MASSIVE Multilingual Abstract Meaning Representation: A Dataset and Baselines for Hallucination Detection

Michael Regan
University of Washington
Allen School
mregan@cs.washington.edu

Shira Wein
Georgetown University
Dept. of Computer Science

Emilio Monti
Amazon
Cambridge, UK

Abstract

Abstract Meaning Representation (AMR) is a semantic formalism that captures the core meaning of a text. There has been substantial work developing AMR corpora in English and more recently across languages, though the limited size of existing datasets and the cost of collecting more annotations are prohibitive to leveraging such structured data in engineering applications. With both engineering and scientific questions in mind, we introduce MASSIVE-AMR, a dataset with more than 84,000 *manually annotated* items, currently the largest and most diverse of its kind: AMR graphs for 1,685 information-seeking utterances mapped to 50+ typologically diverse languages. We detail unique characteristics of the dataset and, building on work in generative AI, report on experiments using large language models for multilingual AMR and SPARQL parsing, as well as experiments utilizing AMRs for hallucination detection in the context of knowledge base question answering (KBQA), with results shedding light on many of the challenges of multilingual question answering.

1 Introduction

Knowledge base question answering (KBQA) has a long history in natural language processing, with the task of retrieving an answer from a knowledge base (KB) such as Wikidata or DBPedia (Lehmann et al., 2015) integral to many large-scale question answering systems (Kapanipathi et al., 2021). In KBQA, a natural language question is converted into a structured query language such as SPARQL, an executable semantic parse. However, data to train models is expensive, few multilingual resources are available, and performance is limited for long-tail queries when generative models are prone to hallucinate relations, a problem compounded by arbitrary variability in form-meaning mappings across languages (Croft, 2002).

Most notably, research in multilingual KBQA is hindered by lack of data (Usbeck et al., 2018; Cui et al., 2022; Perevalov et al., 2022). To address this, we think big, creating a dataset 20 times larger and with 5-6 times more languages than currently available (Table 1). To reach this scale, we select 1685 QA utterances with existing manual translations from the MASSIVE corpus[1] (FitzGerald et al., 2022) and create Abstract Meaning Representation (AMR) graphs (Banarescu et al., 2013), amounting to 84,000 gold text-to-graph annotations, hopefully a significant boon to AMR and KBQA research.

	AMR3.0	QALD9-AMR	MASSIVE-AMR
# of languages	1	9+	52
# utterances (utt)	59K	508	1685
# utts-to-graphs	59K	5K	84K
mean tokens/utt	15.9	EN: 7.5	EN: 8.2
AMR nodes	EN	EN	EN+local

Table 1: Existing AMR corpora compared with our dataset, MASSIVE-AMR. QALD9-AMR and MASSIVE-AMR target multilingual QA utterances.

Reaching this number of text-to-graph mappings is possible making use of entity annotations in MASSIVE (Table 2). Multilingual graphs, in addition to the long-tail nature of utterances selected for MASSIVE-AMR, add to the challenges of our multilingual QA dataset, details about which we document (Section 3).

To explore the increased challenge and utility of MASSIVE-AMR, we examine how we can leverage AMRs to gauge a model's confidence in SPARQL query generation (Section 4), reporting on experiments in structured generation and SPARQL relation hallucination detection using large language models (LLMs) (Section 5).

[1] https://github.com/alexa/massive

	Utterance	AMR
MASSIVE-AMR	when was obama born	(b / bear-02 :ARG1 (o / "obama") :time (a / amr-unknown))
	quand est né sarkozy	(b / bear-02 :ARG1 (s / "sarkozy") :time (a / amr-unknown))
	+50 langs.	+50 local AMRs
QALD9-AMR	Who developed Skype? Qui a développé Skype?	(d / develop-02 :ARG0 (a / amr-unknown) :ARG1 (s / "Skype"))
	+7-8 langs.	Same AMR, all languages

Table 2: Compared with existing multilingual AMR datasets, MASSIVE-AMR has local entities (English 'obama', French 'sarkozy') and covers >5x more languages. AMRs simplified to fit table.

Our research contributions thus include:

- Creation of the largest-scale multilingual AMR question corpus to date;
- Evaluation of LLMs on generation of SPARQL and AMRs structures across languages; and
- Design, development, and evaluation of generative models leveraging AMRs for SPARQL relation hallucination detection.

We will publicly release the MASSIVE-AMR training and validation data upon publication.

2 Related Work

In this section, we discuss important related work in question answering, KBQA, the AMR formalism, applications of AMRs for KBQA, as well as hallucination detection, structure transfer, and multilingual QA resources.

2.1 Question Answering

Question answering (QA) is the task of retrieving or predicting an answer to a natural language query given a document, a collection of documents, a list of answers, a list of knowledge triples, or with a generative model. The broad task of QA encompasses research in Information Retrieval (Lewis et al., 2020), Machine Reading Comprehension (MRC) (Das et al., 2018), and Open-Domain Question Answering (Lewis et al., 2021; Zhang et al., 2023), with the general objective of identifying documents or passages in large, raw text corpora relevant to a given question. As studies of model confidence in answers, important work done in calibration of QA systems (Jiang et al., 2021; Kadavath et al., 2022) has aims similar to our own.

2.2 KBQA

Knowledge base question answering (KBQA) is the task of retrieving answers from a knowledge base given a natural language question. The challenges in retrieving and making sense of textual information are fundamentally different from the primary challenge of KBQA: generating semantically accurate KB queries.

Various approaches to KBQA have been proposed over the decades, including converting queries to logical forms, semantic parses, and decomposing complex questions (Zelle and Mooney, 1996; Zettlemoyer and Collins, 2005; Talmor and Berant, 2018). Scalable KBQA systems utilize structured representations (SPARQL) to query a KB (e.g., DBPedia[2]), a collection of triples of form <subject, rel_j, object>, with rel_j a semantic relation from ontology $\mathcal{R}$ (of various sizes, e.g., $|\mathcal{R}_{\text{DBPedia}}| > 2500$). Baselines for SPARQL generation are available (Banerjee et al., 2022), with a central challenge being how to identify generated queries not covered by a given $\mathcal{R}$, cases where models tend to hallucinate relations.

In the age of large language models, querying manually-curated knowledge bases provides numerous advantages such as: (1) factuality guarantees, (2) the ability to update information in real time, and (3) risk mitigation for users, reducing exposure to sensitive or toxic content. With these motivations in mind, we turn our attention to AMRs.

2.3 Abstract Meaning Representation

Abstract Meaning Representation (AMR) (Banarescu et al., 2013) is a linguistic formalism that represents the meaning of utterances as directed, mostly acyclic graphs. Graph nodes denote key concepts associated with an utterance, primarily event and event participant semantics. Nodes in turn are connected by labeled edges for event-event, event-entity, and entity-entity relations.

Early AMR research targeted parsing, with the JAMR parser (Flanigan et al., 2014) paving the way for state-of-the-art models based on transitions (Drozdov et al., 2022), seq2seq approaches (Bevilacqua et al., 2021), and ensemble distillation

[2]Open, curated, cross-domain, and up-to-date: `https://www.dbpedia.org/`

(Lee et al., 2022). In lieu of such heavily engineered approaches, we target generative models with in-context learning and fine-tuning following recent work in this area (Ettinger et al., 2023).

The original AMR reference-based metric is Smatch (Cai and Knight, 2013), a measure of overlapping triples, which has led to the newly optimized Smatch++ (Opitz, 2023) and S2match (Opitz et al., 2020) which uses embeddings to match concepts within triples. Wein and Schneider (2022) released multilingual AMR metrics such as XS2match using LaBSE embeddings (Feng et al., 2022) for cross-lingual AMR evaluation.

2.4 AMR for KBQA

Correspondences between symbolic meaning representations and SPARQL queries can be leveraged for knowledge base question answering (KBQA) (Kapanipathi et al., 2021; Bornea et al., 2022). Observed patterns of AMR nodes and SPARQL concepts and variables support a deterministic mapping (Kapanipathi et al., 2021) and sequence-to-sequence models can apply such rules selectively for improved generalization (Bornea et al., 2022). Other work shows similar means of using symbolic representations for QA (Niu et al., 2023; Wang et al., 2023).

2.5 Hallucination detection

Hallucinations, the generation of flawed or incongruous natural language assertions, represent a persistent problem with LLMs (Ji et al., 2023). Much research in hallucination detection has targeted text-to-text settings, checking the factuality of summarized texts (Gabriel et al., 2021) or proposing mitigation strategies such as retrieval methods to make generations attributable (Aksitov et al., 2023; Rashkin et al., 2023). In contrast, we target text-to-graph generation, namely executable semantic parse generation, experimenting with AMRs to detect *easy* and *hard* cases of hallucination, ranking generations in a joint space as we will detail in Section 4.

2.6 Structure transfer

Leveraging knowledge about one structure to help model another is common across fields. In network science, graph topology correspondences in circuits, food networks, and gene replication are studied as *motifs* (Milo et al., 2002). Early work in semantic parsing explored integrating syntactic information (Gildea and Jurafsky, 2002), and resources such as FrameNet (Johnson and Fillmore, 2000) and VerbNet (Schuler, 2006) provide mappings between syntactic to semantic structures. Transfer learning, the training of models for multiple tasks, works for semantic parsing (Fan et al., 2017), while AMRs improve Combinatory Categorial Grammar parsing (Artzi et al., 2015) and information extraction (Zhang and Ji, 2021). In research on training LMs, syntactic and semantic representations may be helpful as inductive biases (Alberti et al., 2022; Prange et al., 2022).

2.7 Multilingual QA resources

To support research for multilingual dialogue systems, MASSIVE (FitzGerald et al., 2022) is a collection of 20K utterances with manual translations into 50+ typologically diverse languages[3]. For our MASSIVE-AMR dataset, we select all utterances from the MASSIVE QA slice, and add AMR annotations, as we describe in Section 3.

The closest multilingual QA resource to MASSIVE-AMR is the QALD9-AMR treebank (Lee et al., 2022) which maps AMR graphs to questions in nine languages and SPARQL queries (Usbeck et al., 2018). QALD9 consists of short factoid questions, in comparison with MASSIVE-AMR which has factoid and multi-step reasoning questions about currency, stocks, and arithmetic, mapped to English AMRs with local entities like in Table 2.

AMRs were not designed to function across languages (Banarescu et al., 2013), and while language has a measurable effect on AMR graph structure (Wein et al., 2022), efforts have been made to effectively represent meaning of non-English sentences in AMRs (Xue et al., 2014; Hajič et al., 2014). Work grounding representations in linguistic typology such as Uniform Meaning Representation (UMR) (Van Gysel et al., 2021) better accounts for formal and semantic differences across languages, and work tying large multilingual resources to a common linguistic formalism (Navigli et al., 2022) is also promising.

3 Data: Corpus Creation

To create a corpus of multilingual AMR graphs, we started with an existing dataset of QA utterances, tailored AMR 3.0 guidelines to our use case, trained a team of professional annotators to create AMRs for English utterances, and then made

[3] 52 languages in v1.1

automatic mappings to multilingual utterances using existing entity mention spans, a process which from start to finish took three months. In this section, we report details about the data we started with, guidelines, and annotation agreement scores.

Acquiring scaleable multilingual data. We target a variety of QA utterances and thus select 1685 English examples from MASSIVE (FitzGerald et al., 2022) including entity annotations like in the multilingual examples in Table 3.

Lang.	Example utterance
en-US	what is the population of [place_name: new york]
sl-SL	koliko prebivalcev ima [place_name: ljubljana]
it-IT	qual è la popolazione di [place_name: roma]
sq-AL	cila është popullësia e [place_name: tiranës]
cy-GB	beth yw poblogaeth [place_name: efrog newydd]
am-ET	Abyssinica SIL[place_name: á áá ááá] ááá ááá ááá áá
af-ZA	wat is die bevolking van [place_name: kaapstad]
is-IS	hver er íbúafjöldi [place_name: reykjavíkur]
az-AZ	[place_name : sumqayıtın] halisi n qdrdir

Table 3: Examples from MASSIVE asking about the population of a local city: the English utterance asks about New York, the Slovenian (sl-SL) about Ljubljana, etc.

Long-tail QA. Many utterances in MASSIVE are described as long-tail, that is, associated with low user feedback in interactions with a digital assistant. In some cases, it is clear what increases friction (an incomplete utterance, or a speech-to-text error). Examining translations of English utterances provides insight (Appendix A.2).

Entity localization. In comparable datasets (Cui et al., 2022; Perevalov et al., 2022), entities are shared across languages (e.g., English *Where did Abraham Lincoln die?* corresponds to German *Wo starb Abraham Lincoln?*). To help with challenges of large-scale QA systems, MASSIVE entities are local, e.g. German questions target German entities (*wo starb otto von bismarck*[4]).

Comparing AMR corpora. AMR datasets differ in composition: AMR 3.0 (Banarescu et al., 2013) is based on news and other written discourse and consists of relatively few factoid or information-seeking questions (less than 10%)[5]. In contrast, MASSIVE-AMR includes requests about currency conversions, quantities, comparative and superlatives, and simple arithmetic. For more details about how the corpora compare, see Appendix A, Table 11.

Annotation principles: Canonical forms. In keeping with original AMR guidelines[6], an AMR captures meaning, not form. We hence prefer canonical forms for utterances like currency conversion and arithmetic: e.g., 'how much is the euro versus the dollar' and 'what is the euro worth compared to the dollar' map to similar graphs. Likewise, arithmetic questions are associated with top node 'equal-01' even without token 'equal' present ('how much is two plus two' and 'sum of two and two' treated like 'what does two and two equal').

Question-imperative continuum. It proved difficult to reach agreement for annotations of question versus imperative forms. In English, 'could you tell me the price of google', 'what is the price of google', and 'tell me the price of google' share the same meaning. However, treating the imperative (e.g., an embedded question 'tell me what the price is') as a question is out-of-line with AMR 3.0. The guideline we adopt is to preserve imperative form and treat polite questions (e.g., English 'could you tell me the price') the same as the base question forms (e.g., 'what is the price').

Annotation agreement scores. 4-5 trained annotators created AMRs for 1685 utterances, examining differences in batches of 200 weekly, with inter-annotator agreement ranging from 78-82% Smatch, comparable to reported agreement for AMR experts (Banarescu et al., 2013). We note that MASSIVE-AMR consists of many similar questions and simple utterances, with on average 50% less length compared to AMR 3.0 (Table 1). We select the single best AMR in candidate sets and manually retrofit to increase consistency.

For non-English entities, we replace AMR node labels using MASSIVE annotations. We note that not all utterances have annotations, and that a lack of entity alignments adds noise since often word order matters (e.g., currency conversion). To improve data quality, we manually curate validation and test sets (25% of total).

4 SPARQL hallucination detection

Scaleable QA systems often utilize structured representations such as SPARQL for knowledge base retrieval, pairing a natural language utterance with an executable semantic query. The SPARQL in the Wikidata or DBPedia case is straightforward: we get a question in, the system produces an answer out. However, in practice we simply need a system capable of judging if a given answer is correct or

[4]MASSIVE utterances are uncased.

[5]Graphs with concepts amr-unknown or amr-choice

[6]https://github.com/amrisi/amr-guidelines/

not, which in the context of generative AI we study as *hallucination detection*.

Hallucinations. A problem in open-domain question-answering regards *hallucinations*, cases when effectively the target Ontology (in our case, DBPedia) does not have valid symbols for a given input question. For example, if the relation 'crimeRate' does not exist, a SPARQL generation model will stumble on a question like 'What is the crime rate in LA?' by generating a query with a non-existing relation, which we can verify with a set membership check. A harder case to detect is when the model predicts a relation that exists but which is imprecise, e.g., answering a question about crime rate with details about crime types. We would like to design and test methods for the detection of such hallucinations using LLMs.

An advantage of AMR is that its ontology is open: i.e. if a given concept is missing, we can practically lemmatize the English. Or more often, AMR tends to be more granular, and more complex meanings (that in an Ontology might be collapsed into a single symbol) are split into several constituents (i.e. 'crimeRate' might be a single symbol in an Ontology, but it is instead split into constituents by AMR). Hence, hallucinations are much less of a problem in AMR.

We hypothesize that if we train a single semantic parser to generate both SPARQL and AMR parses, simply mixing the training data, and generate multiple semantic parse candidates in a target `N`-best, the inclusion of AMR parses would allow us to detect SPARQL hallucinations. We could thus use a very high confidence AMR and very low confidence SPARQL as a signal that the given question is out-of-ontology, like we see in Table 4.

We examine dual subtasks of SPARQL generation: (1) How accurate are models at checking *set membership*, in our case, verifying generated relations are in a given relation set, or:

$$r_{pred} \overset{?}{\in} \mathcal{R}_{known}$$

and, (2) How good are models at flagging queries with possibly imprecise relations (e.g., predicting 'author' for 'Who created Iron Man?'), a task we call *hard hallucination detection*, described next.

5 Experiments

To gain insight into the hypothesis that AMRs may help detect hallucinations in generated SPARQL queries, we first report on experiments in semantic representation generation, a first-of-its-kind in a diverse multilingual setting. We next experimentally confirm that models do indeed hallucinate relations, before moving on to the target task of hallucination detection. For our experiments, we compare in-context learning and fine-tuned LLMs, training and evaluating on an existing corpus of questions with gold AMRs and SPARQL (QALD9) and sampled MASSIVE-AMR. We are guided by the following research questions:

1. How good are LLMs at generating AMRs and SPARQL queries across languages?
2. How prevalent are SPARQL relation hallucinations with generative models?
3. In a simulated setting, how well do models do detecting hallucinated SPARQL relations?
4. Can we use a joint AMR-SPARQL model to do better relation hallucination detection?

The standard approach to study the coverage of a set of relations is use all the data associated with a relation set $\mathcal{R}$ to train semantic parser $SP_{\mathcal{R}}$; we then remove all examples that contain relation r_j and train $SP_{\{\mathcal{R}-r_j\}}$, measuring how well the model does for queries likely to require r_j.

An advantage of training a joint AMR-SPARQL model from scratch is having complete control over the input relations; a disadvantage is that, in the case we use a LLM, we have no knowledge about what relations the model may have seen in pretraining. For our early experiments, we use LLMs trained on 1000s of examples without hard constraints on allowed relations[7].

In experiments, we measure how well a LLM verifies generated relations are members of a predefined set of relations, a task we refer to as *easy hallucination detection*. We also look at *hard hallucination detection*, when a model hallucinates a relation that exists but that is not quite right, that is, the needed relation for a query is potentially not covered by a given $\mathcal{R}$. For experiments, we compare in-context learning with fine-tuned LLMs.

5.1 In-context learning

For few-shot generation, we use GPT models (OpenAI, 2023) with prompts of length <2400 tokens

[7]This could be done in decoding, setting logits of all non-relation tokens to `-inf` after a colon, an unambiguous signal of a SPARQL relation.

Utterance	Rank #1	Rank #2	Prediction
Who created Iron Man?	SELECT DISTINCT ?uri WHERE { res:Iron_Man **dbo:creator** ?uri }	(c / create-01 :ARG0 (a / amr-unknown) :ARG1 (i / Iron_Man))	SPARQL query OK
Who created Iron Man?	(c / create-01 :ARG0 (a / amr-unknown) :ARG1 (i / Iron_Man))	SELECT DISTINCT ?uri WHERE { res:Iron_Man **dbo:author** ?uri }	Relation may be imprecise creator vs. author 'Hard' to detect
Crime rate in NYC?	(c / crime-02 :location (n / NYC) :frequency (r / rate-entity-91 :ARG1 (a / amr-unknown)))	SELECT ?crimeRate WHERE { res:NYC **dbo:crimeRate** ?crimeRate . }	Relation does not exist Hallucination: crimeRate 'Easy' to detect

Table 4: A joint AMR-SPARQL generation model can help detect faulty semantic relations (relations in **bold**) by leveraging a N-best list of candidates of mixed representation types. For 'Who created Iron Man?', when the relation '**creator**' exists and is precise (top), the model should rank its SPARQL query higher than an associated AMR. Or, the AMR ranks higher when a relation exists but is imprecise (middle). Models also generate non-existent relations (bottom), detected via ranking or a look-up operation.

Relations	Subset descriptions
All observed	$\mathcal{R}_{\text{obs}}$
In-context	$\mathcal{R}_{\text{context}} \subset \mathcal{R}_{\text{obs}}$
Subsets similar	$\{\mathcal{R}_1^{\text{sim}}, \ldots, \mathcal{R}_j^{\text{sim}}\}$, $\mathcal{R}_i^{\text{sim}} \subset \mathcal{R}_{\text{obs}}$
Controlled	$r_{\text{cntl}} \in \mathcal{R}_i^{\text{sim}}, \notin \mathcal{R}_{\text{context}}$
Ground truth	$\{r_m, .., r_{\text{cntl}}, .., r_n\} \subset \mathcal{R}_{\text{obs}}$

Table 5: To test how well a generation model adheres to following instructions for allowed relations, we leave similar relations out as a control.

(see Appendix C) composed employing strategies we describe in this section.

Strategy #1: Constrain and verify relations. We give a list of allowed SPARQL relations and which the model uses to verify each predicted relation. We include eight (8) examples of joint AMR-SPARQL predictions in multiple languages[8].

Strategy #2: Simulate missing relations. To control for relations (Table 5), we count DBPedia relations in QALD9-AMR training data, select the 140 more frequent relations[9], and set aside 1+ relations for utterances in prompt where the model should prefer AMR over SPARQL, ensuring examples abide by constraints.

To test our *hard hallucination detection* hypothesis, we determine DBPedia relations to control for by manually grouping similar relations (e.g., 'creator,' 'writer,' and 'developer' are similar; Table 5, row 3) and select questions associated with any of these relations. We compare predictions allowing all relations versus the allowed list less the controlled relation (Table 5, row 4).

Strategy #3: Simulate ranking. We would like the model to rank without access to ground truth confidence scores, so we assign random confidence scores using a Dirichlet distribution (K=3), dropping the minimum value[10]. SPARQL with only allowed relations rank higher than AMRs.

Strategy #4: Demonstrate hallucination detection. Examples (Appendix C) show easy and hard hallucination detection, and we direct the model's attention to AMRs ranked higher[11].

5.2 Additional controls

We include results with an oracle, in which we direct the model's attention to the disallowed relation, providing an upper bound on achievable performance and giving insight into analysis. For consistency across datasets, we normalize all utterances (lower case, no punctuation).

5.3 Fine-tuning

We fine-tune a joint AMR-SPARQL model using various publicly available LLMs: a knowledge distilled variant of GPT-2-XL (West et al., 2022) and LLaMA-13B (Touvron et al., 2023); for model details, see Appendix B. For challenging test data, we use same-sized samples (900) from QALD9 and MASSIVE-AMR of the same languages[12].

[8] In our experiments, English and Spanish

[9] Observed >1 times, about 50% of data

[10] The minimum value represents the probability density of bottom predictions in latent N-best ranking

[11] The instruction reads: "Rank AMRs higher when predicted SPARQL is likely wrong, like in exs 5 and 8."

[12] English, Spanish, German, French, Russian

5.4 Evaluation guidelines

For AMR generation, we report Smatch (Cai and Knight, 2013), and for SPARQL we check query executability[13] and verify if answers exist in DBPedia. We do not evaluate answer factuality, as our objective is to measure model confidence in semantic parse correctness, not the model's knowledge of the contents of a given KB[14].

For hard hallucination detection experiments using in-context learning, we employ quantitative and qualitative means of analysis. For perturbed examples (i.e., generate a query for a question with a known disallowed relation), a predicted ranking is good if the model: (1) ranks the AMR higher, (2) ranks the SPARQL higher and simultaneously verifies the relation is not allowed, or (3) generates a valid alternative SPARQL[15]. For easy hallucination detection, we measure query executability and stratify results by dataset.

For fine-tuned joint AMR-SPARQL, with a diverse beam search (n=5) we examine top-ranked generated sequences, majority `N`-best, and transition scores for first tokens[16]. Our hypothesis is models will prefer SPARQL over AMR for QALD9 and vice versa for MASSIVE-AMR, since all QALD9 is matched with ground truth SPARQL.

For evaluation, models output a queryable object (JSON) with three key-value pairs (generated query, list of relations in query, and relation verification; see Appendix C), with very few structural errors observed (<1% in our studies).

5.5 Results

We present results on in-context learning for generation of AMR (Table 6) and SPARQL (Table 7) across languages, report on SPARQL hallucinations (Table 8), and then present results of fine-tuned joint AMR-SPARQL (Table 10).

5.6 Analysis and discussion

For **AMR generation** (Research question 1), results (Table 6, examples and error analysis in Appendix D) show that state-of-the-art AMR systems still outperform in-context learning with margins between 10-20%, a display of the strengths of engineered modular systems, data augmentation, and

	Model	Data	F1 ↑
Few-shot/EN	GPT-3.5	MASSIVE-EN	$0.43_{\pm 0.20}$
		QALD9-EN	$0.57_{\pm 0.17}$
	GPT-4	MASSIVE-EN	$0.53_{\pm 0.21}$
		QALD9-EN	$0.70_{\pm 0.16}$
Few-shot/non-EN	GPT-3.5	MASSIVE+	$0.33_{\pm 0.22}$
		MASSIVE-	$0.42_{\pm 0.20}$
		QALD9	$0.44_{\pm 0.20}$
	GPT-4	MASSIVE+	$0.46_{\pm 0.21}$
		MASSIVE-	$0.49_{\pm 0.20}$
		QALD9	$0.58_{\pm 0.22}$
SOTA	Struct-BART	QALD9-EN	0.90
		AMR 3.0	0.84

Table 6: AMR generation results, with F1 by model, dataset, and languages, with in-context learning (top two sections) and SOTA (Lee et al., 2022).

	Data	Exec. ↑	Returns ↑
GPT-3.5	MASSIVE+	0.93	0.32
	MASSIVE-	0.94	0.41
	QALD9	0.97	0.53
GPT-4	MASSIVE+	0.94	0.34
	MASSIVE-	0.99	0.50
	QALD9	1.00	0.52

Table 7: Few-shot SPARQL generation results across datasets and models. We report executability and how many return existing records.

AMR post-processing. Comparing few-shot models, GPT-4 outperforms GPT-3.5 by a margin of 10-13% F1, with performance on QALD9 14-17% F1 higher than MASSIVE-AMR, evidence of the challenge of the latter. Models perform 5-12% F1 higher for MASSIVE- compared to more diverse MASSIVE+[17], the first reported AMR results we are aware of for many of these languages.

SPARQL generation. Results of SPARQL generation with in-context learning (Table 7, examples in Appendix E) provide evidence that LLMs perform well in a few-shot setting, exceeding 90% F1 across datasets and languages. However, as LLMs are not trained on up-to-date data, no more than 52% of queries for QALD9 and 32% of MASSIVE-AMR return existing DBPedia records. Models display good performance for MASSIVE+, where AMR performance was observed to decrease, evidence that LLMs contain more knowledge about SPARQL over AMR structures.

SPARQL relation hallucination rates (Research question 2). In Table 8, we examine two

[13] Using Python SPARQLWrapper

[14] KBs change over time, many local entities do not have a DBPedia entry, etc.

[15] We check executability and evaluate manually

[16] Either 'AMR' or 'SPARQL' or first sub-token therein

[17] Details about MASSIVE- and non-overlapping MASSIVE+ language subsets in Appendix A.3

	Data	Perturb	#Utts	Halluc. ↓	Detects ↑
GPT-3.5	MASSIVE+	No	38	0.21	0.0
		Yes	62	**0.71**	0.04
	MASSIVE-	No	38	0.16	0.0
		Yes	62	**0.59**	0.0
	QALD9	No	110	0.22	0.09
		Yes	110	**0.84**	0.0
GPT-4	MASSIVE+	No	34	0.06	0.50
		Yes	66	**0.48**	0.09
	MASSIVE-	No	36	0.0	n/a
		Yes	64	**0.54**	0.14
	QALD9	No	50	0.04	0.0
		Yes	50	**0.46**	0.08

Table 8: SPARQL hallucination and hallucination detection rates with a non-joint model. When we perturb a relation, hallucination rates are high; for both models, detection rates are consistently poor.

Model	Oracle	#Perturb	Halluc. ↓	Detects ↑
GPT-3.5	no	60/120	0.58	0.07
GPT-4	no	60/120	0.39	0.17
GPT-4	yes	150/240	0.31	0.76

Table 9: Results of joint AMR-SPARQL detection with in-context learning (8-shot), targeting 140 SPARQL relations and 8 languages. We report rate of hallucination and hallucination detection.

	Langs.	Data	Top-1	Top-5 maj.	Transition
GPT-2-XL	EN	QALD9	0.50 ×	0.68 ✓	0.83 ✓
		MASSIVE-AMR	0.58	0.62	0.80
	Non-EN	QALD9	0.53 ×	0.55 ×	0.74 ✓
		MASSIVE-AMR	0.54	0.54	0.70
LLaMa	EN	QALD9	0.82 ✓	0.95 ×	0.90 ~
		MASSIVE-AMR	0.76	0.95	0.88
	Non-EN	QALD9	0.78 ×	0.95 ×	0.82 ×
		MASSIVE-AMR	0.88	0.98	0.95

Table 10: Proportion of SPARQL (vs AMR) predictions, fine-tuned models. We use features of N-best generation (top-1 prediction, top-5 majority, and first token transition score) to estimate confidence in a formalism. We hypothesize models will prefer SPARQL for QALD9 over MASSIVE-AMR; checkmarks (✓) indicate evidence in support.

questions: (1) do models hallucinate SPARQL relations when we remove some relations from an allowed list? and (2) can models also detect these hallucinations? We observe that models do hallucinate relations and yet fail at detection consistently. Specifically, we find that under normal, non-perturbed conditions across languages (odd rows of Table 8), GPT-3.5 exhibits hallucination rates between 16-22%, which GPT-4 reduces to 0-6%. As expected, when we perturb (disallow) a relation likely to be needed in the query (even rows), hallucination rates increase considerably: for GPT-3.5 between 40-60% and for GPT-4 between 42-54%.

Hallucination detection, non-joint model. With 2-shot SPARQL query generation, models show poor rates of hallucination detection, with GPT-4 detecting no more than 14% of all hallucinations. In a vast majority of cases (86-100%, gray column, Table 8), models are deceptive, incorrectly verifying disallowed relations (Ex. 2 in Appendix E), providing us with justification to see if we can do better with a joint model.

Hallucination detection, in-context joint model (Research question 3). Results with oracle in-context learning using GPT-4 show improved hallucination detection through dual strategies of: (1) Higher consistency in confirming semantic relations exist (i.e. the model is more honest when it circumvents instructions), and (2) ranking AMRs higher more consistently.

Looking at cases of nearly accurate relations (*hard hallucination detection*), GPT-4 mostly abides by constraints (e.g., generating 'author' instead of 'creator' for 'who created iron man'). However, it is difficult to assess correctness as much depends on the target KB and how we interpret the question[18]. Overall, GPT-4 employs dual hallucination detection strategies well: for 1 in 5 hallucinations, ranking AMRs higher, and, for 3 of 5, generating queries with disallowed relations that it accurate verifies as non-existent.

Hallucination detection, fine-tuned joint model (Research question 4). In contrast to the oracle model, results of fine-tuned joint models are inconclusive (Table 10). With GPT-2-XL$_{\text{distill}}$, preference between SPARQL vs AMR is mostly 50-50, with variation only with the first token transition score metric. Fine-tuned LLaMa, in contrast, shows a bias towards SPARQL under every condition (between 75-95%), though only in one setting (top-1 prediction) does the model prefer SPARQL consistently for QALD9 vs MASSIVE-AMR. Qualitative analysis reveals LLaMa prefers AMR for incomplete utterances such as 'describe' and 'calculate this', and it often misclassifies currency conversion utterances as having valid SPARQL[19]

With our study of fine-tuned models, we high-

[18]Resolving ambiguities such as whether the creator of Iron Man is 'Stan Lee' or 'Tony Stark'

[19]In principle, currency conversion values could be stored in a KB, but in practice KBs are not updated in real-time.

light that we can examine an N-best space from multiple perspectives (top-1 prediction, majority, transition scores), and future research may target how to improve hallucination detection methods with these measurements. We also note that the proportion of AMRs versus SPARQL in fine-tuning likely has an effect: in our experiments, we include more AMRs than SPARQL (Appendix B), an observation suggesting a study with more varied proportions may be warranted. Also, we used limited amounts of fine-tuning data (less than 6k examples), which we can increase in future work.

Overall, we find evidence that in-context learning for hallucination detection is quite challenging. With an oracle (Table 9), GPT-4 misreports 24% of cases of disallowed relations. Without an oracle, the rate of 'deception' exceeds 80%, which proved challenging to overcome despite multiple prompt variations, which included promised rewards for sticking to allowed relations, veiled (and unveiled) threats, repeated warnings, and legalese which bound the model to abide by restrictions, tactics the models consistently disregarded, leaving plenty of space for improvement in future work.

6 Conclusion

We present MASSIVE-AMR, the largest and most diverse dataset to date of multilingual questions paired with AMRs. We present results on independent tasks of AMR and SPARQL generation, the first such results for many languages. AMR generation with in-context learning is low compared with state-of-the-art; however, qualitative assessment reveals many coherent graphs despite low Smatch scores. Future research may look into more robust similarity metrics and how to gather more reference graphs for evaluation.

In a single use case for MASSIVE-AMR, we compared non-joint with joint AMR-SPARQL models for the task of relation hallucination detection. Results of in-context learning demonstrate that models indeed are prone to relation hallucinations, and that 'easy' hallucination detection is actually quite hard, even for GPT-4. Further, 'hard' hallucination detection appears to be only viable with an oracle specifically directing the model's attention to disallowed relations. We report progress in leveraging multiple features of an N-best generation space to assess model confidence in AMR vs SPARQL, and fine-tuned models, though imperfect, show promise with little fine-tuning data. Experimental results provide insight into hypotheses across datasets and models, though without conclusive evidence of a joint AMR-SPARQL model that works consistently in different settings.

Overall, we examined how well a semantic structure serves as a gauge of generative model confidence. We further hope our dataset will support work in multilingual QA systems as well as in many other directions, including research into the use of meaning representations for model interpretability, using linguistic structure as an inductive bias in model training, as well as studies in construction grammar and language typology.

7 Ethical considerations

Informed Consent: We ensured that all individuals providing annotations were fully informed about the purpose of the annotation task, how their data will be used, and what rights they have in relation to their data.

Fair Compensation: We ensured that individuals providing annotations were fairly compensated for their time and effort. For this project, professional annotators were compensated at least $30/hour, working between 20-80 hours each for the duration of data collection.

Transparency: We were transparent about the purpose and scope of the annotation task, as well as the potential benefits of the project, helping to build trust with individuals providing annotations and ensuring that they understood the significance of their contributions. We intend that through these practices data annotation efforts are overall more effective, resulting in a higher quality resource.

Environmental impact: We considered the environmental impact of the research, including the energy consumption of computing resources used. With GPT-4 inference, we limited input to 100s of examples to reduce costs. In-house fine-tuning was done using parameter efficient fine-tuning methods, allowing all experiments to be done on 1-2 NVIDIA Quadro RTX 8000 GPUs in <24 hours per run.

8 Limitations

1. Our work involved research into multilingual SPARQL and AMR parsing; though our dataset includes 52 languages, we report results on no more than 10-12 of these. Many of the languages we included are Indo-European,

with only a few exceptions (Korean, Japanese, Amharic, Vietnamese).

2. No experiments in joint AMR-SPARQL parsing involved hypotheses about performance across languages, though some evidence of performance shifts has been observed.

3. Fine-tuning models was done with less than 6k AMRS and 3-4k SPARQL training examples. Test data was limited to 100s examples per language in order to allow for multiple iterations and explore hyperparameter settings. Increasing the sizes of training and test sets is left for future work.

4. We only tested four large language models in this work (GPT2-XL, GPT-3.5, GPT-4, LLaMa). LLaMa does include multilingual data in training (Touvron et al., 2023), particularly languages using Latin and Cyrillic scripts. We did not test models explicitly trained for multilingual purposes and for other scripts, leaving such work for the future.

5. The MASSIVE-AMR dataset matches multilingual utterances to unique AMR graphs, making it the largest such dataset to date. However, unlike QALD9-AMR (Lee et al., 2022), MASSIVE-AMR does not include gold SPARQL queries. We emphasize that the use case we explore in this paper is only one of many possible, and we hope future research explores beyond this single application.

9 Bibliographical References

References

Renat Aksitov, Chung-Ching Chang, David Reitter, Siamak Shakeri, and Yunhsuan Sung. 2023. Characterizing attribution and fluency tradeoffs for retrieval-augmented large language models. *arXiv:2302.05578*.

Chris Alberti, Ciprian Chelba, Kuzman Ganchev, Michael Collins, and Sebastian Gehrmann. 2022. Towards computationally verifiable semantic grounding for language models. *arXiv:2211.09070*.

Yoav Artzi, Kenton Lee, and Luke Zettlemoyer. 2015. Broad-coverage CCG semantic parsing with AMR. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1699–1710, Lisbon, Portugal. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Debayan Banerjee, Pranav Ajit Nair, Jivat Neet Kaur, Ricardo Usbeck, and Chris Biemann. 2022. Modern baselines for SPARQL semantic parsing. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of AAAI*.

Mihaela Bornea, Ramon Fernandez Astudillo, Tahira Naseem, Nandana Mihindukulasooriya, Ibrahim Abdelaziz, Pavan Kapanipathi, Radu Florian, and Salim Roukos. 2022. Learning to transpile amr into sparql. *arXiv:2112.07877*.

Shu Cai and Kevin Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 748–752, Sofia, Bulgaria. Association for Computational Linguistics.

William Croft. 2002. *Typology and Universals*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Ruixiang Cui, Rahul Aralikatte, Heather Lent, and Daniel Hershcovich. 2022. Compositional Generalization in Multilingual Semantic Parsing over Wikidata. *Transactions of the Association for Computational Linguistics*, 10:937–955.

Rajarshi Das, Tsendsuren Munkhdalai, Xingdi Yuan, Adam Trischler, and Andrew McCallum. 2018. Building dynamic knowledge graphs from text using machine reading comprehension. *arXiv:1810.05682*.

Andrew Drozdov, Jiawei Zhou, Radu Florian, Andrew McCallum, Tahira Naseem, Yoon Kim, and Ramon Fernandez Astudillo. 2022. Inducing and using alignments for transition-based amr parsing. *arXiv:2205.01464*.

Allyson Ettinger, Jena D. Hwang, Valentina Pyatkin, Chandra Bhagavatula, and Yejin Choi. 2023. "you are an expert linguistic annotator"': Limits of llms as analyzers of abstract meaning representation. *EMNLP 2023 Findings*.

Xing Fan, Emilio Monti, Lambert Mathias, and Markus Dreyer. 2017. Transfer learning for neural semantic parsing. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 48–56, Vancouver, Canada. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT Sentence Embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. *arXiv:2204.08582*.

Jeffrey Flanigan, Sam Thomson, Jaime Carbonell, Chris Dyer, and Noah A. Smith. 2014. A discriminative graph-based parser for the Abstract Meaning Representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1426–1436, Baltimore, Maryland. Association for Computational Linguistics.

Saadia Gabriel, Asli Celikyilmaz, Rahul Jha, Yejin Choi, and Jianfeng Gao. 2021. GO FIGURE: A meta evaluation of factuality in summarization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 478–487, Online. Association for Computational Linguistics.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Jan Hajič, Ondřej Bojar, and Zdeňka Urešová. 2014. Comparing Czech and English AMRs. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 55–64, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

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 Comput. Surv.*, 55(12).

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Christopher Johnson and Charles J. Fillmore. 2000. The framenet tagset for frame-semantic and syntactic coding of predicate-argument structure. In *6th Applied Natural Language Processing Conference, ANLP 2000*, pages 56–62. ACL.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know. *arXiv*.

Pavan Kapanipathi, Ibrahim Abdelaziz, Srinivas Ravishankar, Salim Roukos, Alexander Gray, Ramón Fernandez Astudillo, Maria Chang, Cristina Cornelio, Saswati Dana, Achille Fokoue, Dinesh Garg, Alfio Gliozzo, Sairam Gurajada, Hima Karanam, Naweed Khan, Dinesh Khandelwal, Young-Suk Lee, Yunyao Li, Francois Luus, Ndivhuwo Makondo, Nandana Mihindukulasooriya, Tahira Naseem, Sumit Neelam, Lucian Popa, Revanth Gangi Reddy, Ryan Riegel, Gaetano Rossiello, Udit Sharma, G P Shrivatsa Bhargav, and Mo Yu. 2021. Leveraging Abstract Meaning Representation for knowledge base question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3884–3894, Online. Association for Computational Linguistics.

Young-Suk Lee, Ramón Astudillo, Hoang Thanh Lam, Tahira Naseem, Radu Florian, and Salim Roukos. 2022. Maximum Bayes Smatch ensemble distillation for AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5379–5392, Seattle, United States. ACL.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6(2):167–195.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1000–1008, Online. Association for Computational Linguistics.

R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. 2002. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827.

Roberto Navigli, Rexhina Blloshmi, and Abelardo Martinez Lorenzo. 2022. Babelnet Meaning Representation: A fully semantic formalism to overcome language barriers. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Yilin Niu, Fei Huang, Wei Liu, Jianwei Cui, Bin Wang, and Minlie Huang. 2023. Bridging the gap between synthetic and natural questions via sentence decomposition for semantic parsing. *Transactions of the Association for Computational Linguistics*, 11:367–383.

OpenAI. 2023. Gpt-4 technical report. *arXiv:2303.08774*.

Juri Opitz. 2023. SMATCH++: Standardized and extended evaluation of semantic graphs. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1595–1607, Dubrovnik, Croatia. Association for Computational Linguistics.

Juri Opitz, Letitia Parcalabescu, and Anette Frank. 2020. AMR similarity metrics from principles. *Transactions of the Association for Computational Linguistics*, 8:522–538.

Aleksandr Perevalov, Dennis Diefenbach, Ricardo Usbeck, and Andreas Both. 2022. Qald-9-plus: A multilingual dataset for question answering over dbpedia and wikidata translated by native speakers. In *2022 IEEE 16th International Conference on Semantic Computing (ICSC)*.

Jakob Prange, Nathan Schneider, and Lingpeng Kong. 2022. Linguistic frameworks go toe-to-toe at neuro-symbolic language modeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4375–4391, Seattle, United States. Association for Computational Linguistics.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. Measuring Attribution in Natural Language Generation Models. *Computational Linguistics*, pages 1–64.

Karin Kipper Schuler. 2006. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. 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 641–651, New Orleans, Louisiana. Association for Computational Linguistics.

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. *arXiv:2302.13971*.

Ricardo Usbeck, Ria Gusmita, Axel-Cyrille Ngonga Ngomo, and Muhammad Saleem. 2018. 9th challenge on question answering over linked data (qald-9). In *Joint proceedings of the 4th Workshop on Semantic Deep Learning (SemDeep-4) and NLIWoD4*.

Jens Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, Jan Hajič, James Martin, Stephan Oepen, Martha Palmer, James Pustejovsky, Rosa Vallejos, and Nianwen Xue. 2021. Designing a uniform meaning representation for natural language processing. *Künstliche Intelligenz*.

Cunxiang Wang, Zhikun Xu, Qipeng Guo, Xiangkun Hu, Xuefeng Bai, Zheng Zhang, and Yue Zhang. 2023. Exploiting abstract meaning representation for open-domain question answering. *arXiv:2305.17050*.

Shira Wein, Wai Ching Leung, Yifu Mu, and Nathan Schneider. 2022. Effect of source language on AMR structure. In *Proceedings of the 16th Linguistic Annotation Workshop (LAW-XVI) within LREC2022*, pages 97–102, Marseille, France. European Language Resources Association.

Shira Wein and Nathan Schneider. 2022. Accounting for language effect in the evaluation of cross-lingual AMR parsers. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3824–3834, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625, Seattle, United States. Association for Computational Linguistics.

Nianwen Xue, Ondřej Bojar, Jan Hajič, Martha Palmer, Zdeňka Urešová, and Xiuhong Zhang. 2014. Not an interlingua, but close: Comparison of English AMRs to Chinese and Czech. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1765–1772, Reykjavik. European Language Resources Association.

John M. Zelle and Raymond J. Mooney. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 2*, AAAI'96, page 1050–1055. AAAI Press.

Luke S. Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI'05, page 658–666, Arlington, Virginia, USA. AUAI Press.

Qin Zhang, Shangsi Chen, Dongkuan Xu, Qingqing Cao, Xiaojun Chen, Trevor Cohn, and Meng Fang. 2023. A survey for efficient open domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14447–14465, Toronto, Canada. Association for Computational Linguistics.

Zixuan Zhang and Heng Ji. 2021. Abstract Meaning Representation guided graph encoding and decoding for joint information extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 39–49, Online. Association for Computational Linguistics.

10 Appendices

A Characterizing Massive-AMR

A.1 AMR top nodes across datasets

AMR 3.0	#	QALD9-AMR	#	MASSIVE-AMR	#
and	7k	give-01	76	rate-01	105
say-01	3k	have-03	50	define-01	103
contrast-01	3k	have-degree	27	tell-01	94
multi-sentence	1.7k	have-org-role	21	have-quant	87
possible-01	1.7k	be-located-at	15	equal-01	86
cause-01	1.6k	die-01	14	price-01	70
state-01	1.5k	write-01	14	describe-01	66
have-concession	944	bear-02	13	be-located-at	64
think-01	901	marry-01	13	person	58
person	705	show-01	12	mean-01	50
have-03	618	locate-01	10	have-degree	50
have-condition	605	have-rel-role	10	bear-02	46
date-entity	538	person	9	have-org-role	32
know-01	451	name-01	9	show-01	21
have-degree	440	list-01	8	find-01	21

Table 11: 15 most frequent top AMR nodes in AMR 3.0, QALD9-AMR and MASSIVE-AMR, with counts for a single language (English).

A.2 Describing the MASSIVE long tail

We note some long-tail characteristics of utterances in MASSIVE (FitzGerald et al., 2022).

- Outliers in terms of utterance length: some 1-2 tokens, others quite long (40+ tokens)
- Ambiguous referents ('chase' in 'is chase doing good' may refer to a bank, person, or activity)
- Incomplete arithmetic ('tell me what equals two three')
- Less frequent expressions ('who is the better half of obama')
- Incomplete questions ('synonym for word', 'is equal to', 'research someone')

A.3 Languages in QALD-9, MASSIVE-, and MASSIVE+

	Language	# speakers	# Wiki pgs
QALD9/MASSIVE-	English	1.5b	58.7m
	French	320m	12.6m
	Russian	258m	7.7m
	German	76.5m	7.8m
	Italian	66m	7.7m
	Lithuanian	2.8m	0.5m
MASSIVE+	Vietnamese	85.2m	19.4m
	Japanese	125m	4.0m
	Korean	81.7m	3.1m
	Hungarian	8.2m	1.5m
	Urdu	91.5m	1.0m
	Amharic	31m	15k
	Azeri	24m	195k
	Finnish	5.1m	1.4m

Table 12: Common Indo-European languages in QALD-9 and MASSIVE- (top) and a more diverse sample we call MASSIVE+ (bottom) with some statistics based on https://meta.wikimedia.org/wiki/List_of_Wikipedias.

B Model details

B.1 Details about fine-tuning

	Element	Detail
Fine-tuning	Train set (QALD9/MASSIVE-AMR)	6000/2000
	Train set (SPARQL/AMR)	3000/5000
	Train set (langs)	1300 exs for 6 langs.
	Test set (QALD9/MASSIVE-AMR)	700/500
	Test set (langs)	200 exs for 6 langs.
	Block size (GPT-2/LlaMa)	512/2048
	Number epochs	8-16
	Learning rate	$3e^{-5}$
	Optimizer	AdamW
Inference	Number beams	20
	Number beam groups	10
	Diversity penalty	1.0
	Minimum length	8
	Maximum length	256

Table 13: Finetuning and inference details for GPT-2-XL and LlaMa (from Hugging Face).

B.2 Details about in-context learning

Element	Detail
Number in-context exs.	8-12
Number tokens in prompt	2400
In-context langs.	English, Spanish
Test set (QALD9/MASSIVE-AMR)	150/150
Test set (langs)	20-40 exs for 10 langs.
Temperature	1.0

Table 14: Settings for GPT-3.5 and GPT-4.

C Example prompts

C.1 SPARQL generation (Research question 1b)


```
{["messages": [
    {
        "role": "system",
        "content": "You are SPARQL-Predictor-GPT, a language model that predicts one SPARQL query for a given question."
    },
    {
        "role": "system",
        "content": "This is important! All relations in the predicted SPARQL query must be in
                the allowed_relation_list: ['abbreviation', 'accessioneudate', ... 'leadername']."
    },
    {
        "role": "system",
        "content": "Make a list of all the relations in the predicted SPARQL query.
                Verify that all relations are in allowed_relation_list,
                reporting True or False for each relation."
    },
    {
        "role": "system",
        "content": "Return the prediction in JSON form:
                {'sparql_query': str, 'relations': list(), 'verification': list()}"
    },
    {
        "role": "user",
        "content": "where did abraham lincoln die"
    },
    {
        "role": "assistant",
        "content": "{'sparql_query': 'PREFIX dbo: <http://dbpedia.org/ontology/>
                                PREFIX res: <http://dbpedia.org/resource/>
                                SELECT DISTINCT ?uri WHERE
                                { res:Abraham_Lincoln dbo:deathPlace ?uri. }',
                    'relations': ['deathplace'],
                    'verification': [True] }"
    },
    {
        "role": "user",
        "content": "give me all cosmonauts."
    }
],
"gold_sparql": "SELECT DISTINCT ?uri WHERE
                { ?uri a <http://dbpedia.org/ontology/Astronaut>
                { ?uri <http://dbpedia.org/ontology/nationality> <http://dbpedia.org/resource/Russia> }
                UNION
                { ?uri <http://dbpedia.org/ontology/nationality> <http://dbpedia.org/resource/Soviet_Union> } }",
"perturbed": 1,
"rel_excluded": "nationality",
"question": "give me all cosmonauts.",
"completion": "{'sparql_query': 'PREFIX dbo: <http://dbpedia.org/ontology/>
                            PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
                            SELECT DISTINCT ?cosmonaut WHERE
                            { ?cosmonaut rdf:type dbo:Astronaut. }',
                'relations': ['type'],
                'verification': [True] }",
"label": "qald9",
"hallucination": 0,
"hallucination-detection": 0,
"sparql_executable": 1,
"sparql_result": 1,
"sparql_malformed": 0}
```


Figure 1: Example prompt for SPARQL generation with completion and associated features. Our controlled setting for hallucination detection is then reduced to verifying all relations in a generated query are in the given list, verification the model outputs along with the generated sparql_query. For considerations of space, we show only 3 (of 140) relations given as the allowed_relation_list (second system message in prompt).

D AMR generation examples (GPT-4, 8-shot learning) and error analysis

Ex. 1 (QALD9, Spanish): cuánto pesa el satélite más ligero de júpiter (English: ‘how much does jupiter’s lightest moon weigh’)

```
Reference

(h2 / heavy
   :quant (a / amr-unknown)
   :domain (m / moon
      :ARG1-of (h / have-degree-91
         :ARG2 (l / light-06
            :ARG1 m)
         :ARG3 (m2 / most)
         :ARG5 (m3 / moon
            :poss (p / planet
               :name (n / name
                  :op1 "Jupiter"))))))
```

```
Generated

(p / pesar-01
   :ARG1 (s / satellite
      :degree (m / more
         :quant (l / light))
      :location-of (j / planet
         :name (n / name
            :op1 "Júpiter"))))

F1: 0.3
Error: missing amr-unknown; non-English frame
```

Ex. 2 (QALD9, French): combien de théories albert einstein a-t-il produit (English: ‘how many theories did albert einstein come up with’)

```
Reference

(c / come-up-11
   :ARG0 (p / person
      :name (n / name
         :op1 "Albert"
         :op2 "Einstein"))
   :ARG1 (t / theory
      :quant (a / amr-unknown)))
```

```
Generated

(p / produce-01
   :ARG0 (p2 / person
      :name (n / name
         :op1 "Albert"
         :op2 "Einstein"))
   :ARG1 (t / theory
      :quant (c / count-01
         :op1 (a / amr-unknown))))

F1: 0.77
Diff: English come-up-11 vs French produce-01
Note: Better metric needed for semantic similarity
```

Ex. 3 (Massive, English): what is the worlds record for the longest head of hair

```
Reference

(h / have-degree-91
    :ARG1 (h2 / hair
        :part-of (h3 / head)
        :mod (a / amr-unknown))
    :ARG2 (l / long-03
        :ARG1 h2)
    :ARG3 (m / most)
    :ARG5 (w2 / world)
    :ARG1-of (r / record-01))
```

```
Generated

(r / record-01
    :mod (w / world)
    :topic (l / long-02
        :degree (m / most)
        :ARG1 (h / hair-01
            :part-of (p / person))
        :duration (a / amr-unknown)))

F1: 0.38
Error: invents hair-01
Semantics of amr-unknown as 'duration'
Good: 'longest head of hair' = 'longest hair'
```

Errors in AMR generation include:

- Lack of AMR token for questions ‘amr-unknown’ (33/100 examples, see Ex. 1); We note 6 of 8 examples in the prompt model this (2 of 8 are imperatives);
- 3/100 generations filtered by Smatch (missing parenthesis or nodes with duplicate variable name);
- Frequent errors with AMR -91 frames (Exs. 1 & 3), e.g., rate-entity-91, have-degree-91;
- 21 of 100 examples have F1<0.5 (Ex. 3); of these, 5 are QALD9 and 16 are MASSIVE-AMR
- Arithmetic problems are correct but do not follow canonical form, though we note there are no math examples in prompts; this suggests more than one reference graph would be helpful for evaluation.

E Results of SPARQL generation

```
Example 1: Massive- (Italian), Perturbed=False, Hallucination=False, Detection=False, Difficult=False

Question: quanti anni ha giorgia (English: `how old is giorgia`)

Completion: {'generated_sparql_query':
    'PREFIX dbo: <http://dbpedia.org/ontology/>
     PREFIX res: <http://dbpedia.org/resource/>
     SELECT (YEAR(NOW()) - YEAR(?birthDate)) AS ?age WHERE
        { res:Giorgia dbo:birthdate ?birthDate. }',
    'relations': ['birthdate'],
    'verification': [True]}

-------

Example 2: Massive+ (Hungarian), Perturbed=True, Hallucination=True, Detection=False, Difficult=False

Relation excluded: ['birthdate']

Question: milyen id s t th vera (English: `how old is toth vera`)

Completion: {'generated_sparql_query':
    'PREFIX dbo: <http://dbpedia.org/ontology/>
     PREFIX res: <http://dbpedia.org/resource/>
     SELECT (YEAR(NOW()) - YEAR(?birthdate)) WHERE
        { res:T th_Vera dbo:birthDate ?birthdate. }',
    'relations': ['birthDate'],
    'verification': [True]}

-------

Example 3: Massive+ (Japanese), Perturbed=True, Hallucination=True, Detection=True, Difficult=False

Relation excluded: ['birthdate']

Question:                                   (English: `how old is atsuro watabe`)

Completion: {'generated_sparql_query':
    'PREFIX dbo: <http://dbpedia.org/ontology/>
     PREFIX res: <http://dbpedia.org/resource/>
     SELECT DISTINCT ?date WHERE
        { res:Atsuro_Watabe dbo:birthDate ?date. }',
    'relations': ['birthDate'],
    'verification': [False]}

-------

Example 4: Massive- (English), Perturbed=True, Hallucination=??, Detection=False, Difficult=True

Relation excluded: ['creator']

Question: who created the comic captain america?

{'generated_sparql_query':
    'PREFIX dbo: <http://dbpedia.org/ontology/>
    PREFIX res: <http://dbpedia.org/resource/>
    SELECT DISTINCT ?uri WHERE
        { res:Captain_America dbo:foundedby ?uri. }',
    'relations': ['foundedby'],
    'verification': [True]}
```

Figure 2: Examples of SPARQL generation using GPT-3.5 showing hallucinations and hallucination detection.