faithfulness and propose six categories of errors such as incorrect numbers or named entities or not checkable information. van Miltenburg et al. (2021) advocate for more careful and comprehensive reporting of the mistakes done by NLG models and list good practices NLG researchers should follow when evaluating their models. In particular, they mention how evaluated samples could be either chosen randomly or corresponding to

knowingly difficult inputs or with some low automatic evaluation scores. They underline the importance of the granularity of the human annotation and the choice of the errors taxonomy.

Controlled test sets Carefully created or curated evaluation datasets and benchmarks have also been proposed. van Miltenburg et al. (2020) suggest to synthetically create a training dataset using grammars and then evaluate NLG models by checking which rules of the training data they managed to learn. The GEM Benchmark (Gehrmann et al., 2021) introduce distractors in test sets such as numerical variations, changes in the input order (e.g. order of the input RDF triples) or typographical errors. It also breaks down the test split and groups test examples according to criteria such as the shape of the input or its syntactic complexity. This extended and reorganized version of the test set allows for more control in the evaluation of the models.

1.2.4 Our contribution : a fine-grained evaluation metric in RDF verbalization

In Chapter 2, we propose a metric for RDF verbalization on the WebNLG dataset, that aims at evaluating a specific aspect of the output text, namely whether it mentions all input RDF entities. We show that this metric correlates with human judgments of semantics. The metric gives a score of semantic adequacy for each text and therefore resembles the metrics of subsection 1.2.3.2. By using automatic detection of each input entity in the output text, we also provide information on which entity has been omitted. This gives a fine-grained evaluation of the part of the input which is missing in the output and could be thus considered as an automatic fine-grained evaluation metric (1.2.3.3). In Chapter 4, we generate a question and its corresponding RDF triple based on an RDF input graph and a dialog context. We propose a two-step evaluation : first, the evaluation of the content by comparing the generated triple with the input RDF graph and the dialog context, and then the evaluation of the question itself by checking if it matches the generated triple and contains correct and non-ambiguous pronouns. This approach can also be considered a fine-grained automatic evaluation.

1.3 Explainability of NLG models

In this section, we first define the concept of explainability (1.3.1), then we mention some of the challenges for the explainability of state-of-the-art NLG models (1.3.2). We introduce some of the most popular XAI methods in NLG. We mention in more detail probing approaches (1.3.3), which our work in Chapter 3 relates to. Finally, we discuss links between the fields of NLG evaluation and XAI and reflect upon how the work done in the evaluation of content-related issues in NLG can be linked to the explainability of NLG models.

1.3.1 Definition

A recent XAI survey (Barredo Arrieta et al., 2019) defines interpretability or transparency as “a passive characteristic of a model referring to the level at which a given model makes sense for a human observer.". Explainability is rather “an *active characteristic* of a model, denoting any action or procedure taken by a model with the *intent of clarifying or detailing its internal function*". They also claim that “explainability is associated with the notion of *explanation as an interface* between humans and a decision-maker [i.e. the model] that is, at the same time, both *an accurate proxy of the decision-maker [i.e. the model] and comprehensible to humans*."

Surveys and studies about XAI usually divide models into *interpretable* (or intrinsically explainable, transparent, white box, or self-explaining models), e.g. logistic regression, decision trees, bayesian

network ; and *black-box* models, that have to be explained through post-hoc techniques. State-of-the-art NLP and NLG models fall into the second category.

1.3.2 Challenges for the explainability of state-of-the-art NLG models

Compared to other types of black-box models, some characteristics of state-of-the-art NLG models such as pretrained models make them especially difficult to explain.

High complexity of the models State-of-the-art pretrained NLG models can have a very high number of parameters (some examples are given in Table 1.4). This limits the use of some of the most computationally expensive explainability methods.

Access to the data and to the models The lack of access to the pretraining data or to the weights of the models (among the 48 LLMs of size larger than 10B surveyed in (Zhao et al., 2023) ; only 21 are publicly available), is also an obvious limitation to the ability to explain such models.

Evaluation of XAI approaches Provided that an explainability method could be applied to a state-of-the-art NLG model, it is not obvious whether the obtained explanations will be of good quality. This question is the one of the evaluation of XAI which is a very active research area within XAI. Indeed, the evaluation of explanations provided by an explanability method is a common challenge in the XAI field as there is in general no ground truth for an explanation. The evaluation of explanations is also an open question in NLP and NLG.

(Madsen et al., 2022) distinguish between 3 evaluation categories : application-, human- and functionally (or model-) grounded. *Application-grounded* evaluation of explanations measures how much the explanations improved the performance on the task the model is performing, i.e. if the explanations helped improve the task completion. *Human-grounded* evaluation checks if the explanations are considered useful by the users. *Functionally-grounded* explanations aim at providing a faithful representation of the model's inner-workings. In chapter 3, we aim at providing functionally-grounded explanations.

The specific case of NLG In NLP, explainability work is mostly done on classification tasks (e.g. Part-of-Speech tagging or sentiment analysis). The explainability of NLG applications is far less studied. This may be due to the fact that there is no standard definition of what should be explained in NLG. Should we explain the entire output sequence or individual tokens of the output sequence ? Depending on the task and the explainability method, explaining an entire sequence might be less interpretable. Also, *token-by-token explanations* and *sentence or text-level explanations* have different angles. The former focus on explaining the next token generation at each decoding step, while the latter gives a more global view of why a model generated the entire sequence. There is no a priori way of deciding which of the two types of explanations to use as the two are relevant in different ways.

If we choose to compute the explanation for each individual output token, it raises the question of computational cost. XAI methods can be computationally already intrinsically expensive (e.g. approximations of Shapley values), which is a problem if they need to be applied to all the output tokens of all the corpus examples.

In both cases, both for the explanation of the full output text and of each output text token, it is not obvious whether the results will remain readable and humanly understandable.

Some approaches have been working on this issue. For instance, in image captioning, Cafagna et al. (2023) provide SHAP explanations (Lundberg and Lee, 2017a) at the sentence level. They also base their explanations on two kinds of input features, i.e. groups of pixels obtained either with a grid on the image

or using an image concept discovery method called Deep Feature Factorization (Collins and Süsstrunk, 2019). The image concept discovery method outputs image zones corresponding to semantical objects in the picture (such as a person, an object, or a set of objects). Cafagna et al. (2023) refer to them as "semantic priors" and found that they produce better explanations, more understandable by humans, than input features obtained with a simple image grid.

1.3.3 Methods

The explainability of NLG models has not been widely and explicitly studied yet. However, some approaches used for NLP models in general, could be used in the specific case of NLG.

(Madsen et al., 2022) survey post-hoc explainability techniques used in NLP. XAI methods are typically divided into *local* (trying to explain the behavior of the model for individual predictions) and *global* (trying to explain all of the model's behavior) techniques. As underlined by Danilevsky et al. (2020) and Balkir et al. (2022), due to the complexity of models and tasks in NLP, the majority of the approaches are local.

Some approaches target specifically *sequence-to-class* NLP models and others also apply to *sequence-to-sequence* models. In the following paragraph, we review some of the NLP XAI methods that are suited for sequence-to-sequence models and could potentially be applied to state-of-the-art NLG models.

Following (Madsen et al., 2022) we classify approaches according to what the explanations are based on, i.e. either input features, adversarial examples, vocabulary explanations, probes, or analysis of the impact of neurons.

Input Features. A standard approach is to compute the contribution of each input feature to the output of the model. Methods include gradient-based methods such as Integrated Gradients, or input perturbations such as LIME (Ribeiro et al., 2016) or SHAP (Lundberg and Lee, 2017b) (which is an approximation of the Shapley values).

Remark. *The computation of Shapley values is a popular XAI technique, initially coming from game theory (Shapley et al., 1953). Originally, in game theory, the computation of Shapley values aims at assessing the contribution of each player to the result of a game. Then, based on individual contributions, the gains can be divided fairly between players.*

The basic idea behind the computation of Shapley values is to compute the hypothetical gain of each possible subset (or coalition) of players. The contribution of a player is its marginal contribution across all coalitions.

In XAI, players usually become input features, and the contribution of each of them to the model's output is computed. In NLP, input features are generally tokens and the cost of computing the Shapley values varies exponentially with the number of tokens.

As mentioned in Molnar (2022), the use of Shapley values-based methods is often considered fundamental because of the firm theoretical grounding of the Shapley values. They respect some axioms such as symmetry, additivity, or efficiency, which guarantee the fairness and efficiency of the computed features' contributions.

The paper by Mosca et al. (2022) surveys how the different SHAP-based XAI approaches adapt to NLP models. They point at HEDGE (Chen et al., 2020), which is particularly relevant as it provides hierarchical explanations. Explanations can be given based on each input token, input word, or even phrase, which can be in particular a good choice in particular for longer texts. They also mention SAGE (Covert et al., 2020) which is also an approximation of the Shapley values aiming at detecting sets of features that have the most impact on the model's output.

Other methods use attention as a way to measure the importance of input tokens. This is the case in (Abnar and Zuidema, 2020), where the authors compute the attention flow in BERT models. They consider the attention graph (i.e. the graph in which the neurons are the nodes, and the weights are the attention weights) and compute the maximum flow between the CLS embedding and each of the input tokens. A higher max flow suggests higher importance of the corresponding input token. Ethayarajh and Jurafsky (2021) also show that attention flow verifies the axioms of the Shapley values and can thus be considered as Shapley value explanations.

Adversarial examples. The basic idea behind the adversarial approach is to find or create inputs that confuse the model and result in wrong outputs. These adversarial examples help to understand the model's limitations and biases. The survey by Goyal et al. (2023) on adversarial techniques in NLP mentions character-level, word-level, and sentence-level adversarial examples.

The creation of character-level adversarial examples is for example done using insertion, deletion, and swapping of characters, i.e. perturbations which can be seen as spelling mistakes.

Word-level adversarial examples are input sequences in which some words were replaced by syntactically or semantically similar words. Other techniques propose to replace or delete important words, an importance score being calculated for instance using gradient or attention-based methods (e.g. Alzantot et al. (2018)).

In some other approaches, perturbations are done at the sentence-level, on multiple words. For instance Wan et al. (2022) use a gradient-based method to replace entire phrases in machine translation inputs.

Goyal et al. (2023)'s survey does not mention any approaches creating adversarial examples in an NLG task. However, recent approaches such as Wang et al. (2023) create adversarial attacks designed specifically for non-openly accessible large language models such as ChatGPT and GPT4 (OpenAI, 2023). Sarti et al. (2023) also developed a toolkit specifically for the interpretability of generation models.

Vocabulary explanations. Explaining the model's word embeddings can be done by applying some transformation on the embedding space. The idea is to provide a visual explanation of the learned embeddings by transforming them in a way that makes clusters or space dimensions more interpretable. In their survey Zini and Awad (2022) cite three main categories of transformations.

As more sparse embeddings have been shown to be more interpretable, *sparsification* applies transformations to the embedding space in order to make it less dense (Luo et al., 2015; Subramanian et al., 2017). *Rotations and projections* of the embedding space along some meaningful dimensions have also been studied (Rothe and Schütze, 2016; Park et al., 2017). The last transformation consists in *integrating external knowledge* in the embedding space by fine-tuning embeddings using semantic lexicons and ontologies (Faruqui et al., 2015; Jha et al., 2018).

Probing. Probing the information encoded in pre-trained models is an active research field in NLP. However, just like for XAI of NLP in general, most previous work has focused on analysing Natural Language Understanding (NLU) rather than NLG models. Belinkov and Glass (2019) survey approaches which analyse NLU models including in particular, approaches based on probing classifiers. Other work has focused on analysing the internals of BERT and Rogers et al. (2020) provides an extensive survey of the different studies which have looked at the knowledge encoded in BERT weights. There are also multiple papers (Conneau et al., 2018; Adi et al., 2017; Tenney et al., 2019b; Ettinger, 2020) which probe the linguistic properties encoded in the embeddings of various NLU models (e.g. BiLSTM, Gated Conv Net), or in the different layers of the encoder. For instance, Koto et al. (2021) probe the different layers of seven pretrained language models (including BART and T5) using tasks related to discourse structure while Tenney et al. (2019a) study how the information from the NLP pipeline can be localized in different

layers of the BERT encoder and found that low-level semantic information (e.g. information related to part-of-speech tagging or dependencies), appears in the first layers, whereas high-level information, such as coreferences and in context semantic roles, can be found in higher layers.

Neurons detection. Some approaches try to find correlations between the output of a model with some of its neurons' activations. Ghorbani and Zou (2020) compute the contributions of individual neurons to the model's output. They use an approximation for the computation of Shapley values and find that the obtained values are sparse. This means that only a few neurons are responsible for the output. One interesting follow-up is therefore to prune the model by zeroing out the other neurons. They experiment on CNNs with image inputs but Mosca et al. (2022) argue their approach could be adapted to NLP models.

1.3.4 Links between the fields of XAI and NLG Evaluation

"The line dividing XAI methods, and methods that are developed more generally for understanding, analysis and evaluation of NLP methods beyond the standard accuracy metrics is not always clear cut [...] as many popular approaches [...] share many of their core motivations with XAI methods." (Balkir et al., 2022). Unlike evaluation, the field of XAI in NLG is not very developed.

However, we can link the fields of XAI and evaluation in different ways which we describe in the following paragraphs.

Shared motivations They share the same motivation of developing more reliable models. Both XAI and evaluation of NLG models aim at understanding how these models work, checking their robustness, and figuring out what their limitations are.

Relationship with accuracy of the models Both XAI and Evaluation introduce the idea that a model's performance or accuracy is not the only criterion defining its quality. Indeed, XAI conceptualizes a tradeoff between performance and explainability (Došilović et al. (2018), Barredo Arrieta et al. (2019)), claiming that the best-performing models have become increasingly complex and hard to explain whereas white-box models generally have much lower performances. Depending on the application and the risk that can be tolerated, one might choose a model with lower performance but higher transparency.

In NLG multiple papers have advocated for better evaluation and error reporting instead of focusing exclusively on performances reported in models' leaderboards (e.g. Gehrmann et al. (2021)). Models being at the top of such leaderboards but failing to provide fine-grained evaluation are not assured to actually be the best models.

Interaction between XAI and Evaluation techniques Evaluation and XAI methods are complementary and can interact at different stages of the model development or generation process.

Evaluation can be performed prior to applying XAI techniques to a model. As we already mentioned, XAI in NLP (and even more in NLG) has to be local and focused on a specific aspect of the model. If we want to focus on understanding and explaining models' mistakes, we first need to evaluate them. Once the models' issues have been identified using (automatic or human) evaluation, we can use explainability methods to try to understand these issues.

Evaluation can also be applied after an explainability method. As we previously mentioned the evaluation of an XAI method can be done with different perspectives (application, human, or model-grounded). In the case of an application-grounded evaluation, evaluating the explainability method can require determining if the introduction of the XAI method helped improve the texts generated by the

model. This involves evaluating the generated texts, which is exactly what NLG evaluation is. In other words, the evaluation of XAI applied to an NLG model can require NLG evaluation.

Some evaluation and explainability methods are in practice very close to each other. Some XAI methods based on adversarial examples are essentially similar to NLG evaluation methods involving perturbations of the input such as in Sai et al. (2021). In that perspective, evaluation and explainability and be performed simultaneously.

Generating explanations in Natural Language Some XAI approaches aim at outputting explanations in natural language. They claim that as the goal of XAI is ultimately to improve the user trust in the models, providing the explanation in a format easily readable by the user is crucial. However, the natural language explanations themselves have to be evaluated, in particular, to make sure that they are faithful and not misleading. Note that making sure that the explanations provided by an XAI method are not misleading but really reflect the characteristics of the model is a central concern in the XAI field. Whereas natural language explanations have a great potential for providing user-friendly explanations they also possibly can introduce ambiguity or even inaccuracies. NLG evaluation would be therefore crucial to ensure their quality.

Self-evaluating vs Self-explaining models We can see a conceptual link between self-evaluating models vs “self-explaining" models. While designing a model, emphasis can be put on anticipating and integrating ways to either evaluate the model or to make it more transparent. This is what we aim at in chapter 4 where we propose a conversational question generation model that generates both a triple and a question instead of only a question. The model is built in that way to be easier to be automatically evaluated in a fine-grained manner.

1.4 Conclusion

In this chapter, we introduced the task, models, and datasets we will study in the following chapters. We also introduced the research fields and key concepts of the evaluation and explainability of NLG. We reviewed some of the works related to our experiments in the following chapters.

This thesis can be partly seen as an attempt to bridge the fields of evaluation and XAI of content-related issues in state-of-the-art NLG models.

In Chapter 2, we propose a fine-grained automatic metric to evaluate omissions and hallucinations in RDF Verbalizers of the WebNLG Challenges. We evaluate this automatic metric by studying its correlation with human judgments of semantics. Even though omissions are far less studied than hallucinations in the literature, we show that RDF verbalization models, including state-of-the-art ones such as T5 and BART, are subject to omitting important input information.

Having made this observation, we propose to focus on understanding why transformer-based encoder-decoder models are prone to generate omissions. Whereas the origin of omissions has not been studied in depth, the origin of hallucinations has received more attention, as mentioned in 1.2.1.

Our assumption is that omissions and distortions are related to the hallucination problem. Their cause could therefore be similar. In Chapter 3, out of all to possible causes we mentioned in 1.2.1, we choose to first focus on analyzing the encoder representation of input RDF graphs in BART and T5. We propose a methodology based on probing classifiers to help analyze omissions and distortions in BART and T5 RDF Verbalizers’ encoders.

In Chapter 4, we propose to modify the task of Conversational Question Generation from RDF graphs by generating not only a question but also its corresponding triple. By doing so we aim at going in the direction of developing self-evaluating NLG models or at least facilitating automatic evaluation.

2

Entity-Based Semantic Adequacy for Data-to-Text Generation

Contents

While powerful pre-trained language models have improved the fluency of text generation, it remains difficult to ensure that the generated texts are semantically faithful to the input. In this chapter, we introduce a novel automatic evaluation metric, Entity-Based Semantic Adequacy, which can be used to assess to what extent generation models that verbalize RDF (Resource Description Framework) graphs produce text that contains mentions of the entities occurring in the RDF input. This is important as RDF subject and object entities make up 2/3 of the input. We use our metric to compare 25 models from the WebNLG Shared Tasks and we examine correlation with results from human evaluations of semantic adequacy. We show that while our metric correlates with human evaluation scores, this correlation varies with the specifics of the human evaluation setup. This suggests that in order to measure the entity-based adequacy of generated texts, an automatic metric such as the one proposed here might be more reliable, as less subjective and more focused on correct verbalization of the input, than human evaluation measures.

2.1 Introduction

With the introduction of pre-trained models, the fluency of text generation systems has improved. However, ensuring semantic adequacy of the generated text (i.e. faithfulness to the input) remains an unsolved issue (Wiseman et al., 2017; Gehrmann et al., 2018).

1 (**short)**	**Output text** Liselotte Grschebina is a German national who was born in the German Empire and has a total area of 20769100000. 0. (*Liselotte_Grschebina*, *nationality*, *Israel*) (*Israel*, *areaTotal* , *20769100000.0*) **RDF Input** (*Israel*, *officialLanguage*, *Modern_Standard_Arabic*) (*Liselotte_Grschebina*, *birthPlace* , *German_Empire*) (*Liselotte_Grschebina*, *training*, *School_of_Applied_Arts_in_Stuttgart*)
2 (**hal)**	**Output text** Born in the Kingdom of England in 1726-01-01, and living in India, on the 18th of July, 1776, the country is the birth place of Joh Davutoglu. (*Lady_Anne_Monson*, *birthPlace*, *Darlington*) (*Lady_Anne_Monson*, *birthDate*, *1726-01-01*) **RDF Input** (*Lady_Anne_Monson*, *deathDate*, *1776-02-18*) (*Lady_Anne_Monson*, *birthPlace*, *Kingdom_of_England*) (*Lady_Anne_Monson*, *residence*, *India*)
3 (**deg+hal)**	**Output text** Born in the Kingdom of England, and died on 1776-02-18, on 1726-01-01, in the Kingdom of England, the prime minister of community of England is called, Germanic duties, and arrabbiata (born on the 18th of July, 1726-01-01). (*Lady_Anne_Monson*, *birthPlace*, *Darlington*) (*Lady_Anne_Monson*, *birthDate*, *1726-01-01*) **RDF Input** (*Lady_Anne_Monson*, *deathDate*, *1776-02-18*) (*Lady_Anne_Monson*, *birthPlace*, *Kingdom_of_England*) (*Lady_Anne_Monson*, *residence*, *India*)
4 (**deg)**	**Output text** The distributor of the distributor of the distributor of the distributor of the distributor of the distribution of the distribution of the distribution of the dish, Roadside Attrón, is Tom Botta, who starred in the preparation of the tennis Katzman. (*Super_Capers*, *editing*, *Stacy_Katzman*) (*Super_Capers*, *starring*, *Michael_Rooker*) **RDF Input** (*Super_Capers*, *starring*, *Tom_Sizemore*) (*Super_Capers*, *language*, *English_language*) (*Super_Capers*, *distributor*, *Roadside_Attractions*)
5 (**dist)**	**Output text** J R Tolkien is a notable work of The Silmarkien. **RDF Input** (*J._R._R._Tolkien*, *notableWork*, *The_Silmarillion*)

FIGURE 2.1 – Examples of outputs with low Entity-based Semantic Adequacy. RDF input entities that are missing in the text are underlined (short : the short output fails to mention all input entities, deg : degenerate output, hal : the text hallucinates entities not present in the input and omits to mention others, dist : distortion of some input entity)

In this chapter, we focus on semantic adequacy for RDF-Verbalizers i.e., models such as those submitted to the WebNLG 2017 and 2020 shared tasks Gardent et al. (2017); Castro Ferreira et al. (2020b) which map an RDF graph to a text verbalizing the content of that graph. In this case, the input to Natural Language Generation (NLG) is a set of triples of the form $(e1, p, e2)$ where $e1, e2$ are RDF entities and p is a property. RDF triplestores are used in particular to model Semantic Web data and their verbalization aims at making the information from these knowledge bases easily accessible to users. As exemplified in Figure 2.1, one necessary condition for the generated text to be semantically adequate is that all entities present in the input should be mentioned at least once in the output. We refer to this requirement as entity-based semantic adequacy (ESA for short). ESA offers one way of formalizing the requirement that the output of a generator should reflect the information in the input. Thus, its significance extends beyond the specific problem domain of RDF verbalization, though the latter provides a useful test case.

We make the following contributions.

We devise metrics which assess to what extent a text verbalizing an RDF graph respects entity-based semantic adequacy. These metrics rely on an algorithm designed to automatically detect whether an entity present in the input graph has a corresponding mention in the output text. We evaluate this algorithm on a corpus of 25,173 (RDF, Text) pairs with manually annotated entity mentions from Castro Ferreira et al. (2018) and show that our algorithm has a recall of 0.74 and a precision of 0.75.

We apply these metrics to the output of 25 RDF verbalizers developed for the WebNLG 2017 and 2020 challenges and show that some of the systems which rank highest in terms of BLEU scores actually rank in the lower half with respect to entity-based semantic adequacy. This indicates that ESA is measuring a different quality from that measured by surface-based metrics such as BLEU.

We compute correlation between our metrics and both automatic and human evaluation scores collected by the WebNLG organizers for these 25 models. We find a stronger correlation with human metrics related to semantic adequacy than with automatic metrics. Among the automatic metrics, correlations are highest with METEOR. Interestingly, we also find that the correlation with human scores varies with the specifics of the human evaluation setup. This suggests that our automatic metric might be a more reliable means of identifying models with low entity-based semantic adequacy than human evaluation. We are publicly releasing our source code. [3]

2.2 Defining E-Based Semantic Adequacy

We assume a corpus of (R, T) instances where R is an RDF graph (a set of RDF triples) and T is a text verbalizing that graph. RDF triples are of the form (e_1, p, e_2) where p is a binary relation holding between a subject (e_1) and an object (e_2) [4]. We use the term "entity" to refer to both RDF triple subjects and objects and we write E_R for the set of RDF entities occurring in RDF graph R.

Entity Mentions. Given a corpus instance (R, T), an *entity mention* m is a text segment in T which denotes an entity e present in the input graph ($e \in E_R$). We write M_T for the set of entity mentions occurring in T and $[\![m]\!] = e$ to indicate that the mention $m \in T$ denotes entity $e \in E_R$.

(Un)Detected Entities. A detected entity $e \in E_R$ is an entity which has a matching mention in M_T i.e., there is a mention $m \in M_T$ such that $[\![m]\!] = e$. Conversely, an undetected entity is an entity $e \in E_R$ which has no corresponding mention in M_T. We define $E_T \subseteq E_R$ as the set of RDF entities which have a corresponding mention in T.

Entity-Based Semantic Adequacy. Given an (R, T) pair, we define entity-based semantic adequacy (ESA) as the proportion of RDF entities in E_R which have a corresponding mention $m \in M_T$. In other

3. https://gitlab.nl4xai.eu/juliette.faille/entity-based-semantic-adequacy

4. Subjects are Uniform Resource Identifiers (URI) and objects are either URIs or literals. Intuitively, RDF subjects and objects refer to things such as persons, locations, abstract entities, dates or phone numbers.

words, ESA is the ratio between the number of entities for which a mention was found (E_T) and the total number of entities occurring in the input RDF (E_R).

$$\text{ESA}_I = \frac{\mid E_T \mid}{\mid E_R \mid}$$

Given a corpus of (R, T) pairs, we also compute the proportion of texts in that corpus with at least n undetected entities. We refer to this metric as corpus-level, entity-based semantic inadequacy at n ($\text{ESI}_C{}^n$ for short).

2.3 Computing E-Based Semantic Adequacy

The metrics introduced in the previous section rely on being able to determine which entities in the RDF input have a matching mention in the corresponding text. We present an algorithm for entity mention detection and we report on an evaluation of that algorithm using a dataset of 25,173 (RDF graph, Text) pairs where entity mentions have been manually annotated.

2.3.1 Detecting Entity Mentions

We define our entity mention detection algorithm using a combination of existing tools and heuristics.

Entity linker We use the state-of-the-art REL entity linker from van Hulst et al. (2020). When applied to a text, REL returns a list of entity mentions and their corresponding DBPedia entities. We filter out the mentions for which the related DBPedia entity does not match any of the input RDF entities.

Approximate string matching of text n-grams and RDF entities We match text n-grams to candidate RDF entities, using approximate string matching with a fixed maximum allowed edit distance (normalized Levenshtein distance). This value is experimentally fixed at 0.4. The algorithm used by our string matching procedure is described in detail in Algorithm 1.

To improve results, we create a dictionary of RDF entity synonyms and compute the approximate match between text n-grams and all RDF entities, including their synonyms. The synonym dictionary was initially created using DBPedia aliases and rules (handcrafted to improve the detection of frequent entities in the WebNLG corpus such as places or quantities). During the evaluation of the entity detection (cf section 2.3.2), an expanded version of the dictionary was developed by updating it with entities for which no mention was detected in an evaluated text; these were manually included in the dictionary. This dictionary provides a symbolic means to improve entity mention detection and more generally, to adapt the algorithm to a new domain. However, it should be noted that, on the WebNLG 2017 dataset, adding this dictionary only slightly improves entity mention detection and is not essential.

Pronominal entity mentions In order to detect which input entity a pronoun refers to, we use two methods. We first use `Stanza` Qi et al. (2020) to compute co-reference chains in our texts, keeping only the pronominal mentions. For the pronouns that were not detected by the previous method, we used a simple heuristic. In the WebNLG corpus, RDF graphs are created with a single entity as "root". Other entities are meant to describe and provide information about this root. As the texts are quite short we assume that most pronominal anaphors refer to the "root" of the graph. We therefore associate all remaining pronouns to the root entity of the RDF graph.

Algorithm 1: Entity mentions detection by Greedy Approximate String Matching

Input : Set of RDF entities $E_R = \{e_{R_1}, \ldots, e_{R_l}\}$,
Generated Text T,
maximum allowed edit distance d_{max}

Output : Lists of detected mentions for each RDF entity $\{[m_i]_j\}, j \in \{1, \ldots, l\}$

$max\_length = \max_{j \in \{1,\ldots,l\}}$ number of words in $e_{R_j} + 1$

for $n \leftarrow max\_length$ **to** 1 **do**
  compute all text n-grams : $g_1, \ldots, g_N$;
end
Initialize the matrix of distances D of size $l \times N$, with all values set to infinity;
for $j \in \{1, \ldots, l\}$ **do**
  for $k \in \{1, \ldots, N\}$ **do**
    Compute the edit distance between e_{R_j} and g_k ;
    Replace value in $D(j, k)$
  end
end
while $\min D \leq d_{max}$ **do**
  Find j and k such that, $j, k = argmin(D)$;
  Append g_k to list of mentions of e_{R_j}, $\{[m_i]_j\}$;
  Set kth row of D to infinity `(A n-gram can only be matched to one entity)`;
  Set rows of D corresponding to n-grams that overlap with g_k to infinity `(Overlap between n-grams is not allowed)`
end

Dates We use the python library `dateparser` to normalize dates both in the text and in the RDF. Results are further filtered using entity type information from the input RDF graph.

Putting it all together. Each method described above yields a list of mentions found in a text for each RDF entity in an input graph. In case different methods identify the same (or overlapping) mentions, we select those mentions with the lowest edit distance to their matched RDF entity (or one of its synonyms). In case of equality, we keep the longest mention.

2.3.2 Evaluating automatic entity mention detection

Castro Ferreira et al. (2018) manually annotated entity mentions in the WebNLG 2017 dataset (an example of annotation is shown in Table 2.1). We use these manual annotations as gold standard to evaluate our entity mentions detection algorithm. Given M^{auto}, the set of mentions detected on this corpus by our mention detection algorithm and M^{human}, the set of manually annotated mentions, we compute Recall and Precision in the usual way : $Recall = \frac{|M^{auto} \cap M^{human}|}{|M^{human}|}$ and $Precision = \frac{|M^{auto} \cap M^{human}|}{|M^{auto}|}$. The intersection $M^{auto} \cap M^{human}$ is the number of exact string matches between the sets of mentions M^{auto} and M^{human}. We obtain a recall of 0.74 and a precision of 0.75.

If we consider approximate string matching in the computation of $M^{auto} \cap M^{human}$ with a maximum allowed normalized edit distance of 0.2, we obtain a recall of 0.82 and a precision of 0.83. This shows that although some of the automatically detected mentions do not match the gold standard annotations

RDF Input	(*Aleksandra_Kovač*, *activeYearsStartYear*, *1990*) (*Aleksandra_Kovač*, *genre*, *Soul_music*)
Output text	"Aleksandra Kovač began her musical career in 1990, she performs soul music."
Manual annotation	entity :"Aleksandra_Kovač", mentions :"Aleksandra Kovač", "her", "she" entity :"1990", mentions :"1990" entity :"Soul_music", mentions : "soul music"

TABLE 2.1 – Example of manual annotations of entity mentions in WebNLG 2017 by Castro Ferreira et al. (2018)

exactly, they are nonetheless close to them. In Section 2.4.2 below, we show that even though imperfect, our entity mention detection algorithm permits reliably identifying models which have low entity-based semantic adequacy.

2.4 Evaluating RDF-to-text Generation Models

25 models participated in the WebNLG 2017 and 2020 challenges. We apply our entity-based semantic adequacy metrics to the output of these models on the WebNLG 2017 and 2020 test data [5]. We group models with respect to BLEU and $\text{ESI}_C{}^1$ rank. As the text output by the models might differ from the crowdsourced texts we used for the evaluation presented in Section 2.3.2, we report on a manual verification of our entity mention detection algorithm on a sample from these model outputs. Finally, we show some example outputs illustrating different ways in which a generated text might have low entity-based semantic adequacy.

2.4.1 Entity-Based Semantic Adequacy in the WebNLG Shared Tasks

For each model in the WebNLG 2017 and 2020 Shared Tasks, we compute the $\text{ESI}_C{}^1$ score (proportion of texts with one or more RDF entities lacking a matching text mention) and the ESA_I score (proportion of RDF entities in the input with a matching entity mention in the output). The ESA_I scores are averaged over the corpus in three different ways, over all texts (ESA_C score), texts that have at least one undetected entity ($\text{ESA}_C{}^1$) and texts with at least two undetected entities ($\text{ESA}_C{}^2$). Table 2.2 shows the results together with the distribution of undetected entities.

2017 vs. 2020. We see a marked improvement between 2017 and 2020. While in 2017, the ratio of generated texts failing to mention at least one entity varies from 10 to 77% , in 2020 it ranges between 3% and 51%. The trend is similar for the various ESA_C scores with e.g., an $\text{ESA}_C{}^2$ range of [0.17,0.64] in 2017 against [0.36,0.71] in 2020. This corroborates the impression that NLG models have strongly improved in recent years.

2020 NLG. Zooming in on the more state-of-the-art 2020 models, we find that out of a total of 1779 texts and 16 model outputs, the average $\text{ESI}_C{}^1$ score is 17% and the median 10%. In other words, on

5. Examples of outputs of the entity mentions detection are given in the Appendix.

	Model= NUIG-DSI, BLEU=41.33, ESA_I=0.63
Text	The record label of Bootleg Series Volume 1 : The Quine Tapes is Polydor Records and it was recorded in St Louis, Missouri, United States. The album was preceded by Squeeze (The Velvet Underground album).
RDF Input	(*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *recordedIn*, *United_States*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *recordedIn*, *St._Louis,_Missouri*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *precededBy*, *Squeeze_ (The_Velvet_ Underground_ album)*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *recordLabel*, *Polydor_Records*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *recordLabel*, *Universal_Music_Group*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *releaseDate*, *2001-10-16*) (*Bootleg_Series_Volume_1 :_The_Quine_Tapes*, *runtime*, *230.05*)
	Model= CycleGT, BLEU=44.59, ESA_I=0.75
Text	the 11th Mississippi Infantry Monument was established in 2000 and is located in Cumberland County, Pennsylvania.
RDF Input	(*11th_Mississippi_Infantry_Monument*, *established*, *2000*) (*11th_Mississippi_Infantry_Monument*, *location*, *Adams_County,_Pennsylvania*) (*Adams_County,_Pennsylvania*, *hasToItsNorth*, *Cumberland_County,_Pennsylvania*)
	Model= NUIG-DSI, BLEU=47.92, ESA_I=0.67
Text	The Acharya Institute of Technology is located in Soldevanahalli, Acharya Dr. Sarvapalli Radhakrishnan Road, Hessarghatta Main Road, Bangalore – 560090. Its director is Dr. G. P. Prabhukumar and it is located in Mumbai.
RDF Input	(*Acharya_Institute_of_Technology*, *campus*, *"In Soldevanahalli, Acharya Dr. Sarvapalli Radhakrishnan Road, Hessarghatta Main Road, Bangalore – 560090."*) (*All_India_Council_for_Technical_Education*, *location*, *Mumbai*) (*Acharya_Institute_of_Technology*, *director*, *"Dr. G. P. Prabhukumar"*) (*Acharya_Institute_of_Technology*, *city*, *Bangalore*) (*Acharya_Institute_of_Technology*, *wasGivenTheTechnicalCampusStatusBy*, *All _India _Council_for_Technical_Education*)

FIGURE 2.2 – Examples of texts with high BLEU and low ESA_I. (Missing RDF input entities are underlined.)

average, models fail to mention at least one entity 17% of the time.

There are strong differences between the models however. The rule-based models (RALI, Baseline-2017, DANGNT-SGU, Baseline-2020) have low $\text{ESI}_C{}^1$. This is unsurprising as such models can integrate lexicons mapping RDF entities to natural language mentions. Interestingly, among the other five models with an $\text{ESI}_C{}^1$ less than 11%, four are bilingual neural NLG models i.e., models which were trained to transform RDF data not only in English but also in Russian [6].

High BLEU does not guarantee Entity-Based Semantic Adequacy. Figure 2.3 clusters models with respect to both BLEU and $\text{ESI}_C{}^1$ ranks. Models that occur right of the vertical axis have high ESI_C rank (they are in the first 8 group), models that occur above the horizontal axis have high BLEU rank. We see that from the 8 models with highest BLEU rank, only three are also among the 8 models with highest $\text{ESI}_C{}^1$ rank (cuni-ufal, FBConvAI and Amazon_AI). The five other models which rank among the first eight in terms of BLEU score (OSU, CycleGT, NUIG, TGen, bt5) have a BLEU score ranging between 0.45 and 0.54 yet their $\text{ESI}_C{}^1$ score ranges between 10 and 22%. This highlights the fact that a high BLEU score does not guarantee semantic adequacy : while their BLEU score is high, on average these models fail to mention at least one of the input entities 10 to 22% of the time. Figure 2.2 shows some examples of 2020 outputs with low ESA_I and high BLEU score.

Figure 2.3 further shows that no model ranks high in term of both BLEU and entity-based semantic adequacy (no model in the top right corner).

2.4.2 Manual Verification of the ESI_C results

Our mention detection algorithm does not detect all mentions, while texts generated by the WebNLG models might differ from the crowdsourced texts on which we evaluated our entity mention detection algorithm (cf. Section 2.3.2). Therefore, we manually verify the result of our mentions detection algorithm for different types of models. We focus on five models with contrasting BLEU and $\text{ESI}_C{}^1$ rank, two models with high $\text{ESI}_C{}^1$ rank but low BLEU rank ; one model with high rank for both dimensions ; one model with high BLEU rank and low $\text{ESI}_C{}^1$ rank ; and one model with low rank in both dimensions. For each of these models, we check texts with different numbers of missing entities (one or two missing entities for the models with high $\text{ESI}_C{}^1$ and one, three and five missing entities for the models with low $\text{ESI}_C{}^1$) and computed the rate of false positives, i.e. entities which were labeled as undetected by our algorithm but which are in fact present in the generated text. While for the three models which rank high in terms of entity-based semantic adequacy, the rate of false positives is high (100% for RALI, 81% for Huawei and 52% for FBConvAI) [7], for models with low ESI_C rank, the number of false positives is much lower (49% for bt5, 13% for Orange). In other words, our entity mention detection algorithm is best at detecting models with low semantic adequacy.

2.4.3 Qualitative Analysis

Examining outputs with low ESA score, we found three main causes for low semantic adequacy : short output, hallucination and degenerate output. Figure 2.1 shows some examples. When the output text is much shorter than expected, many mentions are missing (Ex.1). When the model hallucinates entities

6. The WebNLG Challenge 2020 included the task of RDF-to-Text generation in Russian as well as in English.

7. The repetition of the same entities in different entries of the dataset has a strong impact here. For instance, for the RALI system, on the 1779 texts we checked, our algorithm finds 51 texts with undetected entities but only nine of these entities are distinct. That is, the algorithm fails to detect nine entities and as these are in multiple corpus instances, it has 100% of false positives on these corpus instances.

Model	1	2	3	4	5-8	>1	↓ $\text{ESI}_C{}^1$	Type	BLEU	↑ESA_C	↑$\text{ESA}_C{}^1$	↑$\text{ESA}_C{}^2$
2020												
RALI	50	1			0	51	3%	Symb	11	0.99	0.8	0.6
Baseline-2020	55	1			0	56	3%	Symb	10	0.99	0.78	0.6
Huawei	62	4			0	66	4%	T5	12	0.99	0.79	0.65
DANGNT-SGU	98	2	1		0	101	6%	Symb	9	0.99	0.76	0.56
Baseline-2017	94	27	0	2	7	130	7%	Symb	15	0.97	0.65	0.36
FBConvAI	154	4			0	158	9%	BART	3	0.98	0.77	0.62
cuni-ufal	169	11	2		0	182	10%	mBART	7	0.98	0.78	0.65
Amazon_AI	175	10			0	185	10%	T5	1	0.98	0.78	0.69
OSU	171	13	1		0	185	10%	T5	2	0.98	0.78	0.65
CycleGT	240	25	1		0	266	15%	T5	8	0.97	0.81	0.71
NUIG-DSI	203	62	17	0	1	283	16%	T5	4	0.96	0.76	0.67
bt5	305	45	4		0	354	20%	T5	5	0.95	0.76	0.62
TGen	264	78	26	15	17	400	22%	T5	6	0.94	0.72	0.58
NILC	499	123	13	5	0	640	36%	BART	16	0.89	0.69	0.56
ORANGE	600	190	45	9	2	846	48%	BART	14	0.83	0.65	0.5
UPC-POE	589	230	63	19	7	908	51%	T5	13	0.84	0.7	0.59
2017												
Tilburg SMT	179	8				187	10%	SMT	2	0.97	0.7	0.55
UPF-FORGe	203	18				221	12%	Symb	4	0.97	0.73	0.56
Melbourne	371	74	11			456	24%	NMT	1	0.94	0.76	0.64
Tilburg NMT	555	171	20	3		749	40%	NMT	6	0.89	0.72	0.62
Tilburg Pipeline	304	233	122	72	49	780	42%	Symb	5	0.76	0.42	0.2
Adapt	482	295	130	52	15	974	52%	NMT	8	0.76	0.54	0.38
PKUWriter	529	282	135	106	60	1112	60%	NMT	3	0.71	0.52	0.36
UIT-DANGNT	47	138	238	317	630	1370	74%	Symb	9	0.28	0.02	0
Baseline	377	398	249	207	206	1437	77%	NMT	7	0.47	0.31	0.17

TABLE 2.2 – Entity-Based Semantic Adequacy of the WebNLG Challenge 2020 and 2017 Participant Models. $\text{ESI}_C{}^1$: Proportion of texts with at least one undetected mention (lower is better). The second to sixth columns indicate the number of texts with n undetected entities. The last three columns give the corpus average of the text level ESA_I score, for all texts (ESA_C), for texts with at least one undetected entity ($\text{ESA}_C{}^1$) and for texts with at least two undetected entities ($\text{ESA}_C{}^2$). For ESA_I scores, higher is better. BLEU indicates the rank of the model in terms of BLEU in the WebNLG Shared Task and Type, the type of model (Symb : the model integrates a symbolic component, BART, mBART, T5 : the pre-trained model used).

not present in the input, it also often simultaneously fails to mention those that are (Ex. 2). Finally, entity mentions may be missing because of degenerate output (Ex.3).

2.5 Correlation with Human and Automatic Metrics

We study the correlation of our ESA_I metric with the human and automatic metrics used in the WebNLG challenges.

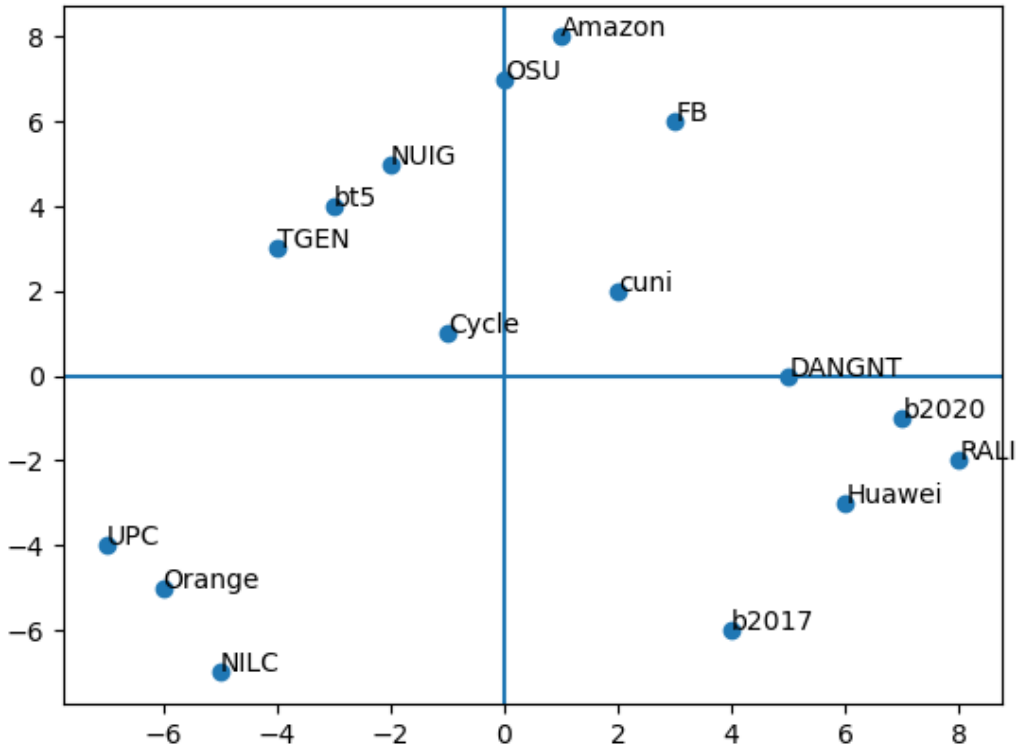


FIGURE 2.3 – **BLEU vs $\mathbf{ESI^1_C}$ ranks for WebNLG2020 models** (higher is better i.e., position 8 in the graph indicates the highest ranked system). The top part of the figure shows the top 8 ranked models w.r.t. BLEU, the right most part the top 8 ranked models w.r.t. ESI^1_C. Of the 8 top ranked models w.r.t. BLEU (top part of the figure), only two (FB and OSU) are among the 8 top ranked w.r.t. ESI^1_C scores.

2.5.1 Evaluation Set-Up

During the WebNLG Challenges 2017 and 2020, 223 texts were sampled from the outputs of the participants models for human evaluation in 2017, and 178 in 2020 (Shimorina et al., 2018; Castro Ferreira et al., 2020a). For our correlation study, we therefore use the automatic and human evaluation scores collected by the WebNLG organizers for 2,007 texts (223 for each of the 9 models) in 2017 and 2,848 texts (178 for each of the 16 models) in 2020. We use the results and scripts from Shimorina et al. (2018) and Castro Ferreira et al. (2020a).

In 2017, the human evaluation metric which focuses on semantic adequacy is Semantics where the annotator is asked to assess semantic faithfulness of the generated output w.r.t. the input (1-low, 2-medium or 3-good). In 2020, the human evaluation metrics concerned with semantic adequacy are Data Coverage, Correctness, Relevance (between 0 and 100). For Data Coverage, evaluators were asked to check whether all input RDF properties were in the text; for Relevance, whether the text describes only such predicates which were present in the input; and for Correctness, whether the output text correctly describes the subject and object of those predicates which matched a property in the input graph. Note that while all these criteria bear on semantic adequacy, none of them specifically target entities.

2.5.2 Results

We compute the correlations with ESA_I metric at text level in three different set-ups (for all texts, for texts with at least one undetected entity and for texts with at least two undetected entities) using three correlation metrics (Pearson correlation, the Spearman rank correlation and Kendall's Tau). Tables 2.3 and 2.4 only report Pearson correlations on texts with at least one undetected entity ($n = 822$ for 2017; $n = 470$ for 2020). Detailed correlation results for the three metrics and considering the three text set-ups are reported in the Appendix.

Human vs. Automatic Metrics. The correlation between ESA_I and human metrics of semantic adequacy is strong in 2017 and moderate in 2020, indicating that ESA_I correctly captures what humans judge to be semantically adequate. ESA also has very strong (2017) and moderate (2020) correlation with METEOR, which suggests that variants of entity mentions involve synonyms and variations captured by stemming.

Varying Correlation Strengths and Scale. The strength of the correlations and their relative order vary with the shared tasks. For automatic metrics, this is likely due to greater variance in metric scores between systems. For human judgments, it might also result from the different criteria used in 2017 vs. 2020 and from their subjectivity. Indeed, as shown in Figure 2.4, some of the collected human judgments are in fact incorrect. Not shown here, but reported in the appendix, correlations with Human, METEOR and BLEU scores also tend to be higher for texts with at least one or two undetected entities – that is, texts which are likely to have semantic adequacy problems – compared to correlations computed over all texts (compare Table 2.4 to correlations over all texts, in the Appendix).

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.74	-0.57	0.39	0.43	0.53	0.59
METEOR	x	-0.54	0.57	0.63	0.72	**0.87**
TER	x	x	-0.42	-0.45	-0.4	-0.42
Fluency	x	x	x	0.89	0.51	0.49
Grammar	x	x	x	x	0.57	0.57
Semantics	x	x	x	x	x	**0.66**

TABLE 2.3 – Pearson correlation coefficients for WebNLG 2017 metrics and ESA_I. Only for texts with at least one undetected entity (i.e. 822 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $\text{ESA}_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Cases of strong disagreement between ESA_I and human evaluation We manually checked some of the texts which received high human evaluation scores but had a low proportion of detected entities ESA_I (resp. texts with low human evaluation scores but high ESA_I). [8]

For WebNLG 2017, there are 7 texts that have $\text{ESA}_I < 0.4$ and Semantics≥ 2. For 6 of them we find that there are indeed missing entities, while the remaining one is a degenerate text. The relatively high scores given by human evaluators for Semantics (2 out of 3) suggest that such scoring tends to be subjective, perhaps especially so with a broad evaluation criterion such as 'Semantics'. Among the 41 texts which received the lowest possible rating for semantics (Semantics=1), but had $\text{ESA}_I > 0.9$, we find that 25 texts do not have missing entities but are indeed semantically incorrect (usually because of mistakes or hallucinations of predicates); 3 texts have missing entities and 9 texts have hallucinated entities. The remaining 4 texts contain all input entities and have correct semantics.

8. Examples are given in Figure 2.4.

	High human evaluation score and Low ESA$_I$
(1) **RDF Input**	(*Mermaid_(Train_song)*, *genre*, *Pop_rock*) (*Mermaid_(Train_song)*, *releaseDate*, *2012-12-27*) (*Mermaid_(Train_song)*, *precededBy*, *Thisll_Be_My_Year*) (*Mermaid_(Train_song)*, *writer*, *Espen_Lind*) (*Mermaid_(Train_song)*, *runtime*, *3.16*) (*Mermaid_(Train_song)*, *genre*, *Reggae*)
Text	The song, "Mermaid", was preceded by a ll Be My My Year, Train, which was released on 27th December 2012. The song was written by the band Train and was launched on the 27th of December, 2012.
Human evaluation scores	Correctness : 99 Data Coverage : 99.3 Relevance : 89.7
ESA$_I$	0.29
Generation problems	Missing entities and strong spelling issue for an entity
	Low human evaluation score and High ESA$_I$
(2) **RDF Input**	(*English_Without_Tears*, *musicComposer*, *Nicholas_Brodszky*) (*English_Without_Tears*, *editing*, *Alan_Jaggs*) (*English_Without_Tears*, *runtime*, *89.0*)
Text	English Without Tears, whose draft team is Nicholas Brodszky, has the following number Alan Jaggs, and the draft pick, 89.0.
Human evaluation scores	Correctness : 23.5 Data Coverage : 79 Relevance : 82.5
ESA$_I$	1
Generation problems	Hallucinated predicates

FIGURE 2.4 – Examples of disagreement cases between human evaluation of semantic adequacy and ESA$_I$

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.97	0.71	0.82	-0.67	0.69	0.66	0.71	0.49	0.42	0.3	0.34	0.33	0.31	<u>0.41</u>
2 BLEU NLTK	x	0.77	0.87	-0.74	0.74	0.72	0.77	0.54	0.45	0.34	0.39	0.36	0.36	0.39
3 METEOR	x	x	0.9	-0.62	0.67	0.82	0.78	0.67	0.49	0.49	0.4	0.42	0.36	**0.45**
4 chrF++	x	x	x	-0.69	0.74	0.82	0.82	0.6	0.51	0.46	0.41	0.43	0.37	**0.45**
5 TER	x	x	x	x	-0.76	-0.67	-0.75	-0.61	-0.41	-0.31	-0.42	-0.39	-0.4	-0.24
6 BERT-score P	x	x	x	x	x	0.83	0.95	0.73	0.6	0.41	0.52	0.56	0.5	0.39
7 BERT-score R	x	x	x	x	x	x	0.95	0.75	0.57	0.52	0.49	0.49	0.45	<u>0.43</u>
8 BERT-score F1	x	x	x	x	x	x	x	0.77	0.61	0.49	0.53	0.55	0.5	**0.44**
9 BLEURT	x	x	x	x	x	x	x	x	0.62	0.54	0.52	0.59	0.5	<u>0.43</u>
10 Correctness	x	x	x	x	x	x	x	x	x	0.75	0.71	0.83	0.67	<u>0.56</u>
11 DataCoverage	x	x	x	x	x	x	x	x	x	x	0.62	0.76	0.57	**0.57**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.67	0.86	0.41
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.65	0.53
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.36
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE 2.4 – Pearson correlation coefficients for WebNLG 2020 metrics ESA_I. Only for texts with at least one undetected entity (i.e. 470 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

The same kind of observations can be made for WebNLG 2020 models. There are 4 texts which received high human evaluation scores for semantic adequacy-related criteria (Data Coverage$>$ 80 or Correctness$>$ 80 or Relevance$>$ 80) and low ESA_I (ESA_I $<$ 0.4) and all of them have missing entities. In contrast, there are 20 texts which got low human evaluation scores (Data Coverage$<$ 30 or Correctness$<$ 30 or Relevance$<$ 30) and high ESA_I (ESA_I $>$ 0.9). Fifteen have wrong or hallucinated predicates, two have significant spelling or repetition problems. Three texts are correct.

From these observations, we can draw two main conclusions. First, detecting input RDF entities in the output text is not a sufficient condition to assess a model's semantic adequacy. It does not give information about hallucination of entities or about correct verbalization of RDF predicates, which are also necessary conditions for semantic adequacy. These observations also illustrate the subjectivity of human evaluation. Sometimes correct texts can be rated badly by human annotators or vice versa. One possible reason for this is that texts can have a correct meaning but not be fluent or grammatical. In that case, texts are more difficult to read and it is harder for annotators to notice correct semantics.

Detection of Hallucinations We can use our entity mention detection algorithm in reverse to detect hallucinations i.e., mentions that have no corresponding RDF entity in the input graph. We gather all (entity, mention) pairs found by the entity linker (2.3.1) for which the entity does not occur in the input RDF graph. Table 2.5 summarizes the results for each model of the WebNLG challenges. Figure 2.5 also shows examples of different types of hallucinations.

2.6 Conclusion

RDF stores have become increasingly popular as a means to make knowledge available on the web Assi et al. (2020). We propose an automatic metric for assessing the entity-based semantic adequacy of RDF verbalizers and show that it is effective in highlighting semantic inadequacy even for state-of-the-art models with high BLEU scores. We further show that models detected by this metric as having low

Model	>1	>1$_\checkmark$	Dist	$\downarrow$ ESI$_C$[1]
RALI	0	0	0	0%
B-2017	1	1	1	0.1%
B-2020	1	1	1	0.1%
NUIG	4	3	3	0.2%
UPC	4	4	3	0.2%
DANGNT	5	5	5	0.3%
TGen	8	7	2	0.5%
cuni-ufal	9	7	6	0.5%
Amazon	9	9	3	0.5%
FBConvAI	17	11	6	1%
CycleGT	19	18	10	1%
OSU	20	19	3	1%
bt5	36	17	3	2%
Huawei	48	47	28	3%
NILC	117	99	66	7%
ORANGE	288	288	60	16%
UIT	1	0	1	0.1%
Tilburg SMT	4	0	4	0.2%
Tilburg NMT	11	4	7	0.6%
UPF	12	8	4	0.6%
Tilburg Pl	14	11	6	0.8%
Melbourne	114	112	24	6%
Adapt	241	234	151	13%
PKUWriter	286	283	135	15%
Baseline	754	144*	147	40%

TABLE 2.5 – Hallucinations (>1 and >1$_\checkmark$) : number of texts with at least one hallucination before and after a manual check of automatically detected hallucinations. *Verification on 144 randomly chosen texts. Dist : number of distinct detected hallucinated entities.

1 (**add**)	**Output text**	Bananaman was created by Steve Bright and starred Bill Oddie. It was broadcast by the BBC, which is based in the Broadcasting House in London, and last aired on 15th April 1986.
	RDF Input	(*BBC*, *city*, *Broadcasting_House*) (*Bananaman*, *starring*, *Bill_Oddie*) (*Bananaman*, *creator*, *Steve_Bright*) (*Bananaman*, *lastAired*, *"1986-04-15"*) (*Bananaman*, *broadcastedBy*, *BBC*)
2 (**repl**)	**Output text**	Aaron Turner performs Trance music and played with the band Bobina.
	RDF Input	(*Andrew_Rayel*, *associatedBand/associatedMusicalArtist*, *Bobina*) (*Andrew_Rayel*, *genre*, *Trance_music*)
3 (**inac**)	**Output text**	Cyril frankel is the director of the film "it's Great to be New (1956"), which was written by "ted willis" and starred cecil Parker and John mills. Mr. millis died in 2005.
	RDF Input	(*Its_Great_to_Be_Young_(1956_film)*, *starring*, *Cecil_Parker*) (*Its_Great_to_Be_Young_(1956_film)*, *writer*, *Ted_Willis*) (*Its_Great_to_Be_Young_(1956_film)*, *starring*, *John_Mills*) (*Its_Great_to_Be_Young_(1956_film)*, *director*, *Cyril_Frankel*) (*John_Mills*, *deathYear*, *2005*)

FIGURE 2.5 – Examples of hallucinations (underlined in the texts). add : output contains additional information, repl : an input RDF entity is replaced by another with the same context (here the name of another musician), inac : the name of the input entities are inaccurate which makes them difficult to link with input entities

entity-based semantic adequacy can still have high scores on surface-based metrics, and that while ESA correlates with human scores on semantic criteria, it may in fact be more reliable as a means of detecting low-performing models than human-based evaluation protocols, which tend to be subjective. ESA offers a fine-grained evaluation of a specific aspect of RDF verbalization, the mention of input entities in the output text. Looking at models' results with this level of detail gives a very different picture compared to using global metrics such as BLEU.

3

Probing Omissions and Distortions in Transformer-based RDF-to-Text Models

Contents

In the previous chapter, we saw that NLG models, and in particular RDF-to-Text models, sometimes omit important information in the output text. To better understand and analyze how this type of mistake arises, we explore two methods of probing omissions in the encoder output of BART (Lewis et al., 2020) and of T5 (Raffel et al., 2019), both models fine-tuned for RDF verbalization. (i) We first propose a novel parameter-free probing method based on the computation of cosine similarity between embeddings of RDF graphs and of RDF graphs in which we removed some entities and then (ii) a parametric probe which performs binary classification on the encoder embeddings to detect omitted entities. We also extend our analysis to distorted entities, i.e. entities that are not fully correctly mentioned in the generated text (e.g. misspelling of entity, wrong units of measurement). We found that both omitted and distorted entities

can be probed in the encoder's output embeddings. This suggests that the encoder emits a weaker signal for these entities and therefore is responsible for some loss of information. This also shows that probing methods can be used to detect mistakes in the output of NLG models.

3.1 Introduction

Neural models have drastically advanced the state of the art in Natural Language Generation (NLG) (Kale and Rastogi, 2020; Agarwal et al., 2020; Guo et al., 2020) but are susceptible to two types of problems : *omissions*, where the generated text fails to mention important information present in the input ; and *hallucinations*, where the generated text includes information not licensed by the input. In particular, we quantified and evaluated these two types of mistakes in WebNLG Challenge RDF-to-Text models in the previous chapter. While various metrics (Balakrishnan et al. (2019); Dhingra et al. (2019); Durmus et al. (2020); Filippova (2020)) and mitigation methods (Shuster et al. (2021); Rebuffel et al. (2021a); Fan et al. (2019); Wang et al. (2021)) have been proposed to address these shortcomings, little has been done to analyze where these failures stem from. Furthermore, little work has studied omissions even though, just like hallucinations, these are a crucial element to mitigate in order to obtain reliable and trustworthy NLG models.

In this chapter, we focus on omissions in KG-to-Text Generation, the task of generating natural language text from a Knowledge Graph (Gardent et al., 2017; Castro Ferreira et al., 2020a; Ribeiro et al., 2020). We consider a KG consisting of RDF triples. We study the omissions of entities in texts generated from RDF inputs whereby both the RDF data and the generated texts are in English. An illustrative example is given in Figure 3.1 which shows an example input graph, a text generated from that graph and a list of omitted entities. We also consider a second phenomenon, closely related to omission, which we term *distortion*, where an entity is mentioned in the generated text, but its name is partially incorrect, due to misspelling of a proper name for example, or due to an incorrect number (examples are given in Figures 3.1 and 3.3).

Our research goal is to understand where these omissions and distortions come from and whether we can predict them from the embeddings of the encoder. Specifically, we hypothesize that the encoding of an input graph whose corresponding output text involves an omission should be distinguishable from the encoding of a graph which does not lead to an omission.

A common methodology to help understand which part of a neural model is responsible for a given behavior, is to use a probe, i.e., to test whether this behavior can be predicted from the model internals. We introduce two probes for the analysis of omissions and distortions : (i) a parameter-free probe where omissions can be detected from the model-internal representations without needing to learn new parameters ; and (ii) a probing classifier trained on a dataset of KG representations which are categorized as 0 if the generated text fails to correctly verbalize some input entity and 1 if all entities are correctly verbalized. Both probes indicate that the encoder representations of graphs which lead to an omission in the generated text can be distinguished from those that do not. To our knowledge, this is the first systematic demonstration that, in encoder-decoder architectures for RDF-to-text generation, the encoder plays a role in determining whether content is omitted in the output. We also provide evidence that omissions cannot entirely be predicted as a result of the decoding strategy used during generation and examine the features characteristic of omitted entities. We make the following contributions.

We create two datasets of (RDF graph, generated text) pairs annotated for omissions and distortions : a dataset of 72k instances where omissions were annotated automatically and a dataset of 12k instances where the annotation was manual. We make these datasets publicly available together with the models and scripts necessary to replicate our results [9].

9. https://gitlab.nl4xai.eu/juliette.faille/probing-omissions

RDF Graph
(*Nurhan_Atasoy*, *award*, *State_Award_for_Superior_Achievement*)
(*Istanbul*, *populationMetroDensity*, *2691.0*)
(*Nurhan_Atasoy*, *residence*, *Turkey*)
(*Nurhan_Atasoy*, *birthPlace*, *Reşadiye*)
(*Nurhan_Atasoy*, *residence*, *Istanbul*)

Linearized Input
Nurhan_AtasoyawardState_Award_for_Superior_AchievementIstanbulpopulationMetroDensity2691.0
Nurhan_AtasoyresidenceTurkeyNurhan_AtasoybirthPlaceReşadiyeNurhan_AtasoyresidenceIstanbul

Generated Text
Reşadiye born and Istanbul based, Guran Ataturk, won the State Award for Superior Excellence. Istanbul has a population density of 2691.0.

RDF entities
Nurhan_Atasoy, State_Award_for_Superior_Achievement, Istanbul, 2691.0, Turkey, Reşadiye

Manually annotated omissions Turkey

Manually annotated distortions (underlined) Nurhan_Atasoy,
State_Award_for_Superior_Achievement

Automatically detected omissions or distortions Nurhan_Atasoy, Turkey

FIGURE 3.1 – Example of an RDF input and Generated Text with corresponding results of the automatic entity detection, and manual annotations of omissions and distortions

We propose two main methods to detect omissions and distortions in the encoder's embeddings : (i) a parameter-free method using cosine similarity between the embedding of an RDF graph and the embedding of the same RDF graph from which we removed mentioned, omitted or distorted entities and (ii) a parametric method using a neural binary classifier.

We find that in most entries of our dataset, omissions and distortions can be detected in the encoder outputs by both probing methods. Using logistic regression, we further explore whether the likelihood of an entity being omitted or distorted can be predicted from its features.

The chapter is structured as follows. In Section 3.3, we describe the generation models (BART and T5) [10] and present the (RDF,Text) data we use to fine-tune them. We further explain how we annotate the output of these models with the (possibly empty) set of RDF entities which are omitted or distorted in the output text. In Section 3.4, we give some motivation for the choice of our two probing methods. Section 3.5 introduces the new parameter-free probing method we propose and presents its results. In Section 3.6, we present the second probe, a neural classifier, and we discuss the results obtained. We also describe different control and upper bound experiments which we use to evaluate the quality of that probe. In Section 3.7, we present the results of a logistic regression classifier, which investigates whether some of the dataset features can be correlated with omitted and distorted entities. Section 3.8 summarizes the probing experiments on T5 and Section 3.9 outlines the main limitations of our approach.

3.2 Related Work

We first discuss work on probing pre-trained models. We then go on to discuss previous work focusing on content-related issues in NLG.

Probing of Pre-trained Models We extend the work on probing classifiers that we mentioned in Chapter 1 by analyzing the source of omissions and distortions in RDF-to-Text models.

Previous work on probing has also pointed out some important methodological issues which arise when using a probing classifier. Belinkov (2021) stresses the importance of using controls and upper bounds while Adi et al. (2017) criticize the lack of connection between probing tasks and downstream tasks in approaches where the embeddings of the encoder are analyzed independently of any task. In line with this work, we use probing to analyze omissions in the downstream task of generation from RDF data ; and we situate the results of our parametric probe with respect to both an upper bound and several control tasks.

Content-related issues in text generation Evaluating and minimizing semantic errors in the output of Neural NLG models has been the topic of extensive research. Li et al. (2022)'s survey paper summarizes the different evaluation and mitigation techniques which can be used to address faithfulness in NLG. Evaluation has given rise to several recent shared tasks such as Gehrmann et al. (2021) and Thomson and Reiter (2021). Multiple papers try to improve the reporting of models' mistakes by giving guidelines to avoid under-reporting of errors (van Miltenburg et al., 2021) or to provide standardized definitions of model errors (Howcroft et al., 2020; Belz et al., 2020). Other works have developed metrics to measure semantic adequacy between output text and source information in summarization (Maynez et al., 2020; Scialom et al., 2021).

The specific problem of hallucinations has been studied in multiple NLG tasks as shown by (Ji et al., 2022)'s survey and some work has analyzed hallucinations in Machine Translation (Lee et al., 2018; Raunak et al., 2021).

10. We carry out most of the experiments on BART and do a shorter experiment on T5.

Overall however, relatively little work has focused on analyzing errors in NLG models and to our knowledge, no work has been done so far on analyzing the source of omissions and distortions in RDF-to-Text generation models.

3.3 NLG Models and Annotated Data

To create the data necessary for analyzing omissions in RDF-to-Text generation models, we train two RDF-to-Text Generation models and use these models to generate two large datasets of (RDF graph, text) pairs. Using both automatic and manual annotation, we then annotate the generated texts for omissions and distortions.

3.3.1 Generation Model

To assess generalization, we train two RDF-to-Text generation models by fine-tuning BART (Lewis et al., 2020) and T5 (Raffel et al., 2019), on the WebNLG training set (Castro Ferreira et al., 2020a) [11], a dataset of 47k (RDF graph, text) pairs where the RDF graphs are subgraphs of DBPedia and texts are crowd-sourced. We choose the WebNLG dataset as this data is a reference dataset for the RDF-to-Text task, annotated manually and therefore less likely to contain noise than synthetic or automatically scraped data. The absence of noise is important as we want to analyze mistakes which can be attributed to the generation model instead of mistakes coming from training on a noisy dataset. Likewise, BART and T5 are well-suited for our experiments since they achieve state-of-the-art performance on many generation tasks, and the best-performing models in the 2020 edition of the WebNLG shared task were based on these models (Yang et al., 2020; Castro Ferreira et al., 2020a).

As manual annotation is costly, we only perform manual annotation on the BART generated texts and therefore focus the presentation on our experiments with BART (Sections 3.3-3.6). The additional T5 experiments are discussed in Section 3.8.

The details of the fine-tuning and evaluation of the fine-tuned BART and T5 models are given in the following paragraph.

Fine-tuning details Our fine-tuning experiments are done using 2 Nvidia GTX 1080 Ti GPUs (11 GiB memory). We use pretrained models from the HuggingFace library `https://huggingface.co/facebook/bart-large` and `https://huggingface.co/t5-small`.

BART We fine-tune the BART large model which has 12 encoder and decoder layers and 400M parameters. We use the AdamW optimizer with an initial learning rate of $1e^{-5}$. We train during 12 epochs on WebNLG Train, using as input linearized RDF graphs without special tokens. For the decoding, we use greedy decoding with a maximum length of the generated sequence of 100 tokens and without length penalty. We evaluate the model on the WebNLG test set and obtain a corpus BLEU score of 0.31 (Papineni et al., 2002), a chrF score of 0.53 (Popović, 2015) and Bert score precision, recall and f-score of 0.86 (Zhang et al., 2019).

We find that these results are lower than those of the submissions of the WebNLG Challenge using BART (Yang et al., 2020; Montella et al., 2020) and pretraining on some external datasets, such as DocRED (Yao et al., 2019) or 103 millions of sentences extracted from Wikipedia. However they are comparable to the submission's result from Sobrevilla Cabezudo and Pardo (2020) which did not use any additional pretraining.

11. To fine-tune the model, we linearized the RDF triplesets without adding any special tokens to separate the triples or the entities or properties within a triple.

T5 We fine-tune a T5 small model with 6 layers in encoder and decoder and 60M parameters. We use the Adafactor optimizer with $2e^{-5}$ learning rate. We use early stopping and trained for 72 epochs, with a 16 batch size. We evaluate the model on the WebNLG test set and obtain a corpus BLEU score of 0.34, a chrF score of 0.58 and Bert score precision, recall and f-score of 0.88.

3.3.2 (RDF,Text) Data

As input to generation, we use graphs from both WebNLG and KELM (Agarwal et al., 2021), a large dataset of Wikidata graphs paired with synthetically generated texts. Using graphs from KELM increases the diversity of entities in our dataset. In addition, including graphs from both the KELM and the unseen part of the WebNLG test data permits comparing the performance of our probes on both in- (WebNLG train, dev and test seen) and out-of-domain (WebNLG test unseen, KELM) input graphs.

We use all 16,657 RDF graphs from the WebNLG V3.0 dataset and 6k graphs from the KELM dataset (1k graphs for each graph size from 1 to 6 triples), which yields a total of 22,657 RDF input graphs.

We then augment this dataset by permuting the linearized graphs as we observe that such permutations can lead to differences in the generated texts. The permutations we use are random changes in the order of the triples in an input graph. We randomly choose a maximum of 6 permutations of the original linearized graph. This means that for triplesets of length 2 or 3, we keep all possible permutations. For triplesets of length 4 to 7, we randomly choose 6 [12]. This yields a total of 93,488 RDF graphs.

We then generate texts from these graphs using our BART model. However, as not all permutations of the input graphs lead to a different text, we keep only the 71,644 (graph,text) pairs which have distinct output texts.

3.3.3 Annotated Data

We annotate each (RDF graph, Generated Text) pair with the (possibly empty) set of RDF entities which are omitted or distorted in the generated text. We create these annotations using both automatic and manual annotation to assess whether our probing methods yield substantially different results when relying on automatic annotations. Finding that there is little or no difference between the two methods would indicate that a cheaper, albeit noisier, automatic method can still yield useful insights.

For the automatic detection of omissions, we use the algorithm developed in 2, which has a recall of 0.74 and a precision of 0.75 on WebNLG data.

For the manual annotation, we asked three annotators to check the automatic detection focusing on texts for which at least one omission was found. [13] For each entity in the input RDF graph, they annotated whether the entity is mentioned in the text, omitted or distorted. Annotators are for instance instructed to label as distortions misspelling of proper names, partial omission of entities or wrong units of measurement. The three annotators were students following an English taught Master in NLP who had native (2) or high proficiency (1) in English. There were employed by the University on a 20 hour/month work contract with a salary determined by the University grids and including social benefits in line with the host country work laws. The annotation task took a total of 587 hours. Instructions and examples given to the annotators are shown in Figures 3.2 and 3.3. For the annotation interface, we use CryptPad (`https://cryptpad.fr/`). We show an annotation example in this interface in Figure 3.4.

We computed the Cohen's kappa score between each pair of annotators on a subset of 100 texts (20 instances randomly chosen from the four dataset subsets, WebNLG train, dev, test and KELM). The scores

12. In case some of the randomly chosen permutations are identical, we remove duplicates.

13. Note, that this potentially introduces a bias as false negatives of the automatic detection won't be included in the manually annotated dataset. This is, however, the best solution we could find to focus the annotation work on texts with omissions and distortions and avoid spending annotation time on correct texts.

An automatically generated text is given, as well as input entities. For each input entity, please choose between 3 options :
- *Mentioned : the entity is mentioned in the text*
- *Not Mentioned : the entity is not in the text*
- *Distorted : the entity is not in the text, but some "distorted" version of it is in the text*

Notes
- *Please, try not to pay attention to the lack of fluency, grammar or meaning of some texts. What is important for us is whether an entity is present in a text or not, independently of the logic or quality of the text.*
- *Variants or Referring Expressions of an entity are considered as mentions. In other words, an entity doesn't have to be expressed in the exact same way as in the input data, e.g. 125800 mm / 125 m, 1703-05-27 / May 27th 1703, Appleton,Winsconsin / Appleton.*
- *Disambiguation information of entities is sometimes given inside parentheses, like for example in The_Honeymoon_Killers_(American_band) or Federal_Assembly_(Switzerland). This information doesn't necessarily have to be in the text.*
- *Missing accents, missing spaces and capitalization problems are small mistakes and can be considered as mentions, e.g. "Cremazie station" can be considered as a mention of the entity "Crémazie station"*

FIGURE 3.2 – Instructions given to the annotators

vary from 0.56 to 0.69, which is considered to be substantial agreement (Artstein and Poesio, 2008).

All texts (71,644 instances) were automatically annotated for omission and 12,886 of these were manually annotated for both omission and distortion. Table 3.1 reports the total number of manually and automatically annotated texts for each dataset, the number of texts which have at least one omission/distortion and the total number of omissions/distortions.

	All	In Domain Graphs		OOD Graphs	
		WNLG Train	WNLG Dev	WNLG Test	KELM
Automatic annotation					
Nb of texts (%)	71,644 (100%)	35,373	4,534	6,367	25,370
Nb of texts with O	33,160 (46%)	5,526 (16%)	661 (15%)	3,440 (54%)	23,533 (93%)
Nb of Os	74,462	10,006	1,110	5,691	57,655
Manual annotation					
Nb of texts (%)	12,886 (18%)	5,526	661	3,440	3,259
Nb of texts with O	6,249 (48%)	1,285 (23%)	170 (26%)	2,146 (62%)	2,648 (81%)
Nb of Os	9,096	1,374	176	3,087	4,459
Nb of texts with D	6,518 (51%)	699 (13%)	96 (15%)	2,556 (74%)	3,167 (97%)
Nb of Ds	9,034	734	104	4,746	3,450
Nb of texts with O or D	8,508 (66%)	1,892 (34%)	255 (39%)	3,194 (93%)	3,167 (97%)
Nb of Os and Ds	18,130	2,108	280	7,833	7,909

TABLE 3.1 – **Omission and Distortion Statistics for the texts generated by the BART RDF-to-Text Model** (O : ommissions, D : Distortion, WNLG : WebNLG). Unsurprisingly, omissions are more numerous on the data not seen at training time (OOD Graphs)

Text 1 : *Ciudad Ayalatus is governed by the Council-manager government, has a population density of 1604.0 and its city manager is the leader. The population density for this city is 177539.0.*
Entities annotations :
Ciudad_Ayala : Distorted
1777539 : Distorted
Council-manager_government : Mentioned
1604.0 : Mentioned
City : Mentioned
"City Manager" : Mentioned

Text 2 : *Olgaondarev died in Saint Petersburg which was founded on 27 May 1703 and has a total area of 1439.0.*
Entities annotations :
Saint_Petersburg : Mentioned
1703-05-27 : Mentioned
Olga_Bondareva : Distorted
1439.0 : Mentioned

Text 3 : *Pitcher Lee Lee-hak is a member of the Donosan Bears and the Vietnamese High School. He is also a citizen of the United States.*
Entities annotations :
Lee Jae-hak : Distorted
Doosan Bears : Distorted
Daegu High School : Not mentioned
Pitcher : Mentioned
Human : Not mentioned

Text 4 : *Edwin E.Aldrin,Jr. was a crew member of Apollo 11 and his alma mater was Massachusetts Institute of Technology.*
Entities annotations :
Buzz_Aldrin : Not mentioned
Edwin E. Aldrin Jr : Mentioned
Apollo 11 : Mentioned
Massachusetts Institute of Technology, Sc D 1963 : Distorted

FIGURE 3.3 – Examples given to the annotators

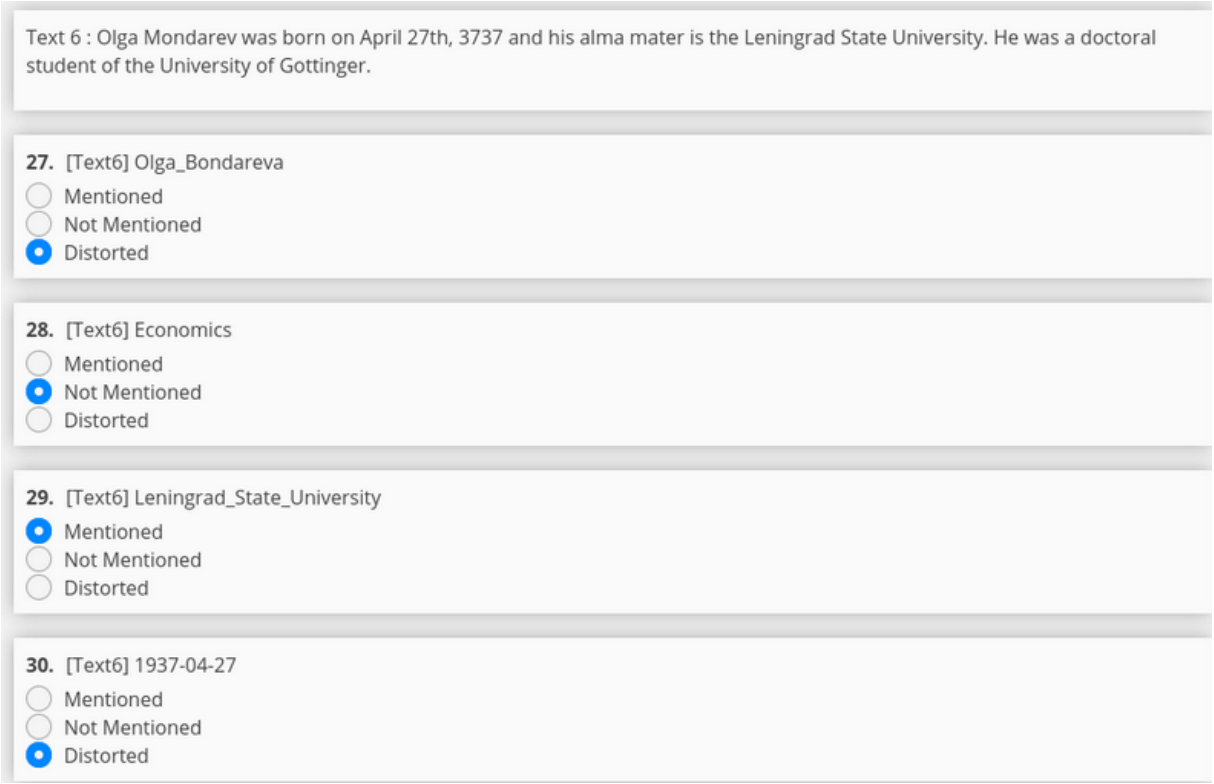


FIGURE 3.4 – Annotation Example on CryptPad

3.3.4 Data for the probing experiments

For our probing experiments, we keep annotated texts with at least one omission and divide them into train (70%), dev (15%), and test (15%) sets. Of the distinct entities which are omitted or distorted, approximately 50% occur in the train split, and 25% occur in the dev and test splits. Hence, the dev and test splits contain omissions and distortions of entities which are not seen during training.

3.3.5 Evaluation of Automatic Annotation

To assess the quality of the automatic annotations, we compare them with the manually labeled omissions [14]. The F-measure is 0.58 (Precision : 0.52, Recall : 0.66). We therefore cannot consider the automatic annotation as completely reliable for our probing experiments. We use it in two ways : (i) as a way to speed up the annotation process as described in subsection 3.3.3 and (ii) as an addendum to our main experiments on the manually annotated data.

3.3.6 Exploring the possible role of decoding strategies

The hypothesis we test in this chapter is that omissions and distortions are substantially due to problems with encoding. However, it is possible that the decoding strategy used also plays a role. To check whether decoding impacts omissions, we compare four decoding strategies : greedy, beam search, top-k (Fan et al., 2018) and top-p (i.e. nucleus sampling, (Holtzman et al., 2020)), with k=50 and p=0.9 (which are standard values for top-k and top-p decoding). We generate texts from RDF graphs from the

14. We only compare to omissions, not distortions, as chapter 2's detection algorithm aims to detect omissions only.

WebNLG test set (1,779 graphs) using each of these four strategies ; we then use the automatic detection of omissions on the generated texts. For each text, we compute the Intersection over Union of omitted entities between decoding strategies (we report the mean and median of IoU across the texts in Table 3.2, in the rows marked 'Auto'). Even though some differences emerge, we find that the entities omitted by BART remain to a large extent similar across the four different decoding strategies. We also find that different decoding strategies result in very similar numbers of texts with at least one omission (844 for greedy decoding, 985 for beam search with 5 beams, 980 texts for top k with k=50 and 974 for top p with p =0.9).

As discussed in section 3.3.5, the automatic annotation has two main limitations : it is noisy and it does not distinguish omissions from distortions. In order to have a more reliable and fine-grained analysis of the impact of the different decoding strategies on omissions and distortions, we manually annotated the texts generated from 500 input graphs (taken from the WebNLG test set) using the four decoding strategies, which yields a total of 2k manually annotated (graph, text) pairs. We then computed the IoU scores for omissions and distortions on this data. The results are reported in Table 3.2 (rows marked 'Manual-O' and 'Manual-D'). The results for the manually annotated data are similar to the results on automatically annotated data. The overlap of omitted entities between the different decoding strategies is quite high, with a mean IoU score across texts ranging from 0.53 to 0.63. For distorted entities, the overlap is even higher, with mean IoU between 0.69 and 0.81. This suggests that the impact of the decoding strategy on omissions is limited, and is even more limited on distortions. Nevertheless, the potentially limited role of the decoding strategy is worth investigating in more detail ; we leave this as a topic for future work.

		Greedy	Beam	Top k
Beam	Auto	0.61/1.0		
	Manual-O	0.62/1.0		
	Manual-D	0.81/1.0		
Top k	Auto	0.52/0.5	0.73/1.0	
	Manual-O	0.53/0.5	0.56/1.0	
	Manual-D	0.69/1.0	0.71/1.0	
Top p	Auto	0.53/0.5	0.73/1.0	0.76/1.0
	Manual-O	0.57/1.0	0.63/1.0	0.63/1.0
	Manual-D	0.72/1.0	0.74/1.0	0.79/1.0

TABLE 3.2 – Mean/Median of Intersection over Union scores for each text of omitted/distorted entities (using automatic annotation, on 7116 texts), of omitted entities (using manual annotation of 2000 texts), of distorted entities (using manual annotation of 2000 texts) for different decoding strategies

The fact that omissions and distortions remain roughly the same when we modify the decoding strategy suggests that the decoding strategy is probably not the main cause of omissions. We should therefore look for the cause of omissions in some other part of BART. In this chapter we choose to focus on the encoder. For simplicity and as the decoding strategy does not seem to exert much impact on omissions, in what follows we only consider greedy decoding.

3.4 Method

To study omissions and distortions, we choose probing rather than some other analysis methods as it provides a direct, model-agnostic and relatively computationally inexpensive way to study a model internals. Furthermore, as discussed in section 3.2, probing has been shown to be effective for analyzing how linguistic phenomena are handled in neural architectures.

We design two probes to test whether omissions can be detected from encoder representations. The first probe is a parameter-free probe which tests whether the encoding of a graph leading to an omission contains less information about an omitted entity than about a mentioned entity (an entity that is mentioned in the output text).

The second probe is a standard probing classifier which seeks to distinguish the encoded representations of input graphs leading to an omission from the representations of input graphs whose output text correctly verbalizes all input entities.

While probing classifiers have been widely used to analyze neural networks, Hewitt and Liang (2019) observe that a probe might memorize the data it is trained on rather than evaluate the information present in the internal representations it is designed to analyze. The parameter-free method we propose in Section 3.5 helps address this issue as it extracts information directly from the internal representations without learning new parameters. For the parametric probing classifier of Section 3.6, we address this issue by introducing a control task which can only be resolved by memorising the data, and computing selectivity i.e., the difference between the probe performance on the control task and on the probing task. We show that our probing classifier has both high performance, which indicates that omissions can be detected from the encoded input graphs and high selectivity, which indicates that the probe is learning rather than memorising the data.

3.5 Parameter-free Probing

This method is based on the intuition that the encoder representations of RDF graphs which lead to omission have a weak signal for the omitted entity. We hypothesize that, because it lacks specificity, the representation of an omitted entity is more similar to the representation of the unknown token <unk> than the representation of an entity that is correctly verbalized in the output text. We test this hypothesis by comparing the similarity between g, the encoder representation of a graph leading to an omission with two alternative representations : (i) $g^{\setminus o}$, the embedding of the same RDF graph in which an omitted entity o has been replaced with the unknown special symbol <unk>; and (ii) $g^{\setminus m}$, the embedding of this graph in which a mentioned entity has been replaced with <unk>. If the encoded representation of omitted entities lacks specificity, we expect the embedding of g to be closer to the encoding of $g^{\setminus o}$ (where an omitted entity has been replaced with <unk>) than to that of $g^{\setminus m}$ (where a mentioned entity has been replaced with <unk>).

Let G be the set of graphs in our data which have at least one mention and one omission. For each graph $g \in G$ which is associated with J_g omissions or distortions and K_g mentions, we denote $\{o_j\}_{j=1,\ldots,J_g}$ the omissions/distortions and $\{m_k\}_{k=1,\ldots,K_g}$ the mentions in g.

For each graph embedding $g \in G$, we compute an average of similarities between g and $g^{\setminus m_k}$:

$$sim(g, g^{\setminus M}) = \frac{1}{K_g} \sum_{k=1}^{K_g} sim(g, g^{\setminus m_k})$$

and similarly for the average of similarities between g and $g^{\setminus o_k}$. We then compute the proportion of texts in our dataset for which :

$$sim(g, g^{\setminus o_j}) > sim(g, g^{\setminus M}) \,,$$

where, for two embeddings g and g',

$$sim(g, g') = cos(mean(g), mean(g'))$$

is the cosine similarity on fixed-sized graph embeddings obtained by mean pooling as fixed-size embeddings are needed to compare embeddings of graphs of arbitrary length using cosine similarity. Various pooling strategies can be used. For instance, Reimers and Gurevych (2019) experiment with three strategies : using the output of the CLS token, and max or mean pooling of all the output vectors. As they find that mean pooling performs best, we experiment with two different mean pooling strategies. The output of the encoder is a matrix of shape $T \times 1024$, with T the number of tokens. We can average it either row-wise (i.e. dimension-based averaging) and obtain a mean vector of length 1024 or column-wise (i.e. token-based averaging) and obtain a mean vector of length T. As the two averaging strategies give similar results, for simplicity we report only the dimension-based averaging results.

	All	**In Domain**			**OOD**	
		W-T	W-D	W-S	W-U	K
Manual						
O	0.68	0.64	0.72	0.61	*0.52*	0.77
D	0.44	0.70	0.68	0.47	0.45	0.47
Auto	0.66	0.83	0.85	0.56	0.44	0.65

TABLE 3.3 – **Proportion of graphs for which** $sim(g, g^{\setminus o_j}) > sim(g, g^{\setminus M})$, O : Omissions, D : Distortion, W : WebNLG, T/D/S/U/K : Training/Development/Seen/Unseen/Kelm Data. The figures in gray correspond to non-statistically significant results. A proportion of 50% indicates failure to distinguish omissions/distortions from mentions. Proportions larger (resp. smaller) than 50% indicate that omissions/distortions can be distinguished from mentions and that our assumption that encodings leading to an omission/distortion have a weaker signal than the ones leading to a mention is supported (resp. contradicted).

Results Table 3.3 shows the proportion of examples in which $sim(g, g^{\setminus o_j})$ is greater than $sim(g, g^{\setminus M})$ for the different subsets of our data. For statistical significance testing, we used a one way chi-square goodness-of-fit test comparing the numbers of samples in each subset of the data for which $sim(g, g^{\setminus o_j}) > sim(g, g^{\setminus M})$, to those where this inequality does not hold. As we did multiple tests for each subset of the data, we adjusted the p-values using Bonferroni correction.

On average, the proportion of graphs for which $sim(g, g^{\setminus o_j}) > sim(g, g^{\setminus M})$ is 66% for the automatically annotated data and 68% for the manually annotated data. Most results are statistically significant which supports our hypothesis that the encoding of a graph leading to an omission has weak signal for that entity.

The subsets of the data on which this proportion is the lowest and the results are not statistically significant are the subsets of the data unseen during training, the WebNLG Test Set and KELM. This trend is even clearer on the subset of the WebNLG test set with entities and (DBPedia) categories unseen during the model's fine-tuning (W-U in Table 3.3). We hypothesize that this is because graphs which are different from the graphs seen during the fine-tuning of the generation model are less distinctively encoded that the ones seen during fine-tuning.

The results also show a clear difference between distortions and omissions. This suggests that the encodings of distortions and omissions are qualitatively different and that distortions are not merely a type of omission.

Finally, we observe that while there is a difference between results on the automatically and manually annotated data, the delta is relatively small (0.66 on avg for the Auto data vs. 0.68 for the manual data), and the overall trend is the same – which indicates that 2's omission detection algorithm performs well enough on out-of-domain data to be used for such experiments.

In this section, we showed that a parameter-free probing method, based on cosine similarities, can be used to probe omissions and distortions in the encodings of RDF graphs.

3.6 Parametric Probing : Binary classifiers

Our second probe is a binary classifier which takes as input the encoder representation of an RDF graph g and of an entity $e \in g$, returning 0 if e is omitted or distorted in the output text, and 1 otherwise. This can be viewed as modeling an entailment relation between a graph representation and an entity : the classifier returns 1 if the graph entails the entity (mention) and 0 if it does not (omission or distortion).

In this section we investigate whether neural binary classifiers can be used to detect omissions and distortions from RDF graph encodings. In subsections 3.6.1 and 3.6.2, we present our probes and the evaluation metrics we use. In subsections 3.6.3 and 3.6.4, we introduce an upper bound and two control tasks which help get better insight into the maximal performance our probes could reach, as well as address the risk that our probes are not really using the information in the embeddings, but instead rely on patterns linked to the entities' identity. In subsection 3.6.5, we discuss and compare the results of the probe, of the upper bound and of the control probe.We also report results on hard examples (entities that can be both mentioned, omitted, or distorted) and provide additional evidence that the probing results do not rely exclusively on entities' identity. In addition, we include two subsidiary experiments : one aiming at comparing the results of the probes trained on the automatic versus the manually annotated dataset and another which investigates the difference between omissions and distortions. The results show that probes trained on detecting omissions do not generalize well to distortions (and vice versa).

3.6.1 Models

Our second probe is a neural classifier with fully-connected linear layers and sigmoid activation functions trained using a cross-entropy loss and the AdamW optimizer. It takes as input the average vector of the embedding matrix (dimension-based averaging, as in Section 3.5). We experiment with both a single- (N1) and a two-layer (N2) network, using the train and test sets defined in 3.3.4. For each network, we train a model and perform hyper parameter tuning for each dataset (Manual-O+D, Manual-O, Manual-D and also Auto) [15]. We tune hyperparameters of the neural classifiers trained on Manual-O, Manual-D, and on Manual-O+D. We use grid search with batch size (8, 16, 32, 64, 128, 256), initial learning rate (0.1, 0.01, 0.001, 0.0001), and size of the second layer (1000, 500, 100, 50, 10) for the 2 layer classifier. We choose the classifier that maximizes the f-measure of class 1. In the table 3.4, we report the hyperparameters and the f-measure on the validation set of the best classifiers.

3.6.2 Evaluation metrics

Our data is unbalanced as it contains more mentioned entities than omitted or distorted ones. Since we are primarily interested in detecting omissions or distortions, which we define as class 0 in our binary classification, we evaluate the results of our probes using the F-measure for class 0. We also report the

15. We also experimented with SVM and Decision Tree classifiers, which performed slightly worse than neural ones. For brevity, we report only the results of the neural classifiers.

Model	Batch Size	Size L2	Lr	F1
N1, HP-O	128	-	0.001	0.91
N1, HP-O+D	64	-	0.001	0.90
N1, HP-D	256	-	0.01	0.91
N2, HP-O	128	1000	0.01	0.93
N2, HP-O+D	16	100	0.001	0.93
N2, HP-D	32	1000	0.001	0.93

HP : hyperparameters, O : omissions, D : distortions
Lr : initial learning rate, L2 :second layer of the classifier

TABLE 3.4 – Hyperparameters of the best classifiers

balanced accuracy (B.Acc), defined as :

$$\frac{1}{2}\left(\frac{TP}{P}+\frac{TN}{N}\right),$$

where TP is the number of true positives, P, the number of positive examples, TN the number of true negatives and N, the number of negative examples.

As suggested in Belinkov (2021), we compare our probes to both an upper bound and a control task.

3.6.3 Upper Bound

Entities that are not present in the input graph can be seen as an extreme case of omission. Therefore a natural upper bound for our probing classifier is a classifier which seeks to distinguish mentioned entities from entities not present in the input graph. We train such a classifier on 18k RDF graphs (randomly selected from our dataset) and 198k entities (which are either mentioned in the RDF graph or randomly selected from the set of all entities in the dataset and that are not in the input RDF graph). Note that the training data for this classifier is different from that used for our omission probes.

The classification results for this upper bound, which are reported in Table 3.5 (last two lines), are high for both N1 (F1 : 0.91) and N2 (F1 : 0.97), showing that it is possible to detect whether or not an entity is present in the embedding of an RDF graph. Furthermore, the comparatively lower scores obtained when probing for omissions (F1 : 0.69) and distortions (F1 : 0.79) suggests that omissions and distortions are harder to detect than entities not present in the input graph.

3.6.4 Control tasks

We implement two control tasks to verify that our probes are predicting omissions from graph embeddings rather than memorising the training data.

Training on randomized probing labels Following Belinkov (2021), we randomize the labels in the training set and compute the F-measure and Weighted Accuracy of a control probe, trained on the randomised dataset. Table 3.5 shows that there is a clear drop in performance when training the probes with randomised labels. We can therefore conclude that our probes are not just memorising the training set.

Testing with randomized NLG model encoder The second control task consists in randomizing the weights of the encoder. We randomly initialize the weights of the BART encoder and train and test the neural probe with two linear layers (our best probe) on the manually annotated dataset for omissions. We try 5 different random seeds and obtain a mean B.Acc of 0.49 (with standard deviation 0.02) and mean F-measure for class 0 of 0.31 (with standard deviation 0.13). This significant performance drop shows that the probing results are not based on mere word identity (which could be captured by random encodings).

3.6.5 Results

	N1	N2
Manual-O+D		
F1 (B.Acc)	0.73 (0.78)	**0.82 (0.85)**
C_{F1} ($C_{B.Acc}$)	0.00 (0.50)	**0.00 (0.50)**
Manual-O		
F1 (B.Acc)	0.60 (0.72)	**0.69** (0.77)
C_{F1} ($C_{B.Acc}$)	0.00 (0.50)	**0.00 (0.50)**
Manual-D		
F1 (B.Acc)	0.71 (**0.88**)	**0.79** (0.79)
C_{F1} ($C_{B.Acc}$)	0.00 (0.50)	**0.00 (0.50)**
Auto		
F1 (B.Acc)	0.60 (0.71)	**0.79 (0.83)**
C_{F1} ($C_{B.Acc}$)	0.00 (0.5)	**0.00 (0.50)**
Upper-Bound		
F1 (B.Acc)	0.91 (0.92)	**0.97 (0.98)**

TABLE 3.5 – F-measure of class 0 (F1), Balanced Accuracy (B.Acc) for each probe and its control $C_{F1}$and $C_{B.Acc}$. N1/N2 : One/Two Layer Network (with hyperparameters tuned on Manual-O, Manual-D or Manual-O+D). In bold, the best results for each dataset.

Best Probes Table 3.5 shows the F-measure scores and balanced accuracies for the various probes on the different datasets. Overall, we find (i) that the two-layer neural network performs best and (ii) that models trained on manually annotated data perform better than those trained on automatically annotated data.

Detecting Omissions and Distortions Focusing on the subset of the data that was manually annotated (Manual-X, Table 3.5), we find that the parametric probes successfully classify omissions (F1 : 0.60, 0.69) and distortions (F1 : 0.71, 0.79) indicating a significant difference between the embeddings of mentioned entities on the one hand and the embeddings of distorted/omitted entities on the other hand.

Table 3.6 gives a more detailed picture of the results for the best probe (N2) on the different subsets of the manually annotated dataset. For statistical significance testing, we used a chi-square test of independence. Once again, we use a Bonferroni correction for the p-values, since we did a test for each of the test data subsets. All results are statistically significant. For most subsets, we observe a similar trend as on the full datasets (Table 3.5) : probing for omissions and distortions together (Manual-O+D) yields the best F-measure on class 0 (omission and distortion) and probing for distortions (Manual-D)

	All	In Domain			OOD	
		W-T	W-D	W-S	W-U	K
O+D						
F1	**0.82**	0.71	0.64	**0.85**	**0.86**	**0.82**
B.Acc	0.85	0.80	0.78	0.87	0.97	0.83
O						
F1	0.69	0.57	0.57	0.71	0.69	0.73
B.Acc	0.77	0.71	0.71	0.78	0.77	0.79
D						
F1	0.79	**0.80**	0.83	0.82	0.79	0.78
B.Acc	0.84	0.85	**0.88**	0.86	0.83	0.83

TABLE 3.6 – F-measure of class 0 (F1) and Balanced Accuracy (B.Acc) on different subsets of the probing test set using N2. In bold, the best results for each dataset.

yields better results than probing for omissions (Manual-O). These results are different from the ones obtained with the parameter-free probe, where omissions were easier to probe than distortions. This can be explained by differences between the two probing methods : the parameter-free probing considers only differences of cosine similarities between embeddings, while the neural probes look at non-linear differences between embeddings. These results suggest that the neural probes are complementary to the parameter-free probing and better suited for the analysis of distortions.

Testing on Hard Examples In different texts, the same entity can be mentioned, omitted or distorted. Such cases permit testing whether our probe accurately classifies graphs that contain omissions and distortions rather than graphs that contain specific entities. We test our best probe (N2) on such entities and report the results in Table 3.7. Comparing these results with those on all entities (Table 3.5), we see that both the F-measure of class 0 and the balanced accuracy are similar for all manually annotated data (and even higher on the manually annotated omissions). Only on the automatically annotated dataset are the results slightly lower. These results show that the probe also performs well on difficult examples.

Training Data	Test Data	%	F1 (B.Acc)
Manual-O	M&O	13%	0.81 (0.74)
Manual-D	M&D	14%	0.84 (0.81)
Manual-O+D	M&O&D	9%	0.78 (0.82)
Manual-O+D	M&O	13%	0.82 (0.82)
Manual-O+D	M&D	14%	0.78 (0.81)
Auto	M&O	31%	0.7 (0.63)

TABLE 3.7 – Results of the N2 probe on entities that are mentioned and omitted (M&O), mentioned and distorted (M&D) or mentioned, omitted and distorted (M&O&D). The third column gives the proportions of such entities in the manual and the automatic dataset.

Correlations between Models trained on Automatically vs. Manually Annotated Data We want to know to what extent the automatically annotated data, which is cheap to obtain in comparison with manually annotated data, leads to the same probing results, compared to the manually annotated data. To answer this question we compute the Spearman correlation coefficient between the labels predicted by

the probes on the automatic and manual datasets. We also compute the Pearson correlation coefficient between the probabilities of the class 0 predicted by the probe for all the test set examples on the automatic and the manual datasets. The correlation between the labels tells us how well the predicted classes are correlated and the correlation between the probabilities of the class 0 additionally gives an indication about whether the two probes have similar levels of confidence in their predictions.

The results are shown in Table 3.8. The correlation strength between the manually labeled omissions and the automatic data both for the labels and for the probabilities of class 0 is high. However, there is no correlation for the labels or for the probabilities of class 0 for distortions identified from manual and from automatic data. This can be explained by the fact that the automatic mention detection from 2 includes approximate string matching. Since approximate matching uses a threshold, a distortion which overlaps substantially with the entity would be counted as a mention and will therefore be labeled as belonging to class 1 by the probing classifier trained on automatic data, whereas it will be labeled as class 0 by the probing classifier trained on the manual dataset.

		Correlation coefficient/P-value		
		O	D	O+D
N1	Sp	0.70/0.00	0.07/0.10	0.55/0.00
	Pe	0.77/0.00	0.02/0.70	0.54/0.00
N2	Sp	0.70/0.00	0.07/0.10	0.55/0.00
	Pe	0.99/0.00	0.01/0.08	0.59/0.00

TABLE 3.8 – Correlation between models trained on Manual vs Automatically annotated data (Sp : Spearman on labels, Pe : Pearson on probabilities of class 0). P-values are computed against the null hypothesis that the correlation is no different from zero.

Omissions vs Distortions Table 3.9 shows the results of the neural probes trained on the subset with only omissions (Manual-O), and tested on the subset with only distortions (Manual-D) and inversely, trained on the subset with only distortions (Manual-D) and tested on the subset with only omissions (Manual-O). The drop in results compared to training and testing on the same dataset shows that the probe trained on omissions does not generalize well to detecting distortions and vice versa. This is in line with the results of the parameter-free probing described in Section 3.5 and suggests that omissions and distortions are errors of a different nature – they can both be identified in the encoder output but seem to have qualitatively different representations.

		Training Data	
		Manual-O	Manual-D
	Test Data		
1 Layer	Manual-O	0.60 (0.72)	0.26 (0.5)
	Manual-D	0.19 (0.49)	0.71 (0.79)
2 Layers	Manual-O	0.69 (0.77)	0.27 (0.51)
	Manual-D	0.22 (0.48)	0.79 (0.79)

TABLE 3.9 – F-measure of class 0 (Balanced accuracy) when training the neural classifiers on Omissions and testing on Distortions and vice versa

Tripleset Features	Number of triples in the tripleset Category of the tripleset (in WebNLG : e.g. 'Athlete', 'Scientist', 'Food') Position of the first occurrence of the entity in the tripleset Number of occurrences of the entity in the tripleset Nature of the entity in the tripleset (i.e. agent, patient or bridge as defined by Castro Ferreira et al. (2018))
Entity Features	DBPedia entity type Number of characters Whether the entity is a date Whether the entity is a number Entity shape (uppercase, lowercase, digits, other) Entity shape (vowels, consonants) as used in van der Goot et al. (2018)
Dataset Feature	Frequency of the entity in the training set of WebNLG

FIGURE 3.5 – Logistic Regression Features

3.7 Dataset Analysis using Logistic regression

We train a logistic regression classifier to predict whether an entity will be omitted or distorted based on data-specific features (the 12 input features we use are detailed in the Figure 3.5 and are about the tripleset, the entity itself and the frequency of the entity in the dataset). If such a classifier proves highly predictive, the feature weights can help understand which features impact the omission or distortion of an entity.

We compute the logistic regression on Manual-O, -D and -O+D datasets, randomly choosing 90% of the omissions or distortions as training set and the remaining 10% as test set (as the results were unstable, we repeated the training and test with three different random seeds and report the mean). The results are shown in Table 3.10. The logistic regression does not perform well on the Manual-O dataset but performs better on Manual-D and Manual-O+D. This shows that whereas omissions seem difficult to characterize based on dataset features, features describing distortions are easier to find. In particular, we find that the position of the first occurrence of the entity in the tripleset, the number of occurrences in the tripleset, the semantic role in the graph (subject, object or both) and the length of the tripleset are the most relevant features for a distortion to happen.

	F1	
	Train	Test
Manual O	0.36	0.37
Manual D	0.60	0.59
Manual O+D	0.72	0.76

TABLE 3.10 – F-measure of class 0 of the logistic regression on the different datasets

In this section, we showed that a logistic regression classifier trained on dataset features does not perform well on detecting omissions, suggesting that information for detecting them can only be found in more complex non-linear features. The same logistic regression classifier seems to indicate that some dataset features are correlated with distortions.

3.8 Study of a T5 encoder

We claim that our probing methods are model-agnostic and can be used to analyze any encoder in a transformer-based NLG model. To support this claim and check whether our results can generalize to another encoder, we carry out experiments on T5 (Raffel et al., 2019). As with BART, we fine-tune T5-small on the training set of WebNLG (the fine-tuning details are given in section 3.3.1). We then generate texts, filter out duplicates and automatically detect omissions as described in Section 3.3. Table 3.11 shows the statistics of the resulting corpus. Note that we did not create a manually annotated corpus for T5. The results are therefore limited to automatically detected omissions. We probe the T5 encoder using the parameter-free probing method described in Section 3.5 and the 2-layer neural probing classifier described in Section 3.6 (we use the same hyperparameters as the probing classifier N2-O+D). The results are reported in Table 3.12.

The results for the probing classifier are similar to those obtained for the BART encoder, with an F-measure of class 0 of 0.8 and a Balanced Accuracy of 0.85 (compared to BART : F1=0.82 and BAcc=0.85). This suggests that our probing method generalizes well to the T5 encoder.

The results for the parameter-free probing are much better with T5 than with BART which suggests that the applicability of this parameter-free probing depends on the type of embeddings/encoder studied. This should be studied in future work to determine how the choice of the model and of the fine-tuning parameters impact the performance of this probing method.

In conclusion, our two probing methods can be used to analyze omissions in the embeddings of another transformer-based encoder.

	# T	# T(O)	# O
WebNLG			
Train	36,704	7,064 (19%)	7,824
Dev	4,658	882 (19%)	993
Test	6,173	2,286 (37%)	2,855
KELM	24,963	17,852 (72%)	29,596
ALL	72,498	28,084(39%)	41,268

TABLE 3.11 – **Corpus Statistics for texts generated by T5** (T : Texts, T(O) : Texts with Omissions, O : Omissions)

3.9 Limitations and Future Work

NLG Task studied In this chapter, we restrict our study to NLG models that generate English text from RDF data. In future work, our study could be extended to other Data-to-Text or Text-to-Text tasks as well as to other languages. A key bottleneck is the annotation of omissions and distortions, in particular whether it should be automatic or manual.

Scope of the study We study two state-of-the-art models for Data-to-Text, BART and T5, and probe their encoders. Further work in that direction could for instance study the impact of *different fine-tuning* of the same encoder and of encoders other than T5 and BART's. Another direction would be to look at *different parts of the model*, i.e. not only the output of the encoder but also the different layers both in the encoder and in the decoder.

	All	In Domain			OOD	
		W-T	W-D	W-S	W-U	K
NP.P						
T5	0.89	0.84	0.84	0.88	0.81	0.91
BART	0.66	0.83	0.85	0.56	0.44	0.65
P.P						
F1						
T5	0.8	0.84	0.83	0.79	0.7	0.78
BART	0.69	0.57	0.57	0.71	0.69	0.73
B.Acc						
T5	0.85	0.88	0.88	0.83	0.77	0.81
BART	0.77	0.71	0.71	0.78	0.77	0.79

TABLE 3.12 – **Results of parameter-free (NP.P) and parametric (P.P) probing of the T5 encoder.** We also recall the results on the BART encoder. All the results are statistically significant results (using chi-square goodness-of-fit for NP.P and independence for P.P tests with Bonferroni correction with alpha=0.05). NB : The results NP.P and P.P are not directly comparable, as they are based on different metrics.

Studying how NLG models perform regarding omissions and distortions when trained on *different RDF-to-Text datasets* could also provide valuable insights on the causes for these mistakes.
Other analysis methods than probing can also be used. Indeed, an intrinsic limitation of probing is that it relies on correlations. Using *causal methods*, for example by performing interventions on the embeddings themselves, would be an important step for further understanding omissions in pretrained language models.

Subjectivity in Manual Annotation For the annotation task, we do not formally define a threshold between omissions and distortions ; rather, we let the annotators decide what they consider as omitted and distorted. The following examples show entities that annotators annotated differently in similar contexts.

Example (a) : The texts 1 and 2 are generated from a tripleset containing the triple Alan_Shepard | selectedByNasa | 1959. In Text 1 the annotator considered the entity "1959" as mentioned, whereas the annotator of Text 2 considered it as distorted.

Text 1 : Alan Shepard was an American born in New Hampshire. He graduated from NWC with an M.A. in 1957 and was selected by NASA the same year. He died in California.

Text 2 : American Alan Shepard was born in New Hampshire on November 18th, 1923. He graduated from NWC with an M.A. in 1957 and was hired by NASA the same year. He served as a test pilot and died in California.

Example (b) : The annotator of Text 1 considered the entity 'GMA_Network_Center' to be omitted, whereas the annotator of Text 2 considered it as distorted.

Text 1 : GMA New Media, Inc. (parent company GMA Network) is an entertainment industry. Philippine Entertainment Portal and Digify Inc. are just two of the company's subsidiaries.

Text 2 : GMA New Media, Inc. (parent company GMA Network) is located in the Philippines. The company is involved in the entertainment industry with Philippine Entertainment Portal and Digify Inc. It is located inside GMA Center.

Application to an improved omissions detection While our probing methods help assess whether omissions can be tracked back to the encoder, 2's algorithm permits detecting omissions based on a model-agnostic comparison between input graph and generated texts. These two methods could be combined to help facilitate the analysis of omissions in arbitrary RDF-to-Text generation models as follows. First, the omission detection algorithm can be applied to the output of the generation model to automatically annotate omissions. Second, these annotations can be used either as is or after manual validation, to fine-tune our probing classifier on the internal representations of the model under consideration. More generally, our methods help provide fine-grained information about both the quantity of omissions generated by a given model and the degree to which these omissions can be assumed to come from the encoder.

3.10 Conclusion

This work took off from the hypothesis that omissions and distortions in RDF-to-Text NLG models are due in large measure to issues with the way input entities are encoded. We collected the first dataset for the analysis of omissions and distortions in the output of BART-based RDF-to-Text model and introduced two probes for understanding which parts of the model internals such semantic errors can be associated with. In addition to a standard, parametric, probing classifier, we introduced a parameter-free method which is based on the similarity between the embeddings of graphs associated with omission or distortions vs. the embeddings of graphs corrupted by removing either the omitted/distorted entity or an entity correctly mentioned in the output text. Both methods support the hypothesis that the encoding of graphs associated with omissions and distortions differs from the encoding of graphs which are not associated with such errors. Thus, we conclude, in line with our hypothesis, that the encoder plays an important role in the omission and/or distortion of input elements.

4

Question Generation in Knowledge-Driven Tutoring Dialog

Contents

In this chapter, we explore question generation in the context of knowledge-grounded tutoring dialogs. Inspired by previous work on planning-based summarization, we present a question generation model which sequentially predicts from the dialog context, first a triple then a question. Our goal is to provide a simple adaptation of a model in order to make it more transparent. The generated triple provides an indication of why a question was generated, which is a step toward providing explanations of the model behavior. We build the model, with the aim of enabling its fine-grained evaluation. Indeed, we show that this approach allows for a detailed analysis of the model behavior as the generated triple can be used to evaluate well-formedness (is it a well-formed triple of the KB ?), relevance (is the generated triple in topic and new with respect to the triples already captured by the dialog context ?), semantic adequacy (does the generated question match the generated triple ?) and referring expression generation (does the question contain correct and non-ambiguous pronouns ?). We evaluate our approach on 37k test dialogs adapted from the KGConv dataset (Brabant et al., 2023) and we show how our approach permits comparing different models in terms of coherence and (co-reference based) cohesion. We further show that conditioning question generation on both a knowledge graph and the dialog context helps increase dialog coherence.

4.1 Introduction

Recent years have seen an increasing number of approaches aiming to ground dialog models in general purpose knowledge graphs either to avoid over-generic dialog turns Han et al. (2015); Sun et al. (2018) or to support information seeking dialogs Liu et al. (2019); Perez-Beltrachini et al. (2023). In this chapter, we focus instead on how knowledge can be used to support tutoring dialogs i.e., dialogs where knowledge is used to inform the generation of a series of questions intended to teach a student about a given topic. An example dialog is shown in Table 4.1.

T_1	(*Sitara Achakzai*, *field of work*, *feminism*)
Q_1	`What was the field of work of Sitara Achakzai?`
A_1	`feminism`
T_2	(*Sitara Achakzai*, *death manner*, *murder*)
Q_2	`What was the cause of death of Achakzai?`
A_2	`homicide`
T_3	(*Sitara Achakzai*, *birthplace*, *Afghanistan*)
Q_3	`Where was she born ?`
A_3	`Afghanistan`
T_4	(*Afghanistan*, *capital*, *Kabul*)
Q_4	`What is the capital of Afghanistan?`
A_4	`Kabul`
T_5	(*Afghanistan*, *lowest point*, *Amu Darya*)
Q_5	`What is the lowest point of Afghanistan?`
A_5	`Amu Darya`

TABLE 4.1 – Example of dialog in the KGCONV dataset (T :Triple, Q :Question, A :Answer)

We confront several challenges. First, the generated question must fit the dialog context in that it should be neither redundant nor irrelevant. Second, it should be factual i.e., it should bear on an attested fact. Third, it should be natural sounding, in particular, integrating the anaphors and elliptical constructions that are typical of conversational dialogs.

Inspired by recent work from Narayan et al. (2021, 2022), we propose an approach where question generation is controlled by triple generation. Specifically, we train the question generation model to first generate a KB triple and then continue generating the corresponding question conditioned on both the predicted KB triple and the dialog context. This has two key advantages.

First, it improves explainability as the triple conditioning question generation can be viewed as a strong indicator of why the model generated that question.

Second, it helps address a key issue for the evaluation of dialog models, namely, how to evaluate a generated turn given that at any time, multiple continuations of the current dialog are possible. By allowing for an analysis of the generated triple and (triple, question) pair, our approach helps address this issue as follows : analyzing the generated triple permits assessing both factuality (is it a well-formed triple of the KB ?) and relevance (is the generated triple in topic and new with respect to the triples already captured by the dialog context ?) while comparing the generated question with the predicted triple can further be used to assess the semantic adequacy of the question (does the generated question match the generated triple ?). In short, by training a model to generate both a triple and a question instead of only a question, we enable a fine-grained, automatic, and referenceless evaluation of the generated question.

Our contributions are as follows. Using the KGConv dataset, a dataset of 70K dialogs where each (question, answer) pair is grounded in a Wikidata triple, we demonstrate how our approach permits assessing factuality, relevance and semantic adequacy. We compare our model with a more traditional approach where question generation is conditioned on the dialog context only rather than conditioning on the context and on a predicted triple and analyze their difference using both automatic metrics and human evaluation. Using an ablation study, we also highlight the impact of knowledge on dialog coherence : conditioning dialog response generation on a knowledge graph (in addition to dialog context) drastically helps reduce the proportion of generated questions that are either off topic or factually incorrect. All data, models, and scripts to reproduce our results will be publicly available.

The chapter is structured as follows. Section 4.2 describes the KGConv dataset, Section 4.3 and 4.4 introduce the generation task, our models and our experimental setting. Sections 4.5 and 4.6 present the results of the evaluation and of the ablation study.

4.2 Dataset

The KGConv dataset consists of 70,596 English dialogs such that each dialog is composed of a sequence of question-answer pairs and each question-answer pair is grounded in a triple (subject, property, object) whose object is the expected answer. For example, the question-answer pair "Q : Where was Obama born ?, A : Hawaii" is associated with the triple (Obama, place of birth, Hawaii). Statistics of the dataset are given in Table 4.2.

	entities	properties	dialogues	questions			
				train	dev	test	total
people	31324	318	25838	184259	29150	11306	224715
country	2039	163	630	4440	716	214	5370
religion	1209	165	493	3418	643	197	4258
astronomical object	2589	119	5961	0	0	50158	50158
molecular entities	17807	154	23067	154725	24632	9573	188930
historical event	4905	186	5274	37521	6040	2380	45941
food	2403	168	1992	14277	2130	930	17337
taxon	3187	213	1902	0	0	16099	16099
with unseen properties	13469	405	5439	0	0	51097	51097
total	63358	457	70596	398640	63311	141954	603905

TABLE 4.2 – KGConv statistics, from the paper Brabant et al. (2023). For each theme, the table gives : the number of different entities and properties appearing in conversations or dialogues, the number of dialogues, and the number of questions for each split. Note that in the entities and properties columns, the "total" values are not the sum of the cells above ; this is because some entities and properties appear in several themes.

Each dialog D is associated with a root entity e which is the subject entity of the triple initiating the dialog, the Wikidata category T_e of that entity and the subgraph K_D of Wikidata triples associated with the question-answer pairs making up D. The number of questions in a dialog is at least 5 and at most 19.

We write $\mathcal{K}_{KGConv}$ the set of all Wikidata triples (143K triples) grounding the KGConv dialogs ($\mathcal{K}_{KGConv} = \cup_{D \in KGConv} K_D$).

4.3 Problem Formulation

Given $K = \{f_1, \ldots, f_k\}$, a knowledge graph representing a set of triples, and D a dialog context, the task consists in generating an adequate follow-up question q which is grounded in a fact $f \in K$. In particular, the goal is to generate a question which is coherent with D (i.e., neither repetitive nor off topic) and natural sounding (i.e., using anaphors whenever appropriate).

To facilitate learning as well as to enhance relevance and explainability, we propose a model which grounds the generation of the next question q (e.g., *Where was Henri Poincaré born ?*), not only on D and K, but also on the corresponding fact (e.g., (*Henri_Poincare*, *birth_place*, *Nancy*)). That is, instead of directly decoding a question q from D and K, we train an encoder-decoder which decodes the concatenation of f_q and q where f_q is the fact underlying the semantics of q. Since KB triples exhibit less surface variability than natural language, we hypothesize that learning to predict a fact (in the form of a KB triple) is easier than learning to predict the corresponding question and further, that conditioning the generation of the question on a given triple helps the model learn to generate well-formed and relevant questions.

To assess generalization and study the impact of the dual information (dialog turns and associated KB facts) present in the training data, we experiment with different knowledge sources and dialog contexts, which we define in the following two paragraphs.

Knowledge Graph. In our tutoring dialog setting, each generated question should be grounded in a fact. This raises the issue of content selection : given some knowledge graph K, to what extent does the model learn to select a fact that is both relevant and non-redundant with respect to the current dialog context ?

To study the ability of our model to select a relevant fact from the Knowledge Graph, we train and test on knowledge graphs of varying size and relevance. Specifically, we condition generation either on K_D, the set of triples associated with dialog D in the KGConv dataset or on K^+, a larger set of triples which includes in addition to K_D ($K_D \subset K^+$), three types of "distractor" triples not in K_D :

- Out-of-Scope triples (entity) : triples in $\mathcal{K}_{KGConv}$ that are not in K_D and whose subject is of the same Wikidata category as the root entity of the dialog.
- Out-of-Scope triples (property) : triples in $\mathcal{K}_{KGConv}$ that are not in K_D but whose property appears in K_D.
- Noise triples : triples that are not in $\mathcal{K}_{KGConv}$ but whose subject, property and object are in $\mathcal{K}_{KGConv}$.

The motivation for these additional triples is that they act as "distractors" for content selection where the three types aim to capture different types of similarity with the relevant triples (all triples in K_D) and therefore different cases for content selection. While Noise triples are markedly distinct from the set K_D of triples associated with the dialog at hand, Out-of-Scope triples are semantically close to triples in K_D and are therefore harder to filter out.

We denote by K_n^+ the set K_D with n additional triples of each type (i.e., $K_0^+ = K_D$ and K_1^+ has three more triples than K_D). We experiment with $n = 0, 1, 2, 3$, capping the total number of added triples to the size of K_D (there are at most as many noisy and out-of-scope triples as relevant ones). An example of the resulting K_1^+, K_2^+ and K_3^+ graphs is given in Table 4.3.

K_D	$\mathbf{K_1^+}$
(*Alberto Moravia, inv. founded by, Nuovi Argomenti*)	(*Alberto Moravia, inv. founded by, Nuovi Argomenti*)
(*Elsa Morante, cause of death, myocardial infarction*)	(*Elsa Morante, place of birth, Rome*)
(*Rome, inv. enclave within, Vatican City*)	(*Elsa Morante, manner of death, natural causes*)
(*Elsa Morante, place of birth, Rome*)	(*Friedrich Fulleborn, place of death, Hamburg*)
(*Elsa Morante, manner of death, natural causes*)	(*Rome, official language, Italian*)
(*Elsa Morante, spouse, Alberto Moravia*)	**(*Aglaja Brix, place of birth, Hamburg*)**
(*Rome, official language, Italian*)	(*Rome, inv. airline hub, Norwegian Air Shuttle*)
(*natural causes, inv. opposite of, unnatural death*)	(*Elsa Morante, spouse, Alberto Moravia*)
(*Alberto Moravia, place of death, Rome*)	(*Argos 20 (Pauly-Wissowa), inv. increased expression in, Afghanistan*)
(*Rome, inv. airline hub, Norwegian Air Shuttle*)	(*natural causes, inv. opposite of, unnatural death*)
	(*Rome, inv. enclave within, Vatican City*)
	(*Alberto Moravia, place of death, Rome*)
	(*Elsa Morante, cause of death, myocardial infarction*)

$\mathbf{K_2^+}$	$\mathbf{K_3^+}$
(*Aglaja Brix, place of birth, Hamburg*)	(*KMT2B, language regulatory body, Battle of Otepaa*)
(*natural causes, inv. opposite of, unnatural death*)	(*Yamaguchi Prefecture, theme music, Expression of KLF15*)
(*Rome, inv. airline hub, Norwegian Air Shuttle*)	(*Alberto Moravia, place of death, Rome*)
(*Argos 20 (Pauly-Wissowa), inv. increased expression in, Afghanistan*)	(*natural causes, inv. opposite of, unnatural death*)
(*Dan Kaminsky, manner of death, natural causes*)	(*Elsa Morante, place of birth, Rome*)
(*Elsa Morante, spouse, Alberto Moravia*)	**(*Aglaja Brix, place of birth, Hamburg*)**
(*Alberto Moravia, place of death, Rome*)	(*Rome, inv. airline hub, Norwegian Air Shuttle*)
(*Elsa Morante, cause of death, myocardial infarction*)	**(*Volker Koepp, place of birth, Szczecin*)**
(*Alberto Moravia, inv. founded by, Nuovi Argomenti*)	(*Friedrich Fulleborn, place of death, Hamburg*)
(*Elsa Morante, place of birth, Rome*)	(*Dan Kaminsky, manner of death, natural causes*)
(*KMT2B, language regulatory body, Battle of Otepaa*)	(*Rome, official language, Italian*)
(*Rome, inv. enclave within, Vatican City*)	(*Alberto Moravia, inv. founded by, Nuovi Argomenti*)
(*Rome, official language, Italian*)	(*Elsa Morante, manner of death, natural causes*)
(*Elsa Morante, manner of death, natural causes*)	(*Wolfgang Thonke, cause of death, stomach cancer*)
(*Friedrich Fulleborn, place of death, Hamburg*)	(*Elsa Morante, cause of death, myocardial infarction*)
(*Michael Attaleiates, place of birth, Antalya*)	(*Argos 20 (Pauly-Wissowa), inv. increased expression in, Afghanistan*)
	(*Michael Attaleiates, place of birth, Antalya*)
	(*Elsa Morante, spouse, Alberto Moravia*)
	(*Rome, inv. enclave within, Vatican City*)

TABLE 4.3 – Example of K_D, K_1^+, K_2^+ and K_3^+ with **Out-of-Scope triples (entity)**, Out-of-Scope triples (property) and *Noise triples*

Dialog Context. As explained in Section 4.2, in the KGCONV dataset, each question-answer pair in a dialog is associated with the corresponding KG triple e.g.,

T : (*Henri_Poincare*, *birth_place*, *Nancy*)
Q : *Where was Henri Poincaré born ?*
A : *Nancy*

We leverage this dual (T, Q :A) information to compare four ways of representing the context : $D_{QA_{nl}}$ the standard dialog context consisting of all the preceding natural language turns ; $D_{Q_{nl}}$ the dialog context consisting of all the preceding natural language questions ; D_{kl}, the factual context consisting of the sequence of facts grounding the preceding dialog turns ; and $D_{QA_{nl}+kl}$, a hybrid context consisting of the sequence of preceding dialog turns and their associated facts. Table 4.4 illustrates these four types of contexts.

e	Sitara Achakzai
T_e	person
Len_K	5
K_D	(*Afghanistan* , *lowest point* , *Amu Darya*) (*Sitara Achakzai* , *birthplace* , *Afghanistan*) (*Sitara Achakzai* , *field of work* , *feminism*) (*Sitara Achakzai* , *death manner* , *murder*) (*Afghanistan* , *capital* , *Kabul*)
$D_{QA_{nl}}$	**<q>**What was the field of work of Sitara Achakzai? **<a>** feminism **<q>** What was the cause of death of Achakzai? **<a>** homicide
$D_{Q_{nl}}$	**<q>** What was the field of work of Sitara Achakzai? **<q>** What was the cause of death of Achakzai?
D_{kl}	<t><sj>Sitara Achakzai<p>field of work <o>feminism <\t> <t><sj>Sitara Achakzai <p> death manner<o> murder <\t>
$D_{QA_{nl}+kl}$	<t><sj>Sitara Achakzai <p>field of work<o>feminism<\t> **<q>**What was the field of work of Sitara Achakzai? **<a>** feminism <t><sj>Sitara Achakzai <p> death manner<o> murder <\t> **<q>** What was the cause of death of Achakzai? **<a>** homicide
Ref	[TRIPLE] <t><sj>Sitara Achakzai <p> birthplace <o> Afghanistan infarction <\t> [QUESTION] Where was she born ?

TABLE 4.4 – Input elements and reference output for the generation of the third question of the dialog of Table 4.1

4.4 Experimental Setting

4.4.1 Model and Task

The generation model we use is T5-small which we fine-tune on the KGCONV dataset. We use T5-small with the implementation from Huggingface. We train on two GPUs (Nvidia GTX 1080 Ti, 11 GiB) with early stopping criteria on the validation loss. We use greedy decoding. The input to the model

is a concatenation of 5 elements

$$e, T_e, Len_{K_D}, K_D, D$$

where e is the root entity of the graph and T_e its semantic category (taken from WikiData). K_D is the set of RDF triples grounding D, Len_{K_D} is the number of triples in K_D and D is the dialog context. The output of the model is the concatenation of a triple f and a natural language question q, where, as illustrated on the last line of Table 4.4, f and q are prefixed with the special markers [TRIPLE] and [QUESTION] respectively. Note that as we have four context types, we finetune four different models.

4.4.2 Data

We use the 46,808 dialogs of KGConv training set, 7,477 dialogs of the validation set and 10,753 of the test set. We augment these sets with corresponding K_1^+, K_2^+ and K_3^+ Knowledge Graphs as described in Section 4.3 and obtain 161,142 dialogs for the train set, 25,701 for the validation set and 37,056 for the test set. To avoid too many repetitions in the training and validation sets, we keep half (randomly selected) of all possible contexts for each dialog. For the test samples, we keep all contexts. We have 666,711 training instances, 105,665 validation, and 323,826 test instances. We remove instances with length longer than maximum length allowed by the model, i.e. with more than 512 tokens. This represents for each of the context types less than 2% of the training samples.

4.5 Evaluation methodology and results

We evaluate the generation outputs along 4 main criteria :

- **Well-formedness :** Is the output of the form RDF Triple | Natural Language Question ?
- **Relevance :** Is the generated triple relevant i.e., is it a triple from K_D that has not been mentioned in the preceding context ?
- **Semantic adequacy :** Does the generated question match the generated triple ?
- **Coreference :** If the question contains a pronoun, does its gender match the gender of its referent and does it refer to the correct entity without any ambiguity ?

We also compare our model to a baseline encoder-decoder which directly predicts the question from the context rather than generating both triple and question. The baseline is also a T5-small, finetuned in the same way.

4.5.1 Well-formedness

The model's task is to generate a triple followed by a question. If the generated output does not contain a `[TRIPLE]` tag followed by a `[QUESTION]` tag, the output is considered as incorrect. As shown in Table 4.5, this happens very rarely (0.1% of the time or less depending on the context type). As illustrated by the examples in Figure 4.1, these are typically cases of degenerate outputs.

	Context			
	$D_{QA_{nl}}$	$D_{Q_{nl}}$	D_{kl}	$D_{QA_{nl}+kl}$
Well-formed	313,583	321,270	315,815	313,865
Ill-formed	126	87	129	3

TABLE 4.5 – **Evaluating Well-Formedness.** Number of "Triple | Question" Format issues on the test set for the different context types

Ref	[TRIPLE] <t> <sj> Typha latifolia <p> inv. depicts <o> Andromeda Chained to the Rocks <\t> [QUESTION] Typha latifolia is depicted in what painting?</s>
Gen	[TRIPLE] <t> <sj> Least Concern <p> contient plenty of information about what topic?</s>
Ref	[TRIPLE] <t> <sj> Rudolph Valentino <p> inv. producer <o> What Price Beauty? <\t> [QUESTION] Which film had been produced by Rudolph Valentino?</s>
Gen	[TRIPLE] <t> <sj> Rudolph Valentino <p> inv. producer <o> gag gaga Rudolph Valentino proved what film?</s>
Ref	[TRIPLE] <t> <sj> Aidanbb<p> taxon rank <o> subfamily <\t> [QUESTION] What's the taxonomic rank of Aidanbb?</s>
Gen	[TRIPLE] <t> <sj> Charles Darwin <p> native language <o> English language," echipesagensagent <\t>
Ref	[TRIPLE] <t> <sj> Ipomoea <p> parent taxon <o> Ipomoeeae <\t> [QUESTION] What is the classification of Ipomoea?</s>
Gen	[TRIPLE] <t> <sj> Ipomoea <p> parent taxon <o> Ipomoeeaarising <\t> What is the higher classification of Ipomoea?</s>

FIGURE 4.1 – Examples of degenerate outputs (Ref : reference question)

4.5.2 Relevance (Content Selection)

We compute various statistics (Table 4.9) to analyze the relevance of the generated triple and the ability of our model to select adequate triples. Examples of errors are given in Tables 4.6, 4.7, and 4.8.

Root entity	Messier 36
Theme	space object
Length of the graph	7
RDF graph	(*Giovan Battista Hodierna*, *place of death*, *Palma di Montechiaro*)
	(*Giovan Battista Hodierna*, *place of birth*, *Ragusa*)
	(*Messier 36*, *constellation*, *Auriga*)
	(*Messier 36*, *discoverer or inventor*, *Giovan Battista Hodierna*)
	(*Messier 36*, *part of*, *Milky Way*)
	(*Auriga*, *named after*, *auriga*)
	(*Milky Way*, *parent astronomical body*, *Sagittarius A**)
Context	<q> What was the name of the discoverer of Messier 36 ?
	<q> What constellation is it in ?
	<q> What astronomical object is it a part of ?
	<q> Where was Giovan Battista Hodierna born ?
Generated Triple	(*Messier 36* , *place of death* , *Palma di Montechiaro*)
Generated Question	Where did Messier 36 die ?

TABLE 4.6 – Example of generated triple not in KGConv

We first check whether the triples are well-formed, i.e. whether they consist of a subject, a property and an object. The ratio of ill-formed triples is very low (1% or less) indicating that the model has learned to produce syntactically well-formed RDF triples.

We then verify whether the generated triple is correct, i.e. whether it is either an exact match with the reference triple or a triple from the input RDF graph K_D which has not been already generated (recall that by construction, triples in K_D are neither out-of-domain nor out-of-scope). We find that $D_{Q_{nl}}$contexts yield more incorrect triples (11%) than the other context types (between 3% and 4% of wrong triples) and that for this context type, the ratio of repeated triples (Repetitions) is higher (7% against 1% for the other types of context). This highlights the fact that to avoid repetitions and irrelevant turns, generation should be conditioned on a dialog context which includes not only questions but also their answers.

Finally, we see that the ratio of generated out-of-scope and noisy triples is low for all models (0 to 2%) and for all numbers of added distractors triples [16]. This indicates that the model can correctly discriminate between relevant and, more or less similar, irrelevant triples.

4.5.3 Semantic Adequacy

By comparing the generated triple and the generated question, we can assess semantic adequacy i.e., how well the question matches the triple. Since, as we saw in the preceding section, the generated triples are generally relevant, if the generated questions match the generated triple, then they can be deemed relevant.

We compute GLEU score (Johnson et al., 2017) between each generated question and verbalizations of the triple it was conditioned on at inference time. The triple verbalizations is the set of all the questions associated with this triple in the KGCONV dataset. The results are shown in Table 4.10.

16. While we do not give the detailed breakdown here for brevity's sake, we found that K_1^+, K_2^+ or K_3^+ yield a similar number of mistakes.

Root entity	Anna-Karin Stromstedt
Theme	person
Length of the graph	6
RDF graph	(*Anna-Karin Stromstedt*, *country of citizenship*, *Sweden*) (*Anna-Karin Stromstedt*, *place of birth*, *Vansbro Municipality*) (*Sweden*, *capital*, *Stockholm*) (*Alfred Schnittke*, *place of birth*, *Engels*) (*Vansbro Municipality*, *located in the administrative territorial entity*, *Dalarna County*) (*Vansbro Municipality*, *country*, *Sweden*) (*Anna-Karin Stromstedt*, *sport*, *biathlon*) (*Cherno More motorway*, *inv. location of formation*, *peroxisome organization*) (*Baptist Kniess*, *place of birth*, *Grunstadt*)
Context	<q> In what city was Anna-Karin Strömstedt born ? <a> Vansbro Municipality <q> Which sport did she play ? <a> biathlon <q> Which political territory is Vansbro Municipality located in ? <a> Dalarna County <q> To which country does Strömstedt belong as its citizen ? <a> Sweden
Generated Triple	(*Anna-Karin Stromstedt* , *country of citizenship* , *Sweden*)
Generated Question	What country is she from ?

TABLE 4.7 – Example of repetition of triple already generated

We find that the average GLEU score is high (from 0.73 to 0.76) across the board. The high GLEU scores further indicate a good match between generated triples and questions suggesting that the generated questions are well-formed relevant questions overall. Examples of generated questions and references are given in Figure 4.2.

4.5.4 Coreference

Dialogs typically include anaphors and ellipses. As in our approach, the generated question is grounded in a fact, we can analyze both whether a pronoun gender matches its referent (the subject entity in the fact the question is conditioned on) and whether its antecedent is easy to identify or in other words, whether the pronoun is ambiguous [17]. We do this as follows.

— **Gender.** For each entity in our dataset, we retrieve its Wikidata 'sex or gender' object if any. We classify the retrieved genders into three main categories : feminine, masculine, and other genders (which contain different types of gender queer categories). For the 'other genders' category, we accept all pronouns as correct pronouns. If an entity does not have any 'sex or gender' property we consider it as a neutral entity. To assess whether a pronoun in a generated question has the correct gender, we check whether this gender (either feminine, masculine, neutral, or other gender) is the same as the gender of its referent, i.e. the subject entity of the triple the question is conditioned on.

— **Ambiguity** We say that a pronoun with gender g is ambiguous if the last entity of gender g mentioned in the dialog context is not the referent of that pronoun. Figure 4.3 shows an example.

17. We limit our analysis to third-person pronouns and ignore first and second-person pronouns, which are usually rhetorical.

Generated Question	Geneva peace talks on Syria (2017) was organized by which organization ?
Generated Triple	(*Geneva peace talks on Syria (2017)*, *organizer*, *United Nations*)
References	['What was the name of the organization that organizes Geneva peace talks on Syria ?', 'Who organized these talks ?', 'Who organized this event ?', 'What organization was responsible for organizing Geneva peace talks on Syria (2017) ?', 'Which organization was the organizer of Geneva peace talks on Syria ?', 'What organization was responsible for organizing ?', 'What was the name of the organization that organized Geneva peace talks on Syria ?', 'What was the name of the organization that organized Geneva peace talks on Syria (2017) ?', 'What organization organizes these talks ?', 'Geneva peace talks on Syria was organized by what organization ?', 'Which organization was the organizer ?', 'What organization was responsible for organizing Geneva peace talks on Syria ?', 'It was organized by what organization ?', 'What organization organized this event ?', 'What was the name of the organization that organizes Geneva peace talks on Syria (2017) ?', 'Which organization was the organizer of Geneva peace talks on Syria (2017) ?', 'Geneva peace talks on Syria (2017) was organized by what organization ?']
GLEU	0.82
Generated Question	Who is Iceland led by ?
Generated Triple	(*Iceland*, *head of government*, *Katrin Jakobsdottir*)
References	['Who is the head of government in it ?', 'Who is the head of government there ?', 'Who is the head of government in Iceland ?', 'Who is it head of government ?', 'Iceland is led by who ?', 'Who leads this country ?', 'Who is it led by ?', 'Who is Iceland led by ?', "Who is it's head of government ?", "Who is Iceland's head of government ?", 'Who is the head of state of this country ?', 'Who is the head of government ?', 'Who is the director of this institution ?', 'Who is this country led by ?', "What is the name of Iceland's head of state ?", 'Who is it led by ?']
GLEU	1
Generated Question	Milky Way is a part of what family of people ?
Generated Triple	(*Milky Way* , *parent astronomical body*, *Sagittarius A**)
References	['What is the parent astronomical body of the Milky Way ?', 'What is the parent celestial body of The Milky Way Galaxy ?', 'What is the parent astronomical body of This Milky Way ?', 'What is the parent celestial body of the Milky Way galaxy ?', 'What is its parent astronomical body ?', 'What is the parent astronomical body of the Milky Way galaxy ?', 'What is the parent celestial body of The Milky Way ?', 'What is the parent celestial body ?', 'What is the parent astronomical body of the Milky Way Galaxy ?', 'What is the parent astronomical body ?', 'What is the parent celestial body of the Milky Way Galaxy ?', 'What is the parent astronomical body of The Milky Way Galaxy ?', 'What is her parent astronomical body ?', 'What is her parent celestial body ?', 'What is the parent astronomical body of The Milky Way ?', 'What is the parent celestial body of the Milky Way ?', 'What is the parent celestial body of This Milky Way Galaxy ?', 'What is the parent celestial body of this galaxy ?', 'What is this body ?', 'What is it ?', 'What is its parent celestial body ?', "What is it's parent astronomical body ?", 'What is the parent astronomical body of this galaxy ?']
GLEU	0.16

FIGURE 4.2 – Examples of references for GLEU score computation for Question Evaluation

Root entity	Giovanni Battista Ferrandini
Theme	person
Length of the graph	5
RDF graph	(*Giovanni Battista Ferrandini*, *place of birth*, *Venice*)
	(Cytoplasmic Ribosomal Proteins, sponsor, Menzlin)
	(March Across the Belts, part of, Dano-Swedish War)
	(*Venice*, *inv. producer*, *The Devil's Gondola*)
	(Nia Long, place of birth, Kingston)
	(*Giovanni Battista Ferrandini*, *genre*, *opera*)
	(*Venice*, *located in or next to body of water*, *Venetian Lagoon*)
	(*baroque music*, *part of*, *Baroque*)
	(*baroque music*, *followed by*, *Classical period*)
	(Nia Long, country of citizenship, United States of America)
	(Madurai, inv. producer, Manonmani)
	(*Giovanni Battista Ferrandini*, *country of citizenship*, *Republic of Venice*)
	(Karim El Ahmadi, place of birth, Enschede)
	(*Munich*, *inv. producer*, *Ilsa, the Wicked Warden*)
	(Salva Kiir Mayardit, licensed to broadcast to, Lake Michigan)
	(Kuksu, made from material, Theodor Zwinger)
	(*Giovanni Battista Ferrandini*, *place of death*, *Munich*)
	(*Giovanni Battista Ferrandini*, *movement*, *baroque music*)
	(Rita Hayworth, inv. producer, Affair in Trinidad)
Context	(*Giovanni Battista Ferrandini*, *genre*, *opera*)
	(*Giovanni Battista Ferrandini*, *country of citizenship*, *Republic of Venice*)
	(*Giovanni Battista Ferrandini*, *movement*, *baroque music*)
	(*Giovanni Battista Ferrandini*, *place of death*, *Munich*)
	(*Giovanni Battista Ferrandini*, *place of birth*, *Venice*)
Generated Triple	(*Nia Long*, *country of citizenship*, *United States of America*)
Generated Question	What was the nationality of Long ?

TABLE 4.8 – Example of distractor triple generated

While the referent of the masculine pronoun occurring in the generated question is "William Hershel", the last entity of masculine gender mentioned in the dialog is "Nevil Maskelyne". We approximate the number of ambiguous pronouns using the following two heuristics. First, we consider that any pronoun occurring in a null dialog context is ambiguous (since it cannot corefer with a previously mentioned entity). Second, we count as ambiguous any pronoun in a generated question whose referent is different from the entity of the same gender last mentioned in the dialog.

The results of the pronoun evaluation are given in Table 4.11. We observe a good ratio of questions containing pronouns (between 8 and 13% of the test examples) and a good diversity of the triples giving rise to such questions (about 25% of the dataset triples lead to the generation of a question containing a pronoun).

Interestingly, while contexts that contain natural language questions have a similar ratio of questions with pronouns, the D_{kl} context, which only consists of triples, induces a much higher rate of pronouns.

Focusing on gender first, we observe a strong bias for masculine pronouns, which is in line with e.g., Fan and Gardent (2022)'s analysis of Wikipedia articles. We further find that the proportion of pronouns

Context Type	$D_{QA_{nl}}$	(%)	$D_{Q_{nl}}$	(%)	D_{kl}	(%)	$D_{QA_{nl}+kl}$	(%)
# test examples	313583		321270		315815		313865	
# distinct generated triples	16519		18146		17875		16597	
Correct triples	303723	97	286439	89	301970	96	304794	97
Exact match with target	123684	39	109031	34	123605	39	131453	42
Other triple from input RDF	180039	57	177408	55	178365	56	173341	55
Incorrect triples	9860	3	34831	11	13845	4	9071	3
Repetitions	1788	1	23149	7	1308	0	1705	1
Out-of-scope (entity) triples	305	0	640	0	340	0	398	0
Out-of-scope (property) triples	5327	2	6987	2	6437	2	5448	2
Noise triples	0	0	0	0	0	0	0	0
Ill-formed triples	460	0	2033	1	1663	1	710	0
Triples not in KGCONV	5514	2	7403	2	7761	2	4977	2

TABLE 4.9 – **Relevance.** Results of the triples evaluation. The different number of test examples between context types come from different numbers of inputs of lengths greater than 512 tokens (as explained in section 4.4.2).

Context	$D_{QA_{nl}}$	$D_{Q_{nl}}$	D_{kl}	$D_{QA_{nl}+kl}$
Avg GLEU	0.76	0.78	0.73	0.76

TABLE 4.10 – Mean of the GLEU scores between a question and the triple it was conditioned on.

Dialog Context	T : (*NGC 2539*, *discoverer or inventor*, *William Herschel*) Q : `Who found NGC 2423?` A : `William Herschel` T : (*NGC 2539*, *constellation*, *Puppis*) Q : `What is the name of the constellation which NGC 2423 belongs?` A : `Puppis` T : (*William Herschel*, *student of*, *Nevil Maskelyne*)) Q : `What was the name of Herschel's teacher?` A : `Nevil Maskelyne`
Last feminine entity	-
Last masculine entity	**`Nevil Maskelyne`**
Last neutral entity	`Puppis`
Generated Question	`where was` **`he`** `buried?`
Generated Triple	(*William Herschel*, *place of burial*, *Westminster Abbey*)
Pronoun	`he`
Pronoun Antecedent	**`William Herschel`**
Gender of the pronoun's antecedent	masculine

FIGURE 4.3 – **Example of Gender Ambiguous Pronoun :** The pronoun denotes a male entity (William Herschel) which is different from the last mentioned male entity (Nevil Maskelyne).

Context Type	$D_{QA_{nl}}$	$D_{Q_{nl}}$	D_{kl}	$D_{QA_{nl}+kl}$
questions with a pronoun	9%	8%	13%	8%
"he"	53%	47%	54%	52%
"it"	32%	35%	34%	35%
"him"	7%	10%	8%	7%
"she"	8%	7%	3%	6%
"her"	<1%	1%	4%	<1%
pronouns with gender mistakes	5%	5%	3%	4%
"he"	29%	44%	68%	52 %
"she"	62 %	39%	18%	34%
"him"	4 %	9 %	9 %	8%
"her"	3%	5 %	2 %	2%
"it"	2%	3%	3 %	4 %
ambiguous pronouns	30%	36%	34%	29%
"it"	64%	67%	76%	66%
"he"	18%	19%	15%	21%
"she"	14%	9%	4%	9%
"him"	3%	4%	4%	3%
"her"	1%	1%	1%	1%
pronominalized distinct triples	22%	19%	24%	19%

TABLE 4.11 – Results of the pronouns evaluation

with incorrect gender is reasonably low ranging from 3% to 4%.

In contrast, the proportion of ambiguous pronouns is quite high, ranging between 29% to 36%, suggesting that referring expression generation in dialog is not yet a solved problem. As our counts are based on heuristics however, a human evaluation would be needed to verify that cases that are deemed ambiguous by these heuristics are not in fact non-ambiguous, given some common sense knowledge. For instance, in the sentence "I hit the window with a stone and it broke", we understand the pronoun "it" as referring to "the window" even though "the window" is not the closest entity of neutral gender.

4.5.5 Baseline

To compare the results of generating both a triple and a question, with the results when generating only a question, we create a baseline model with the same input as the model described in section 4.4, but whose output is now only the next question in natural language. For this baseline, we use the context type $D_{QA_{nl}+kl}$. Note that as the point of our approach is not performance but explainability, our goal is not necessarily to outperform the baseline but rather to check that our approach does not yield any important performance drop.

Automatic Evaluation We then compare the two models by using the GLEU score. We compute the GLEU score between the generated question and a set of references containing the verbalizations of a correct triple (ie a triple in the input RDF graph but not in the dialog context). We do this for all possible

Carefully read the dialog and possible next questions A and B below, then answer the questions Q1.A, Q1.B, Q2.A, Q2.B, Q3.A and Q3.B.

- *For Q1 (Q1.A and Q1.B), questions A and B don't need to be perfectly grammatical/fluent, but to the standard of an average English speaker.*
- *For Q2 (Q2.A and Q2.B), repetitions are questions that have already been asked in the dialog or questions to which the answer has already been given in the dialog.*
- *For Q3 (Q3.A and Q3.B), non-coherent questions include questions that are odd or non-logical to ask given the dialog context.*

Questions :
- *Q1.A. Is Question A written in grammatical/fluent/well-formed English ? (Yes/ No)*
- *Q1.B. Is Question B written in grammatical/fluent/well-formed English ? (Yes/ No)*
- *Q2.A. Is Question A a repetition of some information already given in the dialog ? (Yes/ No)*
- *Q2.B. Is Question B a repetition of some information already given in the dialog ? (Yes/ No)*
- *Q3.A. Is Question A coherent with the dialog ? (Yes/ To some extent /No)*
- *Q3.B. Is Question B coherent with the dialog ? (Yes/ To some extent /No)*

FIGURE 4.4 – Annotations instructions

correct triples and then keep the maximum of the GLEU scores computed. We compute the GLEU in the same way for our model generating a triple together with a question and using the $D_{QA_{nl}+kl}$ context.

On the test set (313,474 instances), the mean GLEU score of the baseline is 0,50 against a mean GLEU score of 0.52 for our model. We use the t-test to compute statistical significance. A p-value smaller than 0.01 indicates that the GLEU scores of the baseline are significantly lower than the GLEU scores of our model.

Human Evaluation We perform a human evaluation with 5 annotators. The annotations are crowdsourced on the Amazon Mechanical Turk crowdsourcing platform. Via the AMT platform, we showed with each dialog, a question generated by our model and a question generated by the baseline model (randomly labeled as questions A and B for the different dialogs). We gave the annotators instructions, shown in Figure 4.4. We paid 0.18$ for dialogs of lengths 1 or 2, 0.27$ for dialogs of lengths 3 and 4 and 0.63$ for dialogs of length 5. The total cost of the 1500 annotations (300 dialogs by five annotators) was 448$.

We randomly select 300 dialog contexts, among the person, historical event, food, ideology or country categories, as they are the easiest categories to assess without specific background knowledge. We select 60 dialog contexts of each length from one to five in order to have some diversity in dialog lengths but also keep the annotation task reasonably simple. We ask the 5 annotators [18] to evaluate the question generated by the baseline and the question generated by the model along three criteria : (1) the fluency of the questions, (2) whether the generated question is a repetition of information from the dialog context, and (3) the coherence of each of the generated questions with the dialog context. Fluency and repetition are evaluated on a binary scale (yes, no), coherence on a ternary scale (high, medium, low). We then use a majority vote among annotators to determine the final annotation of each question and compute the ratio of cases for which each model is assessed to be fluent, non-repetitive and coherent.

18. We recruited 5 annotators who have the Amazon Mechanical Turk "Master Qualification", i.e. who consistently submitted high-quality results in the past.

We evaluate the inter-annotator agreement, we compute Cohen's kappa and the observed agreement (i.e. the proportion of examples for which annotators agree) between each pair of annotators. We report the mean in Table 4.12. The relatively low Cohen's kappa and high observed agreement is due to the fact that our data is skewed as both the model and the baseline perform on average very well regarding the 3 evaluation criteria.

	Avg Cohen's κ	Avg Obs. agreement
fluency (B)	0,18	0,93
fluency (M)	0,19	0,91
repetition (B)	0,31	0,87
repetition (M)	0,26	0,84
coherence (B)	0,15	0,67
coherence (M)	0,16	0,65

TABLE 4.12 – Evaluation of the inter-annotator agreement for the human evaluation (B :baseline, M :model)

The human evaluation shows similar results for the baseline and for our model with 99% of the generated questions judged fluent by the annotators for the baseline vs. 98% for our model ; 94% of the baseline output judged non-repetitive for the baseline vs. 93% for our model ; and 83% of the generated questions considered coherent with the dialog context for our model against 85% for the baseline. In these examples, generating a triple and a question does not seem to particularly improve the quality of the generation but it also does not decrease the quality and provides additional information for fine-grained automatic evaluation.

4.6 Ablation study

We use ablation to assess the respective impact of the input knowledge graph and the dialog context on dialog coherence. We train and test the same model as in Section 4.4 with the exact same data, ablating either the input knowledge graph or the dialog context from the input.

Ablating the Input Knowledge Graph. We hypothesize that by conditioning question generation not only on the dialog context but also on a knowledge graph K helps learn a dialog model which produces coherent sequences of questions. To quantitatively assess the impact of this added input on dialog coherence, we ablate K_D and examine the triples generated by the ablated model. The results are shown in Table 4.13. Depending on the context type between 91% and 92% of generated triples are incorrect. Almost all of them (between 81 % and 84% of the generated triples) are hallucinated triples not belonging to the set of KGCONV triples, a large set of 132K Wikidata triples.We further investigate for the $D_{QA_{nl}+kl}$ context what these hallucinated triples contain. In most of the hallucination cases (81%), even though the triple is not in $\mathcal{K}_{KGConv}$, the subject, the property and the object are in $\mathcal{K}_{KGConv}$. This means that entities and properties from $\mathcal{K}_{KGConv}$ are rearranged into non existing Wikidata triples (examples are given in Figure 4.5).

Ablating the Context D. Unsurprisingly, ablating the dialog context (Table 4.14) drastically reduces the proportion of correct triples (51%) and increases the ratio of repetitions (46%).

With subject, property and object in $\mathcal{K}_{KGConv}$
(Milky Way, located in the administrative territorial entity, New York City)
(Nicolas Louis de Lacaille, place of birth, Paris)
With object not in $\mathcal{K}_{KGConv}$
(Armand David, inv. founded by, Yvonnes-Altas)
(LC3 :PE [autophagosome membrane], encoded by, CL3PE)

FIGURE 4.5 – Examples of triples generated when ablating RDF graph K

Context Type	$D_{QA_{nl}}$	$D_{Q_{nl}}$	D_{kl}	$D_{QA_{nl}+kl}$
# Test examples	323k	302k	323k	323k
Incorrect triple	92%	92%	91%	91%
Repetition	2%	1%	2%	1%
Triple not in KGCONV	84%	81%	83%	82%
Subject not in KGCONV	13%	28%	17%	15%
Property not in KGCONV	14%	33%	17%	16%
Object not in KGCONV	13%	29%	17%	15%

TABLE 4.13 – Ablation of RDF graph K, Results of triple evaluation

	#	%
# test examples	323765	
Correct triples	166716	51
Exact match with target	36474	11
Other triple from input RDF	130242	40
Incorrect triples	157049	49
Repetitions	149363	46
Out-of-scope (entity) triples	327	0
Out-of-scope (property) triples	8713	3
Noise triples generated	0	0
Ill-formed triples	182	0
Triples with a property not in KGCONV	6989	2

TABLE 4.14 – Ablation of context D, Results of triple evaluation

4.7 Conclusion and Future work

In this chapter, we study the task of question generation in knowledge-based dialogs comparing different ways of representing dialog context and exploring the impact of the knowledge graph generation is conditioned on.

Instead of generating only questions, we generate both a triple and the corresponding question and we show that this permits assessing dialog coherence and coreference-based dialog cohesion.

In addition, we find (i) that conditioning generation on both questions and answers or on the corresponding triples is crucial to maintain coherence and (ii) that coreferences are generally better handled when the context includes KB facts. Our ablation study further shows that conditioning question generation on both dialog context and a knowledge graph drastically improves coherence.

In future work, we plan to investigate how the approach generalizes to knowledge-based dialog datasets such as CSQA Liu (2021) which includes more complex questions; and to explore ways of mitigating pronoun ambiguity in knowledge-based, conversational question generation.

Conclusion

In this thesis, we studied content-related issues in state-of-the-art Data-to-Text NLG models. We considered two tasks, RDF verbalization, and conversational question generation. We focused on content-related aspects that are crucial for these two tasks, i.e. semantic adequacy between the output verbalization and the input graph for RDF verbalization and coherence for conversational question generation.

Our study is related to two fields : Evaluation of NLG and Explainability. For the task of RDF verbalization, we studied both aspects separately. First, we proposed an evaluation metric of semantic adequacy, and then we used probing explainability techniques to analyze semantic inadequacies in NLG models' embeddings. This type of explanation can be categorized as a post-hoc XAI and local explanation (the explanation for one specific input). The explanation is also relatively abstract and model-grounded. For the task of conversational question generation, we proposed to enhance the controllability and facilitate a fine-grained automatic evaluation by generating a semantic representation together with the target question. It can be seen as a step going in the direction of self-explaining models. Increasing explainability usually comes at the cost of performance. This is however not the case with this approach, as the performance of the model is not hindered by the generation of the content plan. This also provides a form of explanation that is less abstract than in Chapter 3 and more human-grounded.

Summaries of our contributions and answers to our research questions

Evaluation of RDF Verbalizers In Chapter 2, we evaluated the semantic adequacy of RDF verbalization models based on the verbalization of RDF entities. We first built an entity mention detection tool to automatically detect input RDF entities in an output text. We used a combination of heuristics and existing tools such as the state-of-the-art REL entity linker, including both surface-based and embedding-based mentions detection methods. The performance of this tool was assessed on gold standard manually annotated data by Castro Ferreira et al. (2018) and has a precision of 0.75 and a recall of 0.74.

We applied our automatic entity mentions detection on the outputs of 25 models that participated in the WebNLG Challenges 2017 and 2020. For each model, we could compute the proportion of texts for which some entities were undetected. On average for the most recent models, we detected that 17% of texts had at least one missing input entity. Thanks to a manual check on about 500 texts, we could see that if our entity detection sometimes fails at detecting entities in texts, and therefore reports too many undetected entities, it performs reasonably well for models which have more missing entities and can identify serious errors in entity verbalization. At the instance-level, we computed a semantic adequacy metric ESA_Iwhich is the proportion of RDF input entities we automatically detect in a corresponding text.

We studied the correlation of this metric with automatic and human evaluation metrics used in the WebNLG Challenges 2017 and 2020. We saw that ESA_Icorrelates strongly with the Semantics human evaluation criterion for the WebNLG Challenge 2017 and has a medium-strength correlation with the semantic-related ratings from the human evaluation of WebNLG Challenge 2020.

We finally extended the study to the analysis of hallucinations and found that a significant number of

texts contained hallucinated entities, particularly neural models.

Our first research question was :

RQ1 : To what extent do state-of-the-art RDF-to-Text models have content-related issues, in particular omissions and hallucinations ?

Our experiments in Chapter 2 showed that all recent RDF-to-Text models of the WebNLG Shared Task such as BART or T5-based models are subject to omissions and hallucinations of RDF entities. Using both a fine-grained automatic evaluation metric and a manual evaluation of the texts generated by the WebNLG Challenge models we could assess generated content-related mistakes both quantitatively and qualitatively.

Explainability of RDF Verbalizers In Chapter 3, we presented a methodology to detect omissions and distortions in the embeddings of transformer-based encoder-decoders. We first fine-tuned a pretrained BART model on the WebNLG dataset. We then used this model to generate verbalizations of WebNLG and KELM RDF graphs.

Using the annotation tool we developed in Chapter 2, we automatically annotated omissions in the generated texts. We also recruited three annotators to manually annotate omissions and distortions in a subset of the automatically annotated data.

We then investigated whether omissions and distortions of input RDF entities could be detected in their encodings, i.e. in the output embedding of the encoder of BART. We proposed a novel non-parametric probing classifier based on cosine similarities between embeddings and a neural probing classifier. The non-parametric probing classifier achieved an F-measure of 0.68 and the neural classifier an F-measure of 0.82. Results differ for distortions and omissions suggesting that distorted and omitted entities are not encoded similarly. We also showed our results extend to omissions in a T5 model, which supports our claim that our probing methodology is model agnostic.

RQ2 : Can we detect content-related issues, namely omissions, and distortions in the embeddings of encoder-decoder models ? In particular, can we detect these issues in the encodings of RDF input graphs ?

In Chapter 3, we carried out in-depth probing experiments using both non-parametric and parametric probes on the encodings of RDF graphs by BART and T5. We found that indeed our probes could detect omissions and distortions with relatively high F-measures. This shows that the encoders encode mentions and omissions or distortions differently, which suggests that the encoders are at least partly responsible for the omissions and distortions of RDF input entities.

Conversational Question Generation In Chapter 4, we presented an adaptation of a conversational question generation model that generates a content plan, in the form of an RDF triple, along with the target question. We use a T5 pretrained model which we fine-tune on the task of conversational question generation of the KGConv dataset. Generating a triple and the corresponding question enables a two-step evaluation process. First, we evaluate the generated triple by comparing it to the RDF input triples and the conversation context, checking for semantic adequacy with the input and coherence with the context. Second, we evaluate the question by making sure it matches the generated triple. Generating a triple together with a question does not hinder the NLG model as its performance compared to a base model generating only the questions does not decrease. However, it allows for a simplified fine-grained automatic evaluation.

RQ3 : How can we adapt state-of-the-art NLG models to facilitate the identification and quantification of content-related issues ? One solution to make content-related issues easier to evaluate and increase the model's controllability is to keep future evaluation in mind while designing the models. Our

conversational question generation model from Chapter 4 provides an example of this. By making a T5-based model generate both a question and the RDF triple corresponding to this question, we can design an evaluation that separates content selection from surface realization evaluation making a fine-grained automatic evaluation possible.

Future Directions

The experiments we carried out in this thesis have several limitations. To overcome them, there are different research directions that could be taken. We detail these directions in the following paragraphs.

Multilinguality In this thesis, we only worked with English NLG models. Seeing if our results extend or differ when generating in one or multiple other languages, in particular low-resourced ones, would be a really interesting extension, that would provide better insight into how NLG models work.

Extend the explainability study In Chapter 3, we study in depth a particular explainability technique that we applied to a specific part of the model. However, they are many other aspects of the model that we could have studied.

By analyzing other parts of the model In Chapter 3, we probed the output of the encoder of BART and T5. However, we could also, using the same experiment setting, probe the other layers of the encoder and the decoder and check if the probing performance drops progressively across the encoder and decoder layers. Probing the input of the encoder would also have been an interesting experiment as it would potentially have given a way to analyze how well transforming input tokens into embedding vectors preserves specific information, i.e. in our case RDF entities information.

By using other explainability techniques We used probing classifiers in Chapter 3, however, there are many other techniques such as the ones mentioned in Chapter 1 that could have been used. For instance, using attributions-based explainability methods such as Integrated Gradients or other Shapley values approximations to compare the relative contributions of input tokens corresponding respectively to a mention, an omission, or a distortion in the model's embeddings would be an interesting complementary study.

Most explainability methods, including probing, have the important limitation that they rely on correlations and not causality. Methods using interventions on the embeddings, i.e. in our case finding a transformation of the embeddings that impacts the omissions or distortions of RDF input entities in the output text would have been a way to analyze models using causality. Note that in our case a critical bottleneck is the data annotation. In the absence of an accurate enough automatic tool to detect omissions and distortions, we must rely on human annotations. These are expensive and time-consuming and become impractical if many iterations of annotations must be done until the correct embedding intervention transformation is found.

By studying the impact of the training data The performance of state-of-the-art NLG models is largely influenced by the quality of their training data. Therefore, looking for explanations of the models' behavior - including their tendency to generate texts that contain omissions and distortions - in training data patterns is a very promising approach. In order to draw links between a model's output and the training data used, one could look at either or both the finetuning and pretraining data. There are many factors that can be studied, such as the impact of the number of finetuning steps, or ratio and differences

between pretraining and finetuning data. Another possible path would be to look at how different biases and spurious correlations in the training data impact the mistakes a model makes.

Evaluation of the explanations Lacking in our study is an evaluation of the explanations. As we introduced in Chapter 1, the evaluation can be done along different perspectives. Application-grounded and human-grounded evaluation would have required the use of our NLG models for a particular application or asking users to rate the output of the probing method. There is also no standard method for model-grounded evaluation. Jacovi and Goldberg (2020) review the trends for the evaluation of faithfulness of explainability methods in NLP models and advocate for metrics that define a degree of faithfulness (instead of a binary "faithful vs not faithful" rating) and detail for which subspaces of the input space (similar input examples) the explanations are valid.

Choice of the models and models hyperparameters As we mentioned in Chapter 1, there is no consensus on the best architecture for NLG. As a consequence studying how different pretrained model's architecture but also model sizes impact the generation quality could be beneficial for understanding how the mistakes they produce arise.

Other content-related issues In this thesis, we focused on relatively simple content-related issues. For the task of RDF verbalization, we considered omissions, distortions, or hallucinations of input entities. We did not look at the predicates of the RDF input graphs. The possible verbalizations of the predicates are likely to be more varied than the entities verbalization making their evaluation and detection in the generated texts more challenging. However, they account for one-third of the input graph and should be evaluated as well to get a more holistic view of the models' performance. We also did not look into the other content-related issues apart from which elements are mentioned in the output text, in particular, we did not consider logical links between the entities or more discourse-level coherence aspects. In other words, having all RDF input entities and properties mentioned in the output texts is not a sufficient condition for having a well-formed text that correctly conveys the meaning of the RDF input. For the conversational question generation task, we evaluated adequacy with the input RDF graph as well as discourse coherence, based on the absence of repetition with the context information. However more elaborate criteria could have been used, e.g. imposing constraints on the order in which the RDF graph should be verbalized to forbid, for instance, jumping from one topic to another not directly related.

For both the RDF verbalization and conversational question generation tasks we used input RDF graphs of relatively small sizes yielding relatively short texts or conversations. Studying longer graphs and outputs would involve evaluating and analyzing more complex discourse elements, e.g. more sophisticated referring expressions or sentence aggregations. As we already mentioned in Chapter 1 we did not include any world knowledge and limited ourselves to the information explicitly present in the input RDF graphs. Including world knowledge and common sense reasoning would significantly increase the complexity of the evaluation but would also make the task more realistic.

Other NLG tasks In this thesis, we considered only two NLG tasks within Data-to-Text applications, RDF verbalization, and conversational question generation. Extending our study to other NLG tasks in particular tasks including some content selection, e.g. summarization or image captioning would mean rethinking content-related issues adequate for these tasks. For instance, omissions are harder to define when including content selection.

Content-related issues interactions with other mistakes We considered content-related issues independently from other aspects of the text. It would be interesting to see if some links can be found between

the tendency of the models to generate texts with wrong content and other aspects such as fluency or diversity of the outputs.

In this thesis, we studied content-related issues in state-of-the-art RDF-to-Text NLG models for RDF verbalization and conversational question generation. We found that whereas these models still make semantic mistakes they are multiple techniques which can be used to detect, analyse and mitigate them, with the aim of developping more reliable models.

A

Entity-Based Semantic Adequacy - Appendix

A.1 Examples of Mentions Detection for Different Models' Outputs

Model	RALI
Text ID	('Id714', 'Id1')
Text generated by the model	Bionico country Mexico. Bionico dish variation Cottage cheese
Example of reference text	Bionico, a food found in Mexico, can be varied by using cottage cheese.
RDF Input	Bionico | country | Mexico Bionico | dishVariation | Cottage_cheese
Entitymap	{'AGENT-1' : 'Bionico', 'PATIENT-1' : 'Mexico', 'PATIENT-2' : 'Cottage_cheese'}
Detected mentions	{'PATIENT-2' : [{'mention' : 'cottage cheese.', 'beginIndex' : 45, 'endIndex' : 60, 'edit_distance' : 0.07142857142857142, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'Bionico', 'beginIndex' : 0, 'endIndex' : 7, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-1' : [{'mention' : 'Mexico', 'beginIndex' : 16, 'endIndex' : 22, 'type' : 'LOC', 'method' : 'REL'}]}
Detected entities	['PATIENT-2', 'AGENT-1', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	3
Number of entities in the RDF input	3

Model	RALI
Text ID	('Id1679', 'Id1')
Text generated by the model	University of Burgundy number of undergraduate students 16800. University of Burgundy number of postgraduate students 9400

Example of reference text	There are 16800 undergraduate with 9400 post-graduate students at the University of Burgundy.
RDF Input	University_of_Burgundy | numberOfUndergraduateStudents | 16800 University_of_Burgundy | numberOfPostgraduateStudents | 9400
Entitymap	{'AGENT-1' : 'University_of_Burgundy', 'PATIENT-1' : '16800', 'PATIENT-2' : '9400'}
Detected mentions	{'AGENT-1' : [{'mention' : 'University of Burgundy', 'beginIndex' : 0, 'endIndex' : 22, 'type' : 'ORG', 'method' : 'REL'}], 'PATIENT-1' : [{'mention' : '16,800', 'beginIndex' : 27, 'endIndex' : 33, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : '9,400', 'beginIndex' : 65, 'endIndex' : 70, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-1', 'PATIENT-2']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	3
Number of entities in the RDF input	3

Model	RALI
Text ID	('Id432', 'Id1')
Text generated by the model	Darinka Dentcheva citizenship United States. United States leader John Roberts
Example of reference text	Darinka Dentcheva is a citizen of the United States, the country where John Roberts is one of the leaders.
RDF Input	Darinka_Dentcheva | citizenship | United_States United_States | leader | John_Roberts
Entitymap	{'AGENT-1' : 'Darinka_Dentcheva', 'BRIDGE-1' : 'United_States', 'PATIENT-1' : 'John_Roberts'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Darinka Dentcheva', 'beginIndex' : 0, 'endIndex' : 17, 'type' : 'PER', 'method' : 'REL'}], 'BRIDGE-1' : [{'mention' : 'United States', 'beginIndex' : 33, 'endIndex' : 46, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-1' : [{'mention' : 'John Roberts.', 'beginIndex' : 54, 'endIndex' : 67, 'edit_distance' : 0.08333333333333333, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'BRIDGE-1', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	3
Number of entities in the RDF input	3

Model	RALI
Text ID	('Id615', 'Id1')

Text generated by the model	11th Mississippi Infantry Monument established 2000. 11th Mississippi Infantry Monument location Adams County, Pennsylvania. Adams County, Pennsylvania has to its southwest Frederick County, Maryland. Adams County, Pennsylvania has to its southeast Carroll County, Maryland. 11th Mississippi Infantry Monument category Contributing property. Adams County, Pennsylvania has to its north Cumberland County, Pennsylvania. 11th Mississippi Infantry Monument country United States
Example of reference text	In 2000, the 11th Mississippi Infantry Monument, a contributing property was established in Adams County, Pennsylvania (United States). To its southwest is Frederick County, Maryland, with Carroll County, Maryland to the southeast, and Cumberland County, Pennsylvania to the north.
RDF Input	11th_Mississippi_Infantry_Monument | established | 2000 11th_Mississippi_Infantry_Monument | location | Adams_County,_Pennsylvania Adams_County,_Pennsylvania | hasToItsSouthwest | Frederick_County,_Maryland Adams_County,_Pennsylvania | hasToItsSoutheast | Carroll_County,_Maryland 11th_Mississippi_Infantry_Monument | category | Contributing_property Adams_County,_Pennsylvania | hasToItsNorth | Cumberland_County,_Pennsylvania 11th_Mississippi_Infantry_Monument | country | "United States"
Entitymap	{'AGENT-1' : '11th_Mississippi_Infantry_Monument', 'BRIDGE-1' : 'Adams_County,_Pennsylvania', 'PATIENT-1' : '2000', 'PATIENT-2' : 'Frederick_County,_Maryland', 'PATIENT-3' : 'Carroll_County,_Maryland', 'PATIENT-4' : 'Contributing_property', 'PATIENT-5' : 'Cumberland_County,_Pennsylvania', 'PATIENT-6' : '"United States"'}

Detected mentions	{'AGENT-1' : [{'mention' : '11th Mississippi Infantry Monument', 'beginIndex' : 0, 'endIndex' : 34, 'edit_distance' : 0.08823529411764706, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'its', 'beginIndex' : 243, 'endIndex' : 246, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'its', 'beginIndex' : 293, 'endIndex' : 296, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'its', 'beginIndex' : 350, 'endIndex' : 353, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'it', 'beginIndex' : 76, 'endIndex' : 78, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'it', 'beginIndex' : 107, 'endIndex' : 109, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'It', 'beginIndex' : 150, 'endIndex' : 152, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'it', 'beginIndex' : 258, 'endIndex' : 260, 'method' : 'coref_and_heursitics_pronouns'}], 'BRIDGE-1' : [{'mention' : 'Adams County, Pennsylvania.', 'beginIndex' : 122, 'endIndex' : 149, 'edit_distance' : 0.08, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'Adams County', 'beginIndex' : 182, 'endIndex' : 194, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-4' : [{'mention' : 'Contributing property,', 'beginIndex' : 53, 'endIndex' : 75, 'edit_distance' : 0.047619047619047616, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-5' : [{'mention' : 'Cumberland County,', 'beginIndex' : 315, 'endIndex' : 333, 'edit_distance' : 0.058823529411764705, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : 'Frederick County,', 'beginIndex' : 213, 'endIndex' : 230, 'edit_distance' : 0.0625, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : 'Carroll County,', 'beginIndex' : 265, 'endIndex' : 280, 'edit_distance' : 0.07142857142857142, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-6' : [{'mention' : 'United States.', 'beginIndex' : 167, 'endIndex' : 181, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : '2000', 'beginIndex' : 98, 'endIndex' : 102, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'BRIDGE-1', 'PATIENT-4', 'PATIENT-5', 'PATIENT-2', 'PATIENT-3', 'PATIENT-6', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	8
Number of entities in the RDF input	8

Model	RALI
Text ID	('Id129', 'Id1')
Text generated by the model	Abraham A. Ribicoff spouse Casey Ribicoff. Abraham A. Ribicoff death place United States. United States ethnic group African Americans
Example of reference text	In the United States, one of the national ethnic groups are the African Americans. The United States is where Abraham A. Ribicoff, who was married to Casey Ribicoff, died.
RDF Input	Abraham_A._Ribicoff | spouse | Casey_Ribicoff Abraham_A._Ribicoff | deathPlace | United_States United_States | ethnicGroup | African_Americans

Entitymap	{'AGENT-1' : 'Abraham_A._Ribicoff', 'BRIDGE-1' : 'United_States', 'PATIENT-1' : 'Casey_Ribicoff', 'PATIENT-2' : 'African_Americans'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Abraham A. Ribicoff', 'beginIndex' : 0, 'endIndex' : 19, 'edit_distance' : 0.1111111111111111, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : 'African Americans.', 'beginIndex' : 110, 'endIndex' : 128, 'edit_distance' : 0.058823529411764705, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : 'Casey Ribicoff', 'beginIndex' : 34, 'endIndex' : 48, 'edit_distance' : 0.07142857142857142, 'method' : 'basic_detection_0.4_1'}], 'BRIDGE-1' : [{'mention' : 'United States', 'beginIndex' : 68, 'endIndex' : 81, 'type' : 'LOC', 'method' : 'REL'}]}
Detected entities	['AGENT-1', 'PATIENT-2', 'PATIENT-1', 'BRIDGE-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	4
Number of entities in the RDF input	4
Model	RALI
Text ID	('Id1004', 'Id1')
Text generated by the model	Agremiação Sportiva Arapiraquense league Campeonato Brasileiro Série C. Campeonato Brasileiro Série C country Brazil. Agremiação Sportiva Arapiraquense number of members 17000. Agremiação Sportiva Arapiraquense nickname "Alvinegro
Example of reference text	Agremiação Sportiva Arapiraquense is a club that plays in Brazil's Campeonato Brasileiro Série C league. Also known as Alvinegro, the club represents 17,000 members.
RDF Input	Agremiação_Sportiva_Arapiraquense | league | Campeonato_Brasileiro_Série_C Campeonato_Brasileiro_Série_C | country | Brazil Agremiação_Sportiva_Arapiraquense | numberOfMembers | 17000 Agremiação_Sportiva_Arapiraquense | nickname | "Alvinegro"
Entitymap	{'AGENT-1' : 'Agremiação_Sportiva_Arapiraquense', 'BRIDGE-1' : 'Campeonato_Brasileiro_Série_C', 'PATIENT-1' : 'Brazil', 'PATIENT-2' : '17000', 'PATIENT-3' : '"Alvinegro"'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Agremiação Sportiva Arapiraquense', 'beginIndex' : 0, 'endIndex' : 33, 'type' : 'ORG', 'method' : 'REL'}, {'mention' : 'it', 'beginIndex' : 54, 'endIndex' : 56, 'method' : 'coref_and_heursitics_pronouns'}], 'BRIDGE-1' : [{'mention' : 'Campeonato Brasileiro Série C', 'beginIndex' : 107, 'endIndex' : 136, 'edit_distance' : 0.10344827586206896, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : '"Alvinegro", 'beginIndex' : 75, 'endIndex' : 86, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : 'Brazil', 'beginIndex' : 148, 'endIndex' : 154, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-2' : [{'mention' : '17,000', 'beginIndex' : 38, 'endIndex' : 44, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'BRIDGE-1', 'PATIENT-3', 'PATIENT-1', 'PATIENT-2']

Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	5
Number of entities in the RDF input	5
Model	RALI
Text ID	('Id1658', 'Id1')
Text generated by the model	Nurhan Atasoy residence Istanbul. Nurhan Atasoy birth place Turkey. Nurhan Atasoy nationality Turkish people. Nurhan Atasoy residence Turkey
Example of reference text	Nurhan Atasoy, whose nationality is Turkish, was born in Turkey. He resides in Turkey in Istanbul.
RDF Input	Nurhan_Atasoy | residence | Istanbul Nurhan_Atasoy | birthPlace | Turkey Nurhan_Atasoy | nationality | Turkish_people Nurhan_Atasoy | residence | Turkey
Entitymap	{'AGENT-1' : 'Nurhan_Atasoy', 'PATIENT-1' : 'Istanbul', 'PATIENT-2' : 'Turkey', 'PATIENT-3' : 'Turkish_people'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Nurhan Atasoy', 'beginIndex' : 0, 'endIndex' : 13, 'type' : 'PER', 'method' : 'REL'}, {'mention' : 'she', 'beginIndex' : 43, 'endIndex' : 46, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-1' : [{'mention' : 'Istanbul', 'beginIndex' : 81, 'endIndex' : 89, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-2' : [{'mention' : 'Turkish', 'beginIndex' : 27, 'endIndex' : 34, 'type' : 'MISC', 'method' : 'REL'}, {'mention' : 'Turkey', 'beginIndex' : 59, 'endIndex' : 65, 'type' : 'LOC', 'method' : 'REL'}, {'mention' : 'Turkey', 'beginIndex' : 94, 'endIndex' : 100, 'type' : 'LOC', 'method' : 'REL'}]}
Detected entities	['AGENT-1', 'PATIENT-1', 'PATIENT-2']
Undetected entities	['PATIENT-3']
Number of undetected entities	1
ESA_I	0.25
Number of entities in generated text	4
Number of entities in the RDF input	4
Model	FBConvAI
Text ID	('Id893', 'Id1')
Text generated by the model	McVeagh of the South Seas was written by Harry Carey (actor born 1878) and directed by Cyril Bruce. It stars Harry Carey who was born in 1878. The IMDB id of McVeagh of the South Seas is 0004319.
Example of reference text	Harry Carey, born in 1878, wrote, directed and starred in the movie McVeagh of the South Seas which was also directed by Cyril Bruce and registered in IMDb with the ID 0004319.
	McVeagh_of_the_South_Seas | imdbId | 0004319 McVeagh_of_the_South_Seas | director | Cyril_Bruce McVeagh_of_the_South_Seas | director | Harry_Carey_(actor_born_1878)

Entitymap	McVeagh_of_the_South_Seas | starring | Harry_Carey_(actor_born_1878) McVeagh_of_the_South_Seas | writer | Harry_Carey_(actor_born_1878) {'AGENT-1' : 'McVeagh_of_the_South_Seas', 'PATIENT-1' : '0004319', 'PATIENT-2' : 'Cyril_Bruce', 'PATIENT-3' : 'Harry_Carey_(actor_born_1878)'}
Detected mentions	{'AGENT-1' : [{'mention' : 'McVeagh of the South Seas', 'beginIndex' : 0, 'endIndex' : 25, 'edit_distance' : 0.16, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'It', 'beginIndex' : 100, 'endIndex' : 102, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-2' : [{'mention' : 'Cyril Bruce.', 'beginIndex' : 87, 'endIndex' : 99, 'edit_distance' : 0.09090909090909091, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : 'Harry Carey', 'beginIndex' : 41, 'endIndex' : 52, 'edit_distance' : 0.09090909090909091, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : '0004319.', 'beginIndex' : 187, 'endIndex' : 195, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-2', 'PATIENT-3', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	4
Number of entities in the RDF input	4

Model	FBConvAI
Text ID	('Id1159', 'Id1')
Text generated by the model	Aleksandr Prudnikov is 185 cm tall and played for FC Spartak Moscow's youth team. His current club is FC Amkar Perm and he plays for FC Terek Grozny, the ground of which, is based in Grozny.
Example of reference text	Aleksandr Prudnikov, height 185 cm., a member of the youth side of FC Spartak Moscow, and who plays for FC Amkar Perm is in the FC Terek Grozny club based at Grozny.
Entitymap	FC_Terek_Grozny | ground | Grozny Aleksandr_Prudnikov | currentclub | FC_Amkar_Perm Aleksandr_Prudnikov | club | FC_Terek_Grozny Aleksandr_Prudnikov | height | 185.0 (centimetres) Aleksandr_Prudnikov | youthclub | FC_Spartak_Moscow {'AGENT-1' : 'Aleksandr_Prudnikov', 'BRIDGE-1' : 'FC_Terek_Grozny', 'PATIENT-1' : 'Grozny', 'PATIENT-2' : 'FC_Amkar_Perm', 'PATIENT-3' : '185.0 (centimetres)', 'PATIENT-4' : 'FC_Spartak_Moscow'}

Detected mentions	{'AGENT-1' : [{'mention' : 'Aleksandr Prudnikov', 'beginIndex' : 0, 'endIndex' : 19, 'type' : 'PER', 'method' : 'REL'}, {'mention' : 'His', 'beginIndex' : 82, 'endIndex' : 85, 'method' : 'coref_and_heursitics_pronouns'}, {'mention' : 'he', 'beginIndex' : 120, 'endIndex' : 122, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-4' : [{'mention' : 'FC Spartak Moscow', 'beginIndex' : 50, 'endIndex' : 67, 'type' : 'ORG', 'method' : 'REL'}], 'PATIENT-2' : [{'mention' : 'FC Amkar Perm', 'beginIndex' : 102, 'endIndex' : 115, 'type' : 'ORG', 'method' : 'REL'}], 'PATIENT-1' : [{'mention' : 'Grozny,', 'beginIndex' : 142, 'endIndex' : 149, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'Grozny', 'beginIndex' : 183, 'endIndex' : 189, 'type' : 'LOC', 'method' : 'REL'}], 'PATIENT-3' : [{'mention' : '185', 'beginIndex' : 23, 'endIndex' : 26, 'edit_distance' : 0.25, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-4', 'PATIENT-2', 'PATIENT-1', 'PATIENT-3']
Undetected entities	['BRIDGE-1']
Number of undetected entities	1
ESA_I	0.16666666666666666
Number of entities in generated text	6
Number of entities in the RDF input	6

Model	FBConvAI
Text ID	('Id820', 'Id1')
Text generated by the model	The sequel to The Two Towers is The Return of the King.
Example of reference text	The Two Towers was followed by The Return of the King.
Entitymap	The_Two_Towers | followedBy | The_Return_of_the_King {'AGENT-1' : 'The_Two_Towers', 'PATIENT-1' : 'The_Return_of_the_King'}
Detected mentions	{'PATIENT-1' : [{'mention' : 'The Return of the King.', 'beginIndex' : 32, 'endIndex' : 55, 'edit_distance' : 0.18181818181818182, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'The Two Towers', 'beginIndex' : 14, 'endIndex' : 28, 'type' : 'ORG', 'method' : 'REL'}]}
Detected entities	['PATIENT-1', 'AGENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	2
Number of entities in the RDF input	2

Model	FBConvAI
Text ID	('Id443', 'Id1')
Text generated by the model	Olga Bondareva's profession is Mathematics.
Example of reference text	Olga Bondareva's professional field is mathematics.
	Olga_Bondareva | professionalField | Mathematics

Entitymap	{'AGENT-1' : 'Olga_Bondareva', 'PATIENT-1' : 'Mathematics'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Olga Bondareva', 'beginIndex' : 0, 'endIndex' : 14, 'type' : 'PER', 'method' : 'REL'}], 'PATIENT-1' : [{'mention' : 'Mathematics.', 'beginIndex' : 31, 'endIndex' : 43, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	2
Number of entities in the RDF input	2
Model	FBConvAI
Text ID	('Id1261', 'Id1')
Text generated by the model	Hypermarcas, a pharmaceutical company located in Brazil, is managed by their CEO.
Example of reference text	Located in Brazil, Hypermarcas is an S.A. corporation that produces pharmaceutical drugs. It is headed by the CEO.
Entitymap	{'AGENT-1' : 'Hypermarcas', 'PATIENT-1' : 'Brazil', 'PATIENT-2' : 'Drugs', 'PATIENT-3' : 'Pharmaceuticals', 'PATIENT-4' : 'CEO', 'PATIENT-5' : 'S.A._(corporation)'}
Detected mentions	{'PATIENT-3' : [{'mention' : 'pharmaceutical', 'beginIndex' : 15, 'endIndex' : 29, 'edit_distance' : 0.06666666666666667, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'Hypermarcas,', 'beginIndex' : 0, 'endIndex' : 12, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'their', 'beginIndex' : 71, 'endIndex' : 76, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-1' : [{'mention' : 'Brazil,', 'beginIndex' : 49, 'endIndex' : 56, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-4' : [{'mention' : 'CEO.', 'beginIndex' : 77, 'endIndex' : 81, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['PATIENT-3', 'AGENT-1', 'PATIENT-1', 'PATIENT-4']
Undetected entities	['PATIENT-2', 'PATIENT-5']
Number of undetected entities	2
ESA_I	0.3333333333333333
Number of entities in generated text	4
Number of entities in the RDF input	6
Model	CycleGT
Text ID	('Id799', 'Id1')
Text generated by the model	the University of Burgundy, Dijon, has 1299 doctoral students and 2900 staff.
Example of reference text	The University of Burgundy, located in Dijon, employs 2900 staff members including 1299 doctoral students. University_of_Burgundy | staff | 2900

Entitymap	University_of_Burgundy | numberOfDoctoralStudents | 1299 University_of_Burgundy | campus | Dijon {'AGENT-1' : 'University_of_Burgundy', 'PATIENT-1' : '2900', 'PATIENT-2' : '1299', 'PATIENT-3' : 'Dijon'}
Detected mentions	{'AGENT-1' : [{'mention' : 'University of Burgundy,', 'beginIndex' : 4, 'endIndex' : 27, 'edit_distance' : 0.09090909090909091, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : 'Dijon,', 'beginIndex' : 28, 'endIndex' : 34, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : '2900', 'beginIndex' : 66, 'endIndex' : 70, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : '1299', 'beginIndex' : 39, 'endIndex' : 43, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-3', 'PATIENT-1', 'PATIENT-2']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	4
Number of entities in the RDF input	4

Model	CycleGT
Text ID	('Id125', 'Id1')
Text generated by the model	Olga Bondareva was born on 1937-04-27 and died on 1991-12-09. her alma mater was Leningrad State University.
Example of reference text	Olga Bondareva, who lived from April 27th, 1937 to December 9th, 1991, was a student at Leningrad State University.
Entitymap	Olga_Bondareva | birthDate | 1937-04-27 Olga_Bondareva | almaMater | Leningrad_State_University Olga_Bondareva | deathDate | 1991-12-09 {'AGENT-1' : 'Olga_Bondareva', 'PATIENT-1' : '1937-04-27', 'PATIENT-2' : 'Leningrad_State_University', 'PATIENT-3' : '1991-12-09'}
Detected mentions	{'PATIENT-2' : [{'mention' : 'Leningrad State University.', 'beginIndex' : 81, 'endIndex' : 108, 'edit_distance' : 0.07692307692307693, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'Olga Bondareva', 'beginIndex' : 0, 'endIndex' : 14, 'type' : 'PER', 'method' : 'REL'}, {'mention' : 'her', 'beginIndex' : 62, 'endIndex' : 65, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-3' : [{'mention' : '1991-12-09.', 'beginIndex' : 50, 'endIndex' : 61, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : '1937-04-27', 'beginIndex' : 27, 'endIndex' : 37, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['PATIENT-2', 'AGENT-1', 'PATIENT-3', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0

Number of entities in generated text	4
Number of entities in the RDF input	4
Model	CycleGT
Text ID	('Id1084', 'Id1')
Text generated by the model	Bootleg Series Volume 1 : The Quine Tapes was recorded in San Francisco, which is in the Pacific Time Zone. San Francisco is the location of St. Louis, Missouri.
Example of reference text	The location that the Bootleg Series Volume 1 : The Quine Tapes was recorded in was St. Louis, Missouri, and San Francisco, which is in the Pacific Time Zone.
Entitymap	Bootleg_Series_Volume_1 :_The_Quine_Tapes | recordedIn | St._Louis,_Missouri San_Francisco | timeZone | Pacific_Time_Zone Bootleg_Series_Volume_1 :_The_Quine_Tapes | recordedIn | San_Francisco {'AGENT-1' : 'Bootleg_Series_Volume_1 :_The_Quine_Tapes', 'BRIDGE-1' : 'San_Francisco', 'PATIENT-1' : 'St._Louis,_Missouri', 'PATIENT-2' : 'Pacific_Time_Zone'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Bootleg Series Volume 1 : The Quine Tapes', 'beginIndex' : 0, 'endIndex' : 40, 'edit_distance' : 0.15384615384615385, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : 'St. Louis, Missouri.', 'beginIndex' : 140, 'endIndex' : 160, 'edit_distance' : 0.11764705882352941, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : 'Pacific Time Zone.', 'beginIndex' : 88, 'endIndex' : 106, 'edit_distance' : 0.11764705882352941, 'method' : 'basic_detection_0.4_1'}], 'BRIDGE-1' : [{'mention' : 'San Francisco,', 'beginIndex' : 57, 'endIndex' : 71, 'edit_distance' : 0.07692307692307693, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'San Francisco', 'beginIndex' : 107, 'endIndex' : 120, 'type' : 'LOC', 'method' : 'REL'}]}
Detected entities	['AGENT-1', 'PATIENT-1', 'PATIENT-2', 'BRIDGE-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	4
Number of entities in the RDF input	4
Model	CycleGT
Text ID	('Id1013', 'Id1')
Text generated by the model	Liselotte Grschebina was born in the German Empire on 1908-05-02 and died on 1994-06-14.
Example of reference text	Liselotte Grschebina was born in Karlsruhe in the German Empire on May 2nd, 1908 and she died on June 14, 1994.

	Liselotte_Grschebina | birthPlace | Karlsruhe Liselotte_Grschebina | birthDate | 1908-05-02 Liselotte_Grschebina | deathDate | 1994-06-14 Liselotte_Grschebina | birthPlace | German_Empire
Entitymap	{'AGENT-1' : 'Liselotte_Grschebina', 'PATIENT-1' : 'Karlsruhe', 'PATIENT-2' : '1908-05-02', 'PATIENT-3' : '1994-06-14', 'PATIENT-4' : 'German_Empire'}
Detected mentions	{'AGENT-1' : [{'mention' : 'Liselotte Grschebina', 'beginIndex' : 0, 'endIndex' : 20, 'edit_distance' : 0.05, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : '1994-06-14.', 'beginIndex' : 77, 'endIndex' : 88, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : '1908-05-02', 'beginIndex' : 54, 'endIndex' : 64, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-4' : [{'mention' : 'German', 'beginIndex' : 37, 'endIndex' : 43, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['AGENT-1', 'PATIENT-3', 'PATIENT-2', 'PATIENT-4']
Undetected entities	['PATIENT-1']
Number of undetected entities	1
ESA_I	0.2
Number of entities in generated text	5
Number of entities in the RDF input	5

Model	CycleGT
Text ID	('Id829', 'Id1')
Text generated by the model	Super Capers, starring Justin Whalin, is distributed by Lionsgate and is part of the Roadside Attractions franchise, which was founded on 2003-01-01.
Example of reference text	Super Capers was a film starring Justin Whalin distributed by Lionsgate and Roadside Attractions. Attractions was founded in 2003.
	Super_Capers | distributor | Lionsgate Super_Capers | starring | Justin_Whalin Super_Capers | distributor | Roadside_Attractions Roadside_Attractions | foundingYear | 2003-01-01
Entitymap	{'AGENT-1' : 'Super_Capers', 'BRIDGE-1' : 'Roadside_Attractions', 'PATIENT-1' : 'Lionsgate', 'PATIENT-2' : 'Justin_Whalin', 'PATIENT-3' : '2003-01-01'}

Detected mentions	{'BRIDGE-1' : [{'mention' : 'Roadside Attractions', 'beginIndex' : 85, 'endIndex' : 105, 'edit_distance' : 0.05, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : 'Justin Whalin,', 'beginIndex' : 23, 'endIndex' : 37, 'edit_distance' : 0.07692307692307693, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'Super Capers,', 'beginIndex' : 0, 'endIndex' : 13, 'edit_distance' : 0.08333333333333333, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-3' : [{'mention' : '2003-01-01.', 'beginIndex' : 138, 'endIndex' : 149, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : 'Lionsgate', 'beginIndex' : 56, 'endIndex' : 65, 'type' : 'ORG', 'method' : 'REL'}]}
Detected entities	['BRIDGE-1', 'PATIENT-2', 'AGENT-1', 'PATIENT-3', 'PATIENT-1']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0
Number of entities in generated text	5
Number of entities in the RDF input	5

Model	CycleGT
Text ID	('Id458', 'Id1')
Text generated by the model	Mermaid (Train song) is a Pop rock song with a runtime of 3.16. it is released by Columbia Records and Sony Music Entertainment.
Example of reference text	The Train song Mermaid, a pop rock genre, on Columbia Records and Sony Music Entertainment record label has a total runtime of three minutes and sixteen seconds.
Entitymap	Mermaid_(Train_song) | genre | Pop_rock Mermaid_(Train_song) | recordLabel | Columbia_Records Mermaid_(Train_song) | runtime | 3.16 Mermaid_(Train_song) | recordLabel | Sony_Music_Entertainment {'AGENT-1' : 'Mermaid_(Train_song)', 'PATIENT-1' : 'Pop_rock', 'PATIENT-2' : 'Columbia_Records', 'PATIENT-3' : '3.16', 'PATIENT-4' : 'Sony_Music_Entertainment'}
Detected mentions	{'PATIENT-4' : [{'mention' : 'Sony Music Entertainment.', 'beginIndex' : 103, 'endIndex' : 128, 'edit_distance' : 0.08333333333333333, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-2' : [{'mention' : 'Columbia Records', 'beginIndex' : 82, 'endIndex' : 98, 'edit_distance' : 0.0625, 'method' : 'basic_detection_0.4_1'}], 'PATIENT-1' : [{'mention' : 'Pop rock', 'beginIndex' : 26, 'endIndex' : 34, 'edit_distance' : 0.125, 'method' : 'basic_detection_0.4_1'}], 'AGENT-1' : [{'mention' : 'Mermaid', 'beginIndex' : 0, 'endIndex' : 7, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}, {'mention' : 'it', 'beginIndex' : 64, 'endIndex' : 66, 'method' : 'coref_and_heursitics_pronouns'}], 'PATIENT-3' : [{'mention' : '3.16.', 'beginIndex' : 58, 'endIndex' : 63, 'edit_distance' : 0.0, 'method' : 'basic_detection_0.4_1'}]}
Detected entities	['PATIENT-4', 'PATIENT-2', 'PATIENT-1', 'AGENT-1', 'PATIENT-3']
Undetected entities	[]
Number of undetected entities	0
ESA_I	0.0

Number of entities in generated text	5
Number of entities in the RDF input	5

A.2 Correlation Results WebNLG 2017

A.2.1 Correlation Results only for texts with at least one undetected entity

For Spearman correlation results, see table A.2.
For Kendall's tau correlation results, see table A.3.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA$_I$
BLEU	0.89	-0.78	0.56	0.59	0.68	0.76
METEOR	x	-0.78	0.61	0.65	0.77	**0.84**
TER	x	x	-0.54	-0.57	-0.58	-0.63
Fluency	x	x	x	0.91	0.57	0.48
Grammar	x	x	x	x	0.61	0.53
Semantics	x	x	x	x	x	**0.73**

TABLE A.2 – Spearman correlation coefficients for WebNLG 2017 metrics and ESA$_I$. Only for texts with at least one undetected entity (i.e. 822 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of ESA$_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA$_I$
BLEU	0.74	-0.6	0.42	0.45	0.52	0.6
METEOR	x	-0.58	0.46	0.5	0.62	**0.68**
TER	x	x	-0.4	-0.42	-0.43	-0.46
Fluency	x	x	x	0.81	0.46	0.36
Grammar	x	x	x	x	0.5	0.41
Semantics	x	x	x	x	x	**0.59**

TABLE A.3 – Kendall's tau coefficients for WebNLG 2017 metrics and ESA$_I$. Only for texts with at least one undetected entity (i.e. 822 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of ESA$_I$with surface-based (top block) and human evaluation (bottom block) metrics.

A.2.2 Correlation Results only for all texts

For Pearson correlation results, see table A.4.
For Spearman correlation results, see table A.5.
For Kendall's tau correlation results, see table A.6.

A.2.3 Correlation Results only for texts with at least two undetected entities

For Pearson correlation results, see table A.7.
For Spearman correlation results, see table A.8.
For Kendall's tau correlation results, see table A.9.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.84	-0.81	0.36	0.35	0.44	0.4
METEOR	x	-0.83	0.53	0.52	0.64	**0.65**
TER	x	x	-0.49	-0.47	-0.55	-0.52
Fluency	x	x	x	0.87	0.61	0.49
Grammar	x	x	x	x	0.62	0.53
Semantics	x	x	x	x	x	**0.72**

TABLE A.4 – Pearson correlation coefficients for WebNLG 2017 metrics and ESA_I (for 2230 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $\text{ESA}_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.66	-0.66	0.33	0.34	0.45	0.46
METEOR	x	-0.68	0.42	0.41	0.57	**0.59**
TER	x	x	-0.39	-0.37	-0.46	-0.46
Fluency	x	x	x	0.75	0.49	0.35
Grammar	x	x	x	x	0.5	0.35
Semantics	x	x	x	x	x	**0.59**

TABLE A.5 – Spearman correlation coefficients for WebNLG 2017 metrics and ESA_I (for 2230 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $\text{ESA}_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.66	-0.66	0.33	0.34	0.45	0.46
METEOR	x	-0.68	0.42	0.41	0.57	**0.58**
TER	x	x	-0.39	-0.37	-0.46	-0.46
Fluency	x	x	x	0.75	0.49	0.35
Grammar	x	x	x	x	0.5	0.35
Semantics	x	x	x	x	x	**0.59**

TABLE A.6 – Kendall'tau coefficients for WebNLG 2017 metrics and ESA_I (for 2230 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $\text{ESA}_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.79	-0.37	0.43	0.46	0.51	0.65
METEOR	x	-0.26	0.57	0.62	0.67	**0.9**
TER	x	x	-0.32	-0.33	-0.18	-0.16
Fluency	x	x	x	0.9	0.48	0.51
Grammar	x	x	x	x	0.53	0.57
Semantics	x	x	x	x	x	**0.6**

TABLE A.7 – Pearson correlation coefficients for WebNLG 2017 metrics and ESA_I, only for texts with at least 2 undetected entities (i.e. 469 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $\text{ESA}_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.94	-0.57	0.68	0.73	0.64	0.84
METEOR	x	-0.56	0.7	0.74	0.71	**0.9**
TER	x	x	-0.51	-0.53	-0.39	-0.48
Fluency	x	x	x	0.92	0.55	0.57
Grammar	x	x	x	x	0.58	0.62
Semantics	x	x	x	x	x	**0.69**

TABLE A.8 – Spearman correlation coefficients for WebNLG 2017 metrics and ESA_I, only for texts with at least 2 undetected entities (i.e. 469 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $ESA_I$with surface-based (top block) and human evaluation (bottom block) metrics.

Metrics	METEOR	TER	Fluency	Grammar	Semantics	ESA_I
BLEU	0.81	-0.44	0.54	0.58	0.53	0.7
METEOR	x	-0.43	0.55	0.59	0.59	**0.79**
TER	x	x	-0.38	-0.39	-0.3	-0.37
Fluency	x	x	x	0.84	0.47	0.47
Grammar	x	x	x	x	0.5	0.51
Semantics	x	x	x	x	x	**0.6**

TABLE A.9 – Kendall's tau coefficients for WebNLG 2017 metrics and ESA_I, only for texts with at least 2 undetected entities (i.e. 469 texts). All the p-values are <0.01. Bold numbers indicate the highest correlations of $ESA_I$with surface-based (top block) and human evaluation (bottom block) metrics.

A.3 Correlation Results WebNLG 2020

A.3.1 Correlation Results only for texts with at least one undetected entity

For Spearman correlation results, see table A.10.
For Kendall's tau correlation results, see table A.11.

A.3.2 Correlation results for all texts

For Pearson correlation results, see table A.12.
For Spearman correlation results, see table A.13.
For Kendall's tau correlation results, see table A.14.

A.3.3 Correlation results for texts with at least two undetected entities

For Pearson correlation results, see table A.15.
For Spearman correlation results, see table A.16.
For Kendall's tau correlation results, see table A.17.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.97	0.71	0.83	-0.74	0.71	0.67	0.72	0.49	0.42	0.3	0.34	0.35	0.3	**0.44**
2 BLEU NLTK	x	0.76	0.88	-0.82	0.76	0.73	0.78	0.55	0.46	0.34	0.39	0.4	0.36	<u>0.43</u>
3 METEOR	x	x	0.9	-0.71	0.69	0.82	0.79	0.66	0.49	0.47	0.41	0.42	0.38	0.39
4 chrF++	x	x	x	-0.77	0.76	0.82	0.83	0.59	0.49	0.43	0.4	0.42	0.36	**0.44**
5 TER	x	x	x	x	-0.82	-0.75	-0.82	-0.65	-0.46	-0.32	-0.47	-0.42	-0.44	-0.24
6 BERT-SCORE P	x	x	x	x	x	0.83	0.95	0.7	0.59	0.39	0.52	0.54	0.48	0.38
7 BERT-SCORE R	x	x	x	x	x	x	0.95	0.74	0.59	0.53	0.52	0.52	0.47	**0.43**
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.75	0.62	0.48	0.54	0.55	0.5	<u>0.42</u>
9 BLEURT	x	x	x	x	x	x	x	x	0.6	0.52	0.51	0.55	0.51	0.36
10 Correctness	x	x	x	x	x	x	x	x	x	0.72	0.69	0.78	0.66	**0.48**
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.57	0.66	0.53	**0.48**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.61	0.82	0.32
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.61	<u>0.41</u>
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.26
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.10 – Spearman correlation coefficients for WebNLG 2020 metrics with ESA_I. Only for texts with at least one undetected entity (i.e. 470 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.9	0.55	0.67	-0.58	0.56	0.52	0.58	0.36	0.3	0.21	0.24	0.25	0.22	<u>0.32</u>
2 BLEU NLTK	x	0.59	0.71	-0.64	0.6	0.56	0.62	0.4	0.32	0.24	0.27	0.28	0.25	0.31
3 METEOR	x	x	0.74	-0.54	0.53	0.66	0.63	0.49	0.34	0.33	0.29	0.29	0.26	0.29
4 chrF++	x	x	x	-0.59	0.59	0.65	0.66	0.42	0.34	0.3	0.28	0.29	0.25	**0.33**
5 TER	x	x	x	x	-0.66	-0.58	-0.66	-0.49	-0.32	-0.22	-0.33	-0.3	-0.31	-0.17
6 BERT-SCORE P	x	x	x	x	x	0.7	0.87	0.55	0.43	0.28	0.38	0.39	0.35	0.28
7 BERT-SCORE R	x	x	x	x	x	x	0.86	0.58	0.43	0.38	0.38	0.38	0.34	**0.32**
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.6	0.46	0.35	0.4	0.4	0.36	<u>0.31</u>
9 BLEURT	x	x	x	x	x	x	x	x	0.43	0.37	0.37	0.39	0.36	0.26
10 Correctness	x	x	x	x	x	x	x	x	x	0.55	0.52	0.6	0.5	**0.35**
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.41	0.5	0.38	**0.35**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.45	0.65	0.23
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.44	<u>0.3</u>
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.18
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.11 – Kendall's tau coefficients for WebNLG 2020 metrics with ESA_I. Only for texts with at least one undetected entity (i.e. 470 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.97	0.73	0.82	-0.79	0.66	0.65	0.67	0.61	0.35	0.28	0.35	0.29	0.33	0.2
2 BLEU NLTK	x	0.77	0.87	-0.84	0.69	0.69	0.71	0.66	0.37	0.3	0.39	0.31	0.36	0.21
3 METEOR	x	x	0.83	-0.75	0.64	0.7	0.68	0.69	0.35	0.31	0.34	0.29	0.32	0.23
4 chrF++	x	x	x	-0.82	0.79	0.85	0.83	0.77	0.44	0.39	0.4	0.37	0.38	**0.28**
5 TER	x	x	x	x	-0.75	-0.72	-0.75	-0.75	-0.41	-0.33	-0.43	-0.37	-0.41	-0.22
6 BERT-SCORE P	x	x	x	x	x	0.93	0.98	0.81	0.43	0.34	0.42	0.38	0.41	0.19
7 BERT-SCORE R	x	x	x	x	x	x	0.98	0.81	0.44	0.39	0.4	0.37	0.38	0.25
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.82	0.44	0.37	0.42	0.38	0.41	0.22
9 BLEURT	x	x	x	x	x	x	x	x	0.53	0.47	0.49	0.48	0.48	**0.3**
10 Correctness	x	x	x	x	x	x	x	x	x	0.76	0.65	0.79	0.64	0.46
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.53	0.79	0.52	**0.52**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.56	0.87	0.28
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.57	0.41
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.26
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.12 – Pearson correlation coefficients for WebNLG 2020 metrics with ESA_I (for 2848 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between ESA_I and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.98	0.77	0.82	-0.8	0.74	0.73	0.75	0.62	0.32	0.25	0.35	0.26	0.33	0.17
2 BLEU NLTK	x	0.81	0.86	-0.84	0.77	0.77	0.79	0.67	0.36	0.28	0.39	0.3	0.37	0.18
3 METEOR	x	x	0.92	-0.8	0.75	0.84	0.82	0.75	0.42	0.37	0.4	0.34	0.37	**0.28**
4 chrF++	x	x	x	-0.83	0.78	0.86	0.84	0.73	0.41	0.37	0.4	0.34	0.38	0.26
5 TER	x	x	x	x	-0.84	-0.81	-0.85	-0.78	-0.4	-0.33	-0.45	-0.35	-0.43	-0.21
6 BERT-SCORE P	x	x	x	x	x	0.9	0.97	0.8	0.42	0.31	0.46	0.35	0.45	0.18
7 BERT-SCORE R	x	x	x	x	x	x	0.97	0.84	0.46	0.39	0.45	0.37	0.44	0.26
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.84	0.45	0.36	0.47	0.37	0.45	0.23
9 BLEURT	x	x	x	x	x	x	x	x	0.49	0.42	0.51	0.4	0.49	**0.3**
10 Correctness	x	x	x	x	x	x	x	x	x	0.72	0.62	0.73	0.62	0.33
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.47	0.73	0.5	**0.38**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.49	0.84	0.22
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.53	0.26
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.21
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.13 – Spearman correlation coefficients for WebNLG 2020 metrics with ESA_I (for 2848 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between ESA_I and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.92	0.61	0.66	-0.63	0.59	0.58	0.6	0.47	0.23	0.18	0.25	0.19	0.23	0.14
2 BLEU NLTK	x	0.64	0.7	-0.67	0.62	0.61	0.63	0.5	0.25	0.2	0.27	0.21	0.26	0.14
3 METEOR	x	x	0.79	-0.63	0.6	0.7	0.67	0.58	0.3	0.26	0.28	0.24	0.26	**0.23**
4 chrF++	x	x	x	-0.66	0.62	0.71	0.69	0.56	0.29	0.26	0.28	0.24	0.27	0.21
5 TER	x	x	x	x	-0.69	-0.65	-0.7	-0.61	-0.29	-0.23	-0.32	-0.25	-0.31	-0.17
6 BERT-SCORE P	x	x	x	x	x	0.79	0.9	0.65	0.31	0.22	0.34	0.25	0.32	0.15
7 BERT-SCORE R	x	x	x	x	x	x	0.9	0.68	0.34	0.28	0.33	0.27	0.32	0.22
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.69	0.33	0.26	0.34	0.27	0.33	0.19
9 BLEURT	x	x	x	x	x	x	x	x	0.35	0.3	0.36	0.28	0.35	**0.24**
10 Correctness	x	x	x	x	x	x	x	x	x	0.57	0.46	0.58	0.47	0.27
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.35	0.6	0.36	**0.31**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.36	0.68	0.18
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.4	0.21
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.17
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.14 – Kendall's tau correlation coefficients for WebNLG 2020 metrics with ESA_I (for 2848 texts). All the p-values are <0.01. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.97	0.77	0.89	-0.62	0.67	0.75	0.76	0.52	0.43	0.32	0.31	0.38	0.26	0.41
2 BLEU NLTK	x	0.81	0.91	-0.65	0.68	0.79	0.78	0.55	0.41	0.34	0.3	0.35	0.26	0.38
3 METEOR	x	x	0.88	-0.48	0.52	0.8	0.69	0.62	0.42	0.52	0.35	0.38	0.28	**0.45**
4 chrF++	x	x	x	-0.6	0.67	0.83	0.79	0.6	0.45	0.42	0.31	0.42	0.25	0.44
5 TER	x	x	x	x	-0.71	-0.6	-0.7	-0.54	-0.37	-0.26	-0.26	-0.38	(-0.24)	(-0.18)
6 BERT-SCORE P	x	x	x	x	x	0.76	0.94	0.68	0.6	0.31	0.49	0.58	0.42	0.32
7 BERT-SCORE R	x	x	x	x	x	x	0.92	0.72	0.48	0.45	0.4	0.44	0.32	0.31
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.74	0.58	0.4	0.49	0.55	0.42	0.36
9 BLEURT	x	x	x	x	x	x	x	x	0.61	0.5	0.49	0.52	0.42	**0.37**
10 Correctness	x	x	x	x	x	x	x	x	x	0.7	0.81	0.86	0.72	**0.6**
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.61	0.66	0.5	**0.6**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.78	0.81	0.47
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.69	0.58
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.39
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.15 – Pearson correlation coefficients for WebNLG 2020 metrics with ESA_I, only for texts with at least 2 undetected entities (i.e. 104 texts). All the p-values are <0.01, except the ones in brackets. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.98	0.76	0.89	-0.8	0.69	0.73	0.76	0.49	0.41	0.29	0.29	0.4	0.29	0.47
2 BLEU NLTK	x	0.8	0.91	-0.84	0.69	0.76	0.77	0.51	0.41	0.3	0.27	0.39	0.28	0.47
3 METEOR	x	x	0.86	-0.63	0.52	0.79	0.68	0.58	0.38	0.47	0.35	0.38	0.34	0.45
4 chrF++	x	x	x	-0.77	0.67	0.82	0.78	0.55	0.43	0.37	0.28	0.42	0.31	**0.49**
5 TER	x	x	x	x	-0.77	-0.7	-0.8	-0.57	-0.47	(-0.23)	-0.35	-0.43	-0.32	-0.33
6 BERT-SCORE P	x	x	x	x	x	0.75	0.94	0.63	0.61	0.31	0.5	0.59	0.47	0.31
7 BERT-SCORE R	x	x	x	x	x	x	0.91	0.69	0.54	0.51	0.44	0.51	0.43	**0.41**
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.72	0.63	0.43	0.51	0.6	0.49	0.4
9 BLEURT	x	x	x	x	x	x	x	x	0.62	0.5	0.5	0.5	0.46	0.34
10 Correctness	x	x	x	x	x	x	x	x	x	0.64	0.74	0.77	0.72	0.43
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.54	0.46	0.46	**0.5**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.69	0.79	0.32
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.69	0.4
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.27
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.16 – Spearman correlation coefficients for WebNLG 2020 metrics with ESA_I, only for texts with at least 2 undetected entities (i.e. 104 texts). All the p-values are <0.01, except the ones in brackets. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

Metrics	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 BLEU	0.91	0.59	0.74	-0.64	0.52	0.58	0.59	0.36	0.29	0.21	0.19	0.29	0.2	0.34
2 BLEU NLTK	x	0.62	0.76	-0.67	0.52	0.61	0.6	0.37	0.28	0.22	0.18	0.28	0.2	0.33
3 METEOR	x	x	0.7	-0.48	0.39	0.64	0.53	0.42	0.27	0.34	0.25	0.27	0.24	0.34
4 chrF++	x	x	x	-0.6	0.51	0.66	0.62	0.4	0.3	0.27	0.19	0.31	0.22	**0.35**
5 TER	x	x	x	x	-0.62	-0.56	-0.65	-0.42	-0.33	(-0.17)	-0.25	-0.31	-0.23	-0.22
6 BERT-SCORE P	x	x	x	x	x	0.61	0.85	0.48	0.46	0.23	0.37	0.43	0.35	0.22
7 BERT-SCORE R	x	x	x	x	x	x	0.81	0.53	0.39	0.37	0.33	0.39	0.32	**0.31**
8 BERT-SCORE F1	x	x	x	x	x	x	x	0.56	0.47	0.32	0.38	0.45	0.36	0.29
9 BLEURT	x	x	x	x	x	x	x	x	0.45	0.36	0.35	0.35	0.33	0.24
10 Correctness	x	x	x	x	x	x	x	x	x	0.48	0.56	0.59	0.54	0.31
11 Data Coverage	x	x	x	x	x	x	x	x	x	x	0.4	0.34	0.34	**0.36**
12 Fluency	x	x	x	x	x	x	x	x	x	x	x	0.51	0.63	0.23
13 Relevance	x	x	x	x	x	x	x	x	x	x	x	x	0.53	0.3
14 TextStructure	x	x	x	x	x	x	x	x	x	x	x	x	x	0.19
15 ESA_I	x	x	x	x	x	x	x	x	x	x	x	x	x	1

TABLE A.17 – Kendall's tau correlation coefficients for WebNLG 2020 metrics with ESA_I, only for texts with at least 2 undetected entities (i.e. 104 texts). All the p-values are <0.01, except the ones in brackets. Bold (resp. underlined) numbers indicate the highest (resp. second highest) correlation scores between $\text{ESA}_I$and different categories of evaluation metrics, i.e. surface-based similarity metrics (top block), embedding-based similarity (middle block) and human evaluation metrics.

References

Eu ai act : first regulation on artificial intelligence, european parlament news. https://www.europarl.europa.eu/news/en/headlines/society/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence. Accessed : 2023-08-22.

Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4190–4197, Online. Association for Computational Linguistics.

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks.

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2021. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, pages 3554–3565, Online. Association for Computational Linguistics.

Oshin Agarwal, Mihir Kale, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2020. Machine translation aided bilingual data-to-text generation and semantic parsing. In Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 125–130, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice : Semantic propositional image caption evaluation. In European conference on computer vision, pages 382–398. Springer.

Vinsen Andreas, Genta Winata, and Ayu Purwarianti. 2021. A comparative study on language models for task-oriented dialogue systems. pages 1–5.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. Transactions of the Association for Computational Linguistics, 7 :597–610.

Ron Artstein and Massimo Poesio. 2008. Survey article: Inter-coder agreement for computational linguistics. Computational Linguistics, 34(4) :555–596.

Ali Assi, Hamid Mcheick, and Wajdi Dhifli. 2020. Data linking over rdf knowledge graphs : A survey. Concurrency and Computation : Practice and Experience, 32(19) :e5746.

Anusha Balakrishnan, Jinfeng Rao, Kartikeya Upasani, Michael White, and Rajen Subba. 2019. Constrained decoding for neural NLG from compositional representations in task-oriented dialogue. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 831–844, Florence, Italy. Association for Computational Linguistics.

Esma Balkir, Svetlana Kiritchenko, Isar Nejadgholi, and Kathleen Fraser. 2022. Challenges in applying explainability methods to improve the fairness of NLP models. In Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022), pages 80–92, Seattle, U.S.A. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado González, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, V. Richard Benjamins, Raja Chatila, and Francisco Herrera. 2019. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information Fusion, 58.

Yonatan Belinkov. 2021. Probing Classifiers: Promises, Shortcomings, and Advances. Computational Linguistics, pages 1–13.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. Transactions of the Association for Computational Linguistics, 7 :49–72.

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.

Manik Bhandari, Pranav Narayan Gour, Atabak Ashfaq, Pengfei Liu, and Graham Neubig. 2020. Re-evaluating evaluation in text summarization. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9347–9359, Online. Association for Computational Linguistics.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog.

Quentin Brabant, Gwenole Lecorve, Lina M. Rojas-Barahona, and Claire Gardent. 2023. Kgconv, a conversational corpus grounded in wikidata.

Dan Brickley and Libby Miller. 2000. The friend of a friend (foaf) project.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Michele Cafagna, Lina M. Rojas-Barahona, Kees van Deemter, and Albert Gatt. 2023. Interpreting vision and language generative models with semantic visual priors.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020a. The 2020 bilingual, bi-directional WebNLG+ shared task: Overview and evaluation results (WebNLG+ 2020). In Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 55–76, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina, editors. 2020b. Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+). Association for Computational Linguistics, Dublin, Ireland (Virtual).

Thiago Castro Ferreira, Diego Moussallem, Emiel Krahmer, and Sander Wubben. 2018. Enriching the WebNLG corpus. In Proceedings of the 11th International Conference on Natural Language Generation, pages 171–176, Tilburg University, The Netherlands. Association for Computational Linguistics.

Thiago Castro Ferreira, João Victor de Pinho Costa, Isabela Rigotto, Vitoria Portella, Gabriel Frota, Ana Luisa A. R. Guimarães, Adalberto Penna, Isabela Lee, Tayane A. Soares, Sophia Rolim, Rossana Cunha, Celso França, Ariel Santos, Rivaney F. Oliveira, Abisague Langbehn, Daniel Hasan Dalip, Marcos André Gonçalves, Rodrigo Bastos Fóscolo, and Adriana Pagano. 2021. Evaluating recognizing question entailment methods for a Portuguese community question-answering system about diabetes mellitus. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), pages 234–243, Held Online. INCOMA Ltd.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. CoRR, abs/2006.14799.

Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. 2020. Generating hierarchical explanations on text classification via feature interaction detection. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5578–5593, Online. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei,

Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In ICLR.

Edo Collins and Sabine Süsstrunk. 2019. Deep feature factorization for content-based image retrieval and localization. In IEEE International Conference on Image Processing (ICIP), pages 874–878.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Ian Covert, Scott M. Lundberg, and Su-In Lee. 2020. Understanding global feature contributions through additive importance measures. CoRR, abs/2004.00668.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In Advances in Neural Information Processing Systems, volume 28. Curran Associates, Inc.

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. 2020. A survey of the state of explainable AI for natural language processing. In Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, pages 447–459, Suzhou, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4884–4895, Florence, Italy. Association for Computational Linguistics.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia : Knowledge-powered conversational agents. In Proceedings of the International Conference on Learning Representations (ICLR).

Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. 2018. Explainable artificial intelligence: A survey. In 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pages 0210–0215.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5055–5070, Online. Association for Computational Linguistics.

Ondřej Dušek and Zdeněk Kasner. 2020. Evaluating semantic accuracy of data-to-text generation with natural language inference. In Proceedings of the 13th International Conference on Natural Language Generation, pages 131–137, Dublin, Ireland. Association for Computational Linguistics.

Nouha Dziri, Ehsan Kamalloo, Kory Mathewson, and Osmar Zaiane. 2019. Evaluating coherence in dialogue systems using entailment. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers), pages 3806–3812, Minneapolis, Minnesota. Association for Computational Linguistics.

Kawin Ethayarajh and Dan Jurafsky. 2021. Attention flows are shapley value explanations. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2 : Short Papers), pages 49–54, Online. Association for Computational Linguistics.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. Transactions of the Association for Computational Linguistics, 8 :34–48.

European Comission. 2021. Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts.
https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0206.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating summarization evaluation. Transactions of the Association for Computational Linguistics, 9 :391–409.

Angela Fan and Claire Gardent. 2022. Generating biographies on Wikipedia: The impact of gender bias on the retrieval-based generation of women biographies. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 8561–8576, Dublin, Ireland. Association for Computational Linguistics.

Angela Fan, Claire Gardent, Chloé Braud, and Antoine Bordes. 2019. Using local knowledge graph construction to scale Seq2Seq models to multi-document inputs. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4186–4196, Hong Kong, China. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, pages 1606–1615, Denver, Colorado. Association for Computational Linguistics.

Katja Filippova. 2020. Controlled hallucinations: Learning to generate faithfully from noisy data. In Findings of the Association for Computational Linguistics : EMNLP 2020, pages 864–870, Online. Association for Computational Linguistics.

Zihao Fu, Wai Lam, Qian Yu, Anthony Man-Cho So, Shengding Hu, Zhiyuan Liu, and Nigel Collier. 2023. Decoder-only or encoder-decoder? interpreting language model as a regularized encoder-decoder.

Zihao Fu, Bei Shi, Wai Lam, Lidong Bing, and Zhiyuan Liu. 2020. Partially-aligned data-to-text generation with distant supervision. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9183–9193, Online. Association for Computational Linguistics.

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

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

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Chinenye Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Andre Niyongabo Rubungo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021), pages 96–120, Online. Association for Computational Linguistics.

Sebastian Gehrmann, Falcon Dai, Henry Elder, and Alexander Rush. 2018. End-to-end content and plan selection for data-to-text generation. In Proceedings of the 11th International Conference on Natural Language Generation, pages 46–56, Tilburg University, The Netherlands. Association for Computational Linguistics.

Amirata Ghorbani and James Zou. 2020. Neuron shapley: Discovering the responsible neurons.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery &amp Data Mining. ACM.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M. Khapra, and Balaraman Ravindran. 2023. A survey of adversarial defenses and robustness in nlp. ACM Comput. Surv., 55(14s).

Jing Gu, Mostafa Mirshekari, Zhou Yu, and Aaron Sisto. 2021. ChainCQG: Flow-aware conversational question generation. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume, pages 2061–2070, Online. Association for Computational Linguistics.

Qipeng Guo, Zhijing Jin, Xipeng Qiu, Weinan Zhang, David Wipf, and Zheng Zhang. 2020. CycleGT: Unsupervised graph-to-text and text-to-graph generation via cycle training. In Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 77–88, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Sangdo Han, Jeesoo Bang, Seonghan Ryu, and Gary Geunbae Lee. 2015. Exploiting knowledge base to generate responses for natural language dialog listening agents. In Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 129–133, Prague, Czech Republic. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In International Conference on Learning Representations.

David M. Howcroft, Anya Belz, Miruna-Adriana Clinciu, Dimitra Gkatzia, Sadid A. Hasan, Saad Mahamood, Simon Mille, Emiel van Miltenburg, Sashank Santhanam, and Verena Rieser. 2020. Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions. In Proceedings of the 13th International Conference on Natural Language Generation, pages 169–182, Dublin, Ireland. Association for Computational Linguistics.

Lishan Huang, Zheng Ye, Jinghui Qin, Liang Lin, and Xiaodan Liang. 2020a. GRADE: Automatic graph-enhanced coherence metric for evaluating open-domain dialogue systems. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9230–9240, Online. Association for Computational Linguistics.

Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020b. Challenges in building intelligent open-domain dialog systems.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. 2020. Pretraining with contrastive sentence objectives improves discourse performance of language models. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4859–4870, Online. Association for Computational Linguistics.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4198–4205, Online. Association for Computational Linguistics.

Kishlay Jha, Yaqing Wang, Guangxu Xun, and Aidong Zhang. 2018. Interpretable word embeddings for medical domain. In 2018 IEEE International Conference on Data Mining (ICDM), pages 1061–1066.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12) :1–38.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. arXiv preprint arXiv :2202.03629.

Zhijing Jin, Qipeng Guo, Xipeng Qiu, and Zheng Zhang. 2020. GenWiki: A dataset of 1.3 million content-sharing text and graphs for unsupervised graph-to-text generation. In Proceedings of the 28th International Conference on Computational Linguistics, pages 2398–2409, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. Transactions of the Association for Computational Linguistics, 5 :339–351.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics, 8 :64–77.

Mihir Kale and Abhinav Rastogi. 2020. Text-to-text pre-training for data-to-text tasks. In Proceedings of the 13th International Conference on Natural Language Generation, pages 97–102, Dublin, Ireland. Association for Computational Linguistics.

Hassan Kane, Muhammed Yusuf Kocyigit, Ali Abdalla, Pelkins Ajanoh, and Mohamed Coulibali. 2020. NUBIA: NeUral based interchangeability assessor for text generation. In Proceedings of the 1st Workshop on Evaluating NLG Evaluation, pages 28–37, Online (Dublin, Ireland). Association for Computational Linguistics.

Enkelejda Kasneci, Kathrin Sessler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, Stephan Krusche, Gitta Kutyniok, Tilman Michaeli, Claudia Nerdel, Jürgen Pfeffer, Oleksandra Poquet, Michael Sailer, Albrecht Schmidt, Tina Seidel, Matthias Stadler, Jochen Weller, Jochen Kuhn, and Gjergji Kasneci. 2023. Chatgpt for good? on opportunities and challenges of large language models for education. Learning and Individual Differences, 103 :102274.

Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2021. Discourse probing of pretrained language models. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, pages 3849–3864, Online. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In ICLR. OpenReview.net.

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

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, S. Auer, and Christian Bizer. 2015. Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web, 6 :167–195.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880, Online. Association for Computational Linguistics.

Wei Li, Wenhao Wu, Moye Chen, Jiachen Liu, Xinyan Xiao, and Hua Wu. 2022. Faithfulness in natural language generation: A systematic survey of analysis, evaluation and optimization methods.

Percy Liang, Michael Jordan, and Dan Klein. 2009. Learning semantic correspondences with less supervision. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, pages 91–99, Suntec, Singapore. Association for Computational Linguistics.

Python library. Dateparser.

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

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. CoRR, abs/1405.0312.

Yi Liu. 2021. A corpus-based lexical semantic study of Mandarin verbs of zhidao and liaojie. In Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation, pages 646–651, Shanghai, China. Association for Computational Lingustics.

Zhibin Liu, Zheng-Yu Niu, Hua Wu, and Haifeng Wang. 2019. Knowledge aware conversation generation with explainable reasoning over augmented graphs. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1782–1792, Hong Kong, China. Association for Computational Linguistics.

Chi-kiu Lo and Dekai Wu. 2011. MEANT: An inexpensive, high-accuracy, semi-automatic metric for evaluating translation utility based on semantic roles. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies, pages 220–229, Portland, Oregon, USA. Association for Computational Linguistics.

Ryan Lowe, Michael Noseworthy, Iulian Vlad Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. Towards an automatic Turing test: Learning to evaluate dialogue responses. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 1116–1126, Vancouver, Canada. Association for Computational Linguistics.

Scott M Lundberg and Su-In Lee. 2017a. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 4765–4774. Curran Associates, Inc.

Scott M Lundberg and Su-In Lee. 2017b. A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.

Hongyin Luo, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2015. Online learning of interpretable word embeddings. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1687–1692, Lisbon, Portugal. Association for Computational Linguistics.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2022. Post-hoc interpretability for neural nlp: A survey. ACM Comput. Surv., 55(8).

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1906–1919, Online. Association for Computational Linguistics.

R. Thomas McCoy, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. 2023. How much do language models copy from their training data? evaluating linguistic novelty in text generation using RAVEN. Transactions of the Association for Computational Linguistics, 11 :652–670.

Christoph Molnar. 2022. Interpretable Machine Learning, 2 edition.

Sebastien Montella, Betty Fabre, Tanguy Urvoy, Johannes Heinecke, and Lina Rojas-Barahona. 2020. Denoising pre-training and data augmentation strategies for enhanced RDF verbalization with transformers. In Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 89–99, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Edoardo Mosca, Ferenc Szigeti, Stella Tragianni, Daniel Gallagher, and Georg Groh. 2022. SHAP-based explanation methods: A review for NLP interpretability. In Proceedings of the 29th International Conference on Computational Linguistics, pages 4593–4603, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Shashi Narayan, Gonçalo Simões, Yao Zhao, Joshua Maynez, Dipanjan Das, Michael Collins, and Mirella Lapata. 2022. A well-composed text is half done! composition sampling for diverse conditional generation. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers), pages 1319–1339, Dublin, Ireland. Association for Computational Linguistics.

Shashi Narayan, Yao Zhao, Joshua Maynez, Gonçalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. Transactions of the Association for Computational Linguistics, 9 :1475–1492.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics : HLT-NAACL 2004, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017a. Why we need new evaluation metrics for NLG. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.

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

OpenAI. Introducing chatgpt. https://openai.com/blog/chatgpt. Accessed : 2023-08-24.

OpenAI. 2023. Gpt-4 technical report.

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

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1173–1186, Online. Association for Computational Linguistics.

Sungjoon Park, JinYeong Bak, and Alice Oh. 2017. Rotated word vector representations and their interpretability. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 401–411, Copenhagen, Denmark. Association for Computational Linguistics.

Laura Perez-Beltrachini and Claire Gardent. 2017. Analysing data-to-text generation benchmarks. In Proceedings of the 10th International Conference on Natural Language Generation, pages 238–242, Santiago de Compostela, Spain. Association for Computational Linguistics.

Laura Perez-Beltrachini, Parag Jain, Emilio Monti, and Mirella Lapata. 2023. Semantic parsing for conversational question answering over knowledge graphs. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2507–2522, Dubrovnik, Croatia. Association for Computational Linguistics.

Laura Perez-Beltrachini and Mirella Lapata. 2018. Bootstrapping generators from noisy data. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers), pages 1516–1527, New Orleans, Louisiana. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In Proceedings of the Tenth Workshop on Statistical Machine Translation, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics : System Demonstrations, pages 101–108, Online. Association for Computational Linguistics.

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

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

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv :1910.10683.

Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. 2021. The curious case of hallucinations in neural machine translation. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, pages 1172–1183, Online. Association for Computational Linguistics.

Clément Rebuffel, Marco Roberti, Laure Soulier, Geoffrey Scoutheeten, Rossella Cancelliere, and Patrick Gallinari. 2021a. Controlling hallucinations at word level in data-to-text generation. CoRR, abs/2102.02810.

Clement Rebuffel, Thomas Scialom, Laure Soulier, Benjamin Piwowarski, Sylvain Lamprier, Jacopo Staiano, Geoffrey Scoutheeten, and Patrick Gallinari. 2021b. Data-QuestEval: A referenceless metric for data-to-text semantic evaluation. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 8029–8036, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lena Reed, Shereen Oraby, and Marilyn Walker. 2018. Can neural generators for dialogue learn sentence planning and discourse structuring? In Proceedings of the 11th International Conference on Natural Language Generation, pages 284–295, Tilburg University, The Netherlands. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Ehud Reiter. 2017. How to do an nlg evaluation : Metrics. https://ehudreiter.com/2017/05/03/metrics-nlg-evaluation/.

Leonardo F. R. Ribeiro, Claire Gardent, and Iryna Gurevych. 2019. Enhancing AMR-to-text generation with dual graph representations. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3183–3194, Hong Kong, China. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Yue Zhang, Claire Gardent, and Iryna Gurevych. 2020. Modeling global and local node contexts for text generation from knowledge graphs. Transactions of the Association for Computational Linguistics, 8 :589–604.

Marco Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should I trust you?": Explaining the predictions of any classifier. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Demonstrations, pages 97–101, San Diego, California. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. Transactions of the Association for Computational Linguistics, 8 :842–866.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4035–4045, Brussels, Belgium. Association for Computational Linguistics.

Sascha Rothe and Hinrich Schütze. 2016. Word embedding calculus in meaningful ultradense subspaces. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers), pages 512–517, Berlin, Germany. Association for Computational Linguistics.

Amrita Saha, Vardaan Pahuja, Mitesh M. Khapra, Karthik Sankaranarayanan, and Sarath Chandar. 2018. Complex sequential question answering : Towards learning to converse over linked question answer pairs with a knowledge graph. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ananya B. Sai, Akash Kumar Mohankumar, and Mitesh M. Khapra. 2022. A survey of evaluation metrics used for nlg systems. ACM Comput. Surv., 55(2).

Gabriele Sarti, Nils Feldhus, Ludwig Sickert, and Oskar van der Wal. 2023. Inseq: An interpretability toolkit for sequence generation models. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3 : System Demonstrations), pages 421–435, Toronto, Canada. Association for Computational Linguistics.

Guus Schreiber and Yves Raimond. 2014. RDF 1.1 primer. W3C note, W3C. Https ://www.w3.org/TR/2014/NOTE-rdf11-primer-20140624/.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. QuestEval: Summarization asks for fact-based evaluation. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7881–7892, Online. Association for Computational Linguistics.

Lloyd S Shapley et al. 1953. A value for n-person games.

Lei Shen, Fandong Meng, Jinchao Zhang, Yang Feng, and Jie Zhou. 2021. GTM: A generative triple-wise model for conversational question generation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers), pages 3495–3506, Online. Association for Computational Linguistics.

Anastasia Shimorina, Claire Gardent, Shashi Narayan, and Laura Perez-Beltrachini. 2018. WebNLG Challenge: Human Evaluation Results. Technical report, Loria & Inria Grand Est.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In Findings of the Association for Computational Linguistics : EMNLP 2021, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In In Proceedings of Association for Machine Translation in the Americas, pages 223–231.

Marco Antonio Sobrevilla Cabezudo and Thiago A. S. Pardo. 2020. NILC at WebNLG+: Pretrained sequence-to-sequence models on RDF-to-text generation. In Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 131–136, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2017. Spine: Sparse interpretable neural embeddings.

Elior Sulem, Omri Abend, and Ari Rappoport. 2020. Semantic structural decomposition for neural machine translation. In Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics, pages 50–57, Barcelona, Spain (Online). Association for Computational Linguistics.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4231–4242, Brussels, Belgium. Association for Computational Linguistics.

Barkavi Sundararajan, Somayajulu Sripada, and Ehud Reiter. 2022. Error analysis of ToTTo table-to-text neural NLG models. In Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM), pages 456–470, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2017. Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. 2023. Ul2: Unifying language learning paradigms.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations.

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.

Craig Thomson and Ehud Reiter. 2021. Generation challenges: Results of the accuracy evaluation shared task. In Proceedings of the 14th International Conference on Natural Language Generation, pages 240–248, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Craig Thomson, Ehud Reiter, and Somayajulu Sripada. 2020. SportSett:basketball - a robust and maintainable data-set for natural language generation. In Proceedings of the Workshop on Intelligent Information Processing and Natural Language Generation, pages 32–40, Santiago de Compostela, Spain. Association for Computational Lingustics.

Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P. Parikh. 2019. Sticking to the facts: Confident decoding for faithful data-to-text generation. CoRR, abs/1910.08684.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Rob van der Goot, Nikola Ljubešić, Ian Matroos, Malvina Nissim, and Barbara Plank. 2018. Bleaching text: Abstract features for cross-lingual gender prediction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers), pages 383–389, Melbourne, Australia. Association for Computational Linguistics.

Chris van der Lee, Chris Emmery, Sander Wubben, and Emiel Krahmer. 2020. The CACAPO dataset: A multilingual, multi-domain dataset for neural pipeline and end-to-end data-to-text generation. In Proceedings of the 13th International Conference on Natural Language Generation, pages 68–79, Dublin, Ireland. Association for Computational Linguistics.

Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P. de Vries. 2020. Rel : An entity linker standing on the shoulders of giants. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20. ACM.

Emiel van Miltenburg, Miruna Clinciu, Ondřej Dušek, Dimitra Gkatzia, Stephanie Inglis, Leo Leppänen, Saad Mahamood, Emma Manning, Stephanie Schoch, Craig Thomson, and Luou Wen. 2021. Underreporting of errors in NLG output, and what to do about it. In Proceedings of the 14th International Conference on Natural Language Generation, pages 140–153, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Emiel van Miltenburg, Chris van der Lee, Thiago Castro-Ferreira, and Emiel Krahmer. 2020. Evaluation rules! on the use of grammars and rule-based systems for NLG evaluation. In Proceedings of the 1st Workshop on Evaluating NLG Evaluation, pages 17–27, Online (Dublin, Ireland). Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. Commun. ACM, 57(10) :78–85.

David Wan, Mengwen Liu, Kathleen McKeown, Markus Dreyer, and Mohit Bansal. 2023. Faithfulness-aware decoding strategies for abstractive summarization. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2864–2880, Dubrovnik, Croatia. Association for Computational Linguistics.

Juncheng Wan, Jian Yang, Shuming Ma, Dongdong Zhang, Weinan Zhang, Yong Yu, and Zhoujun Li. 2022. PAEG: Phrase-level adversarial example generation for neural machine translation. In Proceedings of the 29th International Conference on Computational Linguistics, pages 5085–5097, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jiongxiao Wang, Zichen Liu, Keun Hee Park, Muhao Chen, and Chaowei Xiao. 2023. Adversarial demonstration attacks on large language models.

Peng Wang, Junyang Lin, An Yang, Chang Zhou, Yichang Zhang, Jingren Zhou, and Hongxia Yang. 2021. Sketch and refine: Towards faithful and informative table-to-text generation. In Findings of the Association for Computational Linguistics : ACL-IJCNLP 2021, pages 4831–4843, Online. Association for Computational Linguistics.

Marc Wick. 2015. Geonames ontology.

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

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer

Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, pages 483–498, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet : Generalized Autoregressive Pretraining for Language Understanding. Curran Associates Inc., Red Hook, NY, USA.

Zixiaofan Yang, Arash Einolghozati, Hakan Inan, Keith Diedrick, Angela Fan, Pinar Donmez, and Sonal Gupta. 2020. Improving text-to-text pre-trained models for the graph-to-text task. In Proceedings of the

3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+), pages 107–116, Dublin, Ireland (Virtual). Association for Computational Linguistics.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED : A large-scale document-level relation extraction dataset. In Proceedings of ACL 2019.

Zheng Ye, Liucun Lu, Lishan Huang, Liang Lin, and Xiaodan Liang. 2021. Towards quantifiable dialogue coherence evaluation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers), pages 2718–2729, Online. Association for Computational Linguistics.

Steve Young, Milica Gašić, Simon Keizer, François Mairesse, Jost Schatzmann, Blaise Thomson, and Kai Yu. 2010. The hidden information state model: A practical framework for pomdp-based spoken dialogue management. Computer Speech Language, 24(2) :150–174.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020a. Bertscore: Evaluating text generation with bert. In International Conference on Learning Representations.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with bert. In International Conference on Learning Representations.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics : System Demonstrations, pages 270–278, Online. Association for Computational Linguistics.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hao Zhou, Minlie Huang, Yong Liu, Wei Chen, and Xiaoyan Zhu. 2021. EARL: Informative knowledge-grounded conversation generation with entity-agnostic representation learning. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 2383–2395, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation with graph attention. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, page 4623–4629. AAAI Press.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In Proceedings of the 20th Chinese National Conference on Computational Linguistics, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

Julia El Zini and Mariette Awad. 2022. On the explainability of natural language processing deep models. ACM Comput. Surv., 55(5).