QASem Parsing: Text-to-text Modeling of QA-based Semantics

Ayal Klein **Eran Hirsch** **Ron Eliav**
Valentina Pyatkin **Avi Caciularu** **Ido Dagan**
Computer Science Department, Bar-Ilan University
{ayal.s.klein,hirsch.eran,roneliav1,valpyatkin,avi.c33}@gmail.com
dagan@cs.biu.ac.il

Abstract

Several recent works have suggested to represent semantic relations with questions and answers, decomposing textual information into separate interrogative natural language statements. In this paper, we consider three QA-based semantic tasks — namely, QA-SRL, QANom and QADiscourse, each targeting a certain type of predication — and propose to regard them as jointly providing a comprehensive representation of textual information. To promote this goal, we investigate how to best utilize the power of sequence-to-sequence (seq2seq) pre-trained language models, within the unique setup of semi-structured outputs, consisting of an unordered set of question-answer pairs. We examine different input and output linearization strategies, and assess the effect of multitask learning and of simple data augmentation techniques in the setting of imbalanced training data. Consequently, We release the first unified QASem parsing tool, practical for downstream applications who can benefit from an explicit, QA-based account of information units in a text.

1 Introduction

A traditional line of research in NLP has been devoted to designing various semantic representations, that aim to explicate textual meaning with a formal, consistent annotation schema. Representations such as Semantic Role Labeling (SRL; e.g. Baker et al., 1998) or Discourse Representation Theory (Kamp et al., 2011), provide applications with an explicit account of semantic relations in a text, facilitating downstream processing (Lee and Goldwasser, 2019; Huang and Kurohashi, 2021; Mohamed and Oussalah, 2019, inter alia). While traditional representations rely on a pre-defined schema or lexicon of linguistic classes (e.g. semantic roles), recent approaches aim for more loosely-structured, easily attainable representations, comprised mainly of natural language fragments (Etzioni et al., 2008; Elazar et al., 2021), and specifically questions and answers (Michael et al., 2018). For example, many recent works leverage QAs as an intermediate structure for assessing information alignment between texts, e.g. for evaluating summarization quality (Eyal et al., 2019; Gavenavicius, 2020; Deutsch et al., 2021) and faithfulness (Honovich et al., 2021; Durmus et al., 2020). Nevertheless, standard QA datasets are not designed for providing a systematic and coherent representation of text meaning.

In this work, we consider an evolving paradigm, consisting tasks that aim to comprehensively capture certain types of predications using question-answer pairs. A pioneering prominent work in this framework is Question Answer driven Semantic Role Labeling (QA-SRL; He et al., 2015). Targeting verbal predicates, QA-SRL labels each predicate-argument relation with a question-answer pair, where a natural language question represents semantic role, while answers correspond to arguments (See Table 1). This appealing framework, well suited for scalable crowdsourcing (Fitzgerald et al., 2018), has been extended to account for deverbal nominalizations (QANom; Klein et al., 2020). A subsequent work targeted information-bearing discourse relations using semi-templated questions and answers (QADiscourse; Pyatkin et al., 2020). We deem these individually-presented tasks as milestones toward a broad-coverage QA-based semantic representation, which we denote as *QASem*. To make this goal accessible, we develop a comprehensive modeling framework and release the first unified tool for parsing a sentence into a systematic set of QAs. This set covers the core information units in a sentence based on the above three predication types: verbs, deverbal nominalizations, and informational discourse relations.

Current best models for QA-SRL/QANom and QADiscourse (Fitzgerald et al., 2018; Pyatkin et al., 2020) are classifier-based pipelines, each targeting

Both were **shot** in the **confrontation** with police and have been **recovering** in hospital since the **attack** .			
QA-SRL	1	When was someone **shot?**	in the confrontation ; the attack
	2	Who was **shot?**	Both
	3	Who **shot** someone?	police
	4	Where has someone been **recovering?**	in hospital
	5	How long was someone **recovering** from something?	since the attack
	6	Who was **recovering** from something?	Both
	7	What was someone **recovering** from?	shot
QANom	8	Who **confronted** with something?	Both
	9	What did someone **confront** with?	police
QADiscourse	10	*Since when* have both been **recovering** in hospital?	since the **attack**
	11	*While what* were both **shot**?	During the **confrontation** with police

Table 1: An example sentence annotated with QA-SRL, QANom and QADiscourse. Target predicates (verbs and nominalizations) are shown in **bold**, while QADiscourse prefixes are shown in *italics*. Multiple answers are delimited by a semicolon (;).

a specific QA format. Predictors of relation labels (questions) use a specialized architecture that suits the task-specific question structure. Question are selected from a constrained set of choices and are modeled independently from relation participants (answers). Our work leverages recent progress in text-to-text pre-trained neural models for predicting semi-structured QA-based annotations in a generic manner. Specifically, we adopt the T5 model (Raffel et al., 2020), a sequence-to-sequence encoder-decoder transformer (Vaswani et al., 2017) massively pre-trained on language modeling.

While text-to-text models were originally used for conditional text generation tasks, such as machine translation and summarization, recent trends utilize them for tasks with other types of outputs, such as classification, tagging and structured prediction. Our QA-based semantics use-case is an interesting mid-ground between natural language generation and structured prediction tasks, as our semi-structured output sequence includes *a set* of *restricted natural language* fragments (the QAs), jointly comprising a linguistically meaningful representation of the input. Thus, unlike common setups of structured prediction via seq2seq, the QA set output can harness the model's pre-trained language generation capabilities rather than merely its language understanding.

We find that fine-tuning T5 on the QA-based semantic tasks is favorable over prior approaches, producing state-of-the-art models for all the aforementioned tasks. Our experiments suggest that T5 is good at learning the grammar characterizing our semi-structured outputs, and that input and output linearization strategies have a significant effect on performance.

In addition, we explore joint multi-task training of nominal and verbal QA-SRL in the seq2seq fine-tuning setting, to address the smaller size of the labeled nominal dataset. Our positive results indicate that verbs and nominalizations can transfer beneficial linguistic knowledge to each other.

Our tool, including models and code, is made publicly available.[1]

2 Background

2.1 QA-based Semantic Representation

The traditional goal of semantic representations is to reflect the meaning of texts in a formal, explicit manner (Abend and Rappoport, 2017). SRL schemes (Baker et al., 1998; Kingsbury and Palmer, 2002; Schuler, 2005), for example, decompose a textual clause into labeled predicate-argument relations specifying "who did what to whom", while discourse-level representations (Mann and Thompson, 1987; Kamp et al., 2011; Prasad et al., 2008) capture inter-clause relations. Such semantic representations can be leveraged by NLP applications that require an explicit handle of textual content units for their algorithms — for example, content selection for text generation tasks (Mohamed and Oussalah, 2019; Liu et al., 2015; Hardy and Vlachos, 2018) or information consolidation in multi-document settings (Liao et al., 2018; Pasunuru et al., 2021; Chen and Durrett, 2021).

A main drawback of these well-designed formalisms is their annotation cost — since they rely on linguistically-oriented categories (e.g. frames, semantic roles or scene types), dataset construction requires extensive annotator training, restricting their applicability to new text domains and new languages.

[1]We publish a unified package for jointly producing all QASem layers of annotation with an easy-to-use API — https://github.com/kleinay/QASem. The repository also includes model training and experiments code.

In recent years, several works proposed to remedy this annotation bottleneck by taking a more "open-ended" approach, capturing semantics using natural language self-explanatory terms (Butnariu et al., 2009; Shi and Demberg, 2019; Elazar et al., 2021). A related trend, mentioned above (§1), have suggested to utilize question-answering models for soliciting a manageable, discrete account of the information units in a text. While it is appealing to use questions and answers as a natural linguistic mechanism of information focus, common QA datasets (e.g. SQuAD; Rajpurkar et al., 2016) were developed for purposes unrelated to semantic representation, e.g. evaluating machine reading comprehension.

This paper targets three question-answer driven semantic representations — namely, QA-SRL, QANom and QADiscourse — hereafter joined under the term *QASem*. The QASem line of research can be seen as an overarching project of developing a comprehensive, layered representation scheme, covering all important types of information conveyed by a text. We now turn to present the three current building blocks of QASem, which are illustrated in Table 1.[2]

2.2 QASem Tasks

QA-SRL With the goal of collecting laymen-intuitive semantic annotations, QA-SRL (He et al., 2015) annotates verbs with a set of natural language question-answer pairs (QAs), where each QA corresponds to a single predicate-argument relation. QA-SRL questions adhere to a 7-slots template, with slots corresponding to a WH-word, the verb, auxiliaries, argument placeholders (SUBJ, OBJ), and a preposition. A question is aligned with one or more answers (when a role has multiple fillers), each is a continuous span from the sentence.

QA-SRL was shown to be attainable through cost-efficient crowdsourced annotations (Fitzgerald et al., 2018). Nonetheless, beyond data collection scalability, QA-SRL yields a richer argument set than traditional, linguistically-rooted formalisms like PropBank (Kingsbury and Palmer, 2002), including valuable implicit arguments (Roit et al., 2020). It was also shown to subsume popular intermediate representations like OpenIE (Stanovsky and Dagan, 2016) and to enhance strong pre-trained encoders (He et al., 2020).

[2] Complementary missing pieces, which are at the stage of ongoing work, are to capture information specified by adjectival predicates and other noun modifiers.

QANom In a follow-up work, Klein et al. (2020) extended the QA-SRL framework to also cover deverbal nominal predicates, which are prevalent in texts. First, candidate nominalizations — nouns that have a derivationally related verb — are extracted using lexical resources (Miller, 1995). QANom annotators then classify whether the candidate carries a verbal, eventive meaning in context ("The **construction** of the offices...") or not ("It was a huge **construction**"). Then, predicative nominalizations undergo QA-SRL annotation, generating QAs in exactly the same format as verbal QA-SRL. The result is a unified framework for verbs and nominalizations (See Table 1), analogous to the relationship between the PropBank (Kingsbury and Palmer, 2002) and NomBank (Meyers et al., 2004) projects.

QADiscourse The relationship between propositions in a text can by itself deliver factual information. Several formalisms, such as Rhetorical Structure Theory (RST; Mann and Thompson, 1987) or the Penn Discourse TreeBank (PDTB; Miltsakaki et al., 2004), have labeled inter and intra-sentential discourse relations using a taxonomy of pre-defined relation senses, e.g. CONTINGENCY.CONDITION or TEMPORAL.ASYNCHRONOUS.SUCCESSION. Following the QA-SRL paradigm, Pyatkin et al. (2020) proposed to annotate discourse relations using natural language question-answer pairs (See Table 1). They devised a list of question prefixes (e.g. *In what case X?* or *After what X?*) corresponding to a subset of PDTB relation types capturing 'informational' relations, excluding senses specifying merely some structural or pragmatic properties of the realized passage. Annotators were presented with a sentence and certain heuristically extracted event targets marked in that sentence. They were then asked to relate such event targets with a question starting with one of the prefixes, if applicable. The question body (after the prefix) was a copied sentence span containing one of the targets whereas the answer span contained the other. Different from QA-SRL and QANom, both copied spans could be slightly edited to sound grammatical and fluent.

2.3 Prior QASem Models

As mentioned above, previous models for QA-SRL/QANom and QADiscourse were designed to match the specific question format of each of the tasks. We hereby provide further details about these models.

Leveraging its intuitive nature, Fitzgerald et al. (2018) crowdsourced a large-scale QA-SRL dataset. The dataset was then used for training a specialized argument-first pipeline model for parsing the concrete QA-SRL format, comprised of a span-level binary classifier for argument detection, followed by a question generator. The latter is an LSTM decoder which, given a contextualized representation of the selected span, sequentially predicts the 7 slots which comprise a QA-SRL question.

Since corresponding verbs and nominalizations share the same semantic frame, but are distinct in their syntactic argument structure, modeling both types of predicates jointly is a non-trivial yet promising approach (Zhao and Titov, 2020). Nevertheless, Klein et al. (2020) have only released a baseline parser, retraining the model of Fitzgerald et al. (2018) on QANom data alone. Their model achieves mediocre performance, presumably due to the limited amount of QANom training data, which is by an order of magnitude smaller than the training data available for verbal QA-SRL.

Pyatkin et al. (2020) modeled the QADiscourse task with a three-step pipeline. Utilizing the discrete set of question prefixes, they employ a prefix classifier, followed by a pointer generator model (Jia and Liang, 2016) to complete question generation. Finally, they fine-tune a machine reading comprehension model for selecting an answer span from the sentence per question.

Differing from previous pipeline approaches, we model each of the QASem tasks using a one-pass encoder-decoder architecture. In addition, we regard the three tasks as sub-tasks of a single unified framework, proposing a single architecture for parsing QA-based semantic annotations, also applicable for future extensions of the QASem framework.

3 Modeling

We release a *QASem tool* for parsing sentences with any subset of the QA-based semantic tasks. Our tool first executes sentence-level pre-processing for QA-SRL/QANom. It runs a part-of-speech tagger to identify verbs and nouns,[3] then applies candidate nominalization extraction heuristics (See §2) followed by a binary classifier for detecting predicative nominalizations (Klein et al., 2020). Identified predicates are then passed into the QA-SRL or QANom text-to-text parsing models, while the QADiscourse model takes a raw sentence as input with no pre-processing required. The models are described in detail in the following subsections.

3.1 Baseline Models

We first finetune pre-trained sequence-to-sequence language models on each of the QASem tasks separately (BASELINE). Unless otherwise mentioned, most modeling details specified hereafter apply also for the joint models (§3.2). We experiment both with BART (Lewis et al., 2020) and with T5 (Raffel et al., 2020), but report results and analyses only for the T5 model for clarity, as we consistently observed its performance to be significantly better. We use `T5-small` due to computational cost constraints.

Our text-to-text modeling for QA-SRL and QANom is at the *predicate-level* — given a single predicate in context, the task is to produce the full set of question-answer pairs targeting this predicate. Our input sequence consists of four components — task prefix, sentence, special markers for the target predicate, and verb-form — as in this nominalization example:

> *parse: Both were shot in the [PREDICATE] confrontation [PREDICATE] with police ... [SEP] confront*

The prefix (*"parse:"*) is added in order to match the T5 setup for multitask learning. Then, the sentence is encoded together with bilateral marker tokens signaling the location of the target predicate (we report alternative predicate highlighting methods in Appendix A.1). At last, the verbal form of the predicate (*"confront"*) is appended to the input sequence. This is significant for QANom, since the output verb-centered QA-SRL questions involve the verbal form of the nominal predicate. Verbal forms are identified during the candidate nominalization extraction phase in pre-processing, and are thus available both at train and at test time.[4]

Since the intended output is a *set* of QAs, one can impose any arbitrary order over them, where the only objective is enhancing model learning. We examine different output serialization strategies, resulting in significant performance differences, especially for QANom where dataset size is modest. See Section 5.2 for details about the

[3]we use SpaCy 3.0 — https://spacy.io/

[4]For verbal QA-SRL, appending the verb-form (which is the predicate itself) did not improve performance. However, in the joint verbal and nominal model, all instances are appended with a verb-form for consistency.

Task	QA-SRL			QANom			QADiscourse		
Dataset	2018	2020		(Klein et al., 2020)			(Pyatkin et al., 2020)		
Split	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Sentences	44476	1000	999	7114	1557	1517	7994	1834	1779
Predicates	95253	1000	999	9226	2616	2401	-	-	-
Questions	215427	2895	2852	15895	5577	4886	10985	2632	2996
Answers	348349	3546	3549	18900	6925	6064	10985	2632	2996

Table 2: QASem Datasets Statistics. QA-SRL Training set comes from Fitzgerald et al. (2018), while evaluation sets are from Roit et al. (2020).

effect of serialization strategy. Finally, the ordered QA list is joined into a structured sequence using three types of special tokens as delimiters — `QA|QA` separator, `Question|Answers` separator, and `Answer|Answer` separator for questions with multiple answers.

For the QADiscourse task we train a *sentence-level* model. The input is the raw sentence, while the output is the set of QA pairs pertaining to all targets occurring in the sentence. Inline with our approach in QA-SRL parsing, we prepend inputs with a new task prefix, and use special tokens as delimiters (`QA|QA` and `Question|Answer`).

3.2 Joint QASem Learning

Leveraging the shared output format of QA-SRL and QANom, we further train a unified model on both datasets combined (JOINT). Taking into account the imbalance in training set size for the two tasks, we duplicate QANom data samples by a factor of 14, approximating a 1:1 ratio between verbal and nominal predicates.

It is worth mentioning that we have tested several methods for incorporating explicit signal regarding the source task (i.e. predicate type — verbal or nominal) of each training instance, aiming to facilitate transfer learning and model generalization. Our experiments include: prefix variation (e.g. *"parse verbal/nominal:"*); typed predicate marker, i.e., having a different special token marker for verbal vs. nominal predicates; and appending the predicate type to the **output** sequence, simulating a predicate-type classification objective in an auxiliary multitask learning framework (e.g. Bjerva, 2017; Schröder and Biemann, 2020). Nonetheless, throughout all our experiments, uninformed joint learning of verbal and nominal predicates works significantly better.

4 Experimental Setup

Datasets We use the QADiscourse and QANom original datasets (Pyatkin et al., 2020; Klein et al., 2020). For QA-SRL, we make use of the large scale training set collected by Fitzgerald et al. (2018). However, prior work (Roit et al., 2020) pointed out that their annotation protocol suffered from limited recall along with multiple, partially overlapping reference answers, hindering parser evaluation. For these reasons, Roit et al. (2020) applied a controlled crowdsourcing procedure and produced a high-quality evaluation set, dedicated for fair comparison of future QA-SRL parsers. We adopt their annotations for validation and test.[5] Datasets statistics are presented in Table 2.

Evaluation Metrics Evaluating QA-based semantic tasks involves two core aspects. First, we would like to estimate how many of the *semantic relations* are captured correctly. For SRL, this is analogous to measuring argument detection, while for discourse, it assesses whether pairs of events are related to each other or not. Second, given that the model identified the same predicate-argument or predicate-predicate relation as present in the gold set, we want to assess its predicted label for the relation type (semantic role or discourse relation sense). A manifestation of these objectives for the QA-SRL and QADiscourse formats considers an *unlabeled* and a *labeled* evaluation measure per task (Roit et al., 2020; Pyatkin et al., 2020).

For computing QA-SRL's unlabeled argument detection (**UA**) metric, QAs in the predicted set are aligned to QAs in the reference set using maximum bipartite matching based on lexical intersection-over-union (IOU) of the answers. A pair of QAs must surpass a minimum IOU threshold Γ to count as aligned. Then, aligned QA pairs are re-inspected for question equivalence to form the labeled argument detection measure (**LA**).

QA-SRL question templates have no plain mapping to semantic roles, and determining whether

[5] All datasets related to the QA-driven Semantics paradigm have been uploaded to Huggingface's dataset hub, while unifying their data format to the extent possible — see the datasets at https://huggingface.co/biu-nlp.

Evaluation Protocol / Model		Klein et al. (2020)			Roit et al. (2020)		
		P	R	F1	P	R	F1
Fitzgerald et al. (2018)	**UA**	-	-	-	87.1	50.2	63.7
	LA	-	-	-	67.8	39.1	49.6
BASELINE (T5)	**UA**	77.6	64.4	**70.35**	79.8	59.3	**68.0**
	LA	65.9	54.7	**59.76**	61.6	45.8	**52.5**
JOINT (T5)	**UA**	76.2	62.4	68.62	80.2	57.9	67.3
	LA	63.9	52.4	57.59	61.2	44.2	51.3

Table 3: Results of verbal QA-SRL parsing on the test set from Roit et al. (2020).

Model		P	R	F1
Fitzgerald et al. (2018)	**UA**	45.1	61.5	52.0
	LA	29.6	40.4	34.2
BASELINE (T5)	**UA**	69.6	55.3	61.7
	LA	51.5	40.9	45.6
JOINT (T5)	**UA**	68.9	58.0	**63.0**
	LA	51.4	43.3	**47.0**

Table 4: Results of nominal QA-SRL parsing on the QANom test set (Klein et al., 2020).

two questions refer to the same role is non-trivial. Here we apply the evaluation measures put forward by Klein et al. (2020), using a technique for mapping questions into a discrete space of "syntactic roles", and setting $\Gamma = 0.3$. However, for a fair comparison to previous results on QA-SRL, we also report the evaluation measures of Roit et al. (2020), which set $\Gamma = 0.5$ and use a more strict question equivalence criterion.

As for QADiscourse, we simply embrace the **UQA** and **LQA** metrics proposed by Pyatkin et al. (2020). These are analogous to **UA** and **LA**, except that the unlabeled alignment between QA pairs is computed as IOU between question-and-answer tokens jointly ($\Gamma = 0.5$), excluding question prefix, while labeled alignment is simply a match over question prefixes.

Training Details We tuned the models' hyperparameters on the validation set with a random search over the learning rate, the dropout probability and the batch size. The joint QA-SRL and QANom model was tuned to optimize QANom evaluation measures.

5 Results

In this section, we present the experiments we conducted on the QASem tasks and the corresponding results.

5.1 Models Performance

QA-SRL and QANom Model evaluation results for QA-SRL are presented in Table 3, and results for QANom are presented in Table 4.[6] We can see that the T5-based models are improving over the previous approach with a substantial margin. Notably, the improvement for QANom is more profound. We ascribe this to its smaller training size, putting more weight on the pre-training phase. Similarly, nominal predicates significantly benefit from the joint learning with verbal instances, while the opposite does not hold true. This can also be attributed to training size — whereas verbal QA-SRL is slightly impaired from combining nominal instances into the training data, the benefit of nominal predicates from enlarging the training data by an order of magnitude overcome this adverse effect. Overall, turning to T5 improved QA-SRL and QANom F1 performance by %7 and %20 respectively compared to previous state-of-the-art parsers, while joint learning gains another %6 recall and %2 F1 for QANom.

QADiscourse Performance evaluation of our QADiscourse model over the QADiscourse task, compared to the previous pipeline model (Pyatkin et al., 2020), is reported in Table 5. While unlabeled detection of discourse relations is improving by a relatively small margin, the question quality — assessed by the LQA and prefix accuracy metrics — is substantially increased. Results suggest that the model is leveraging the generative language modeling pre-training, possibly making its generated question-answer statements more semantically sound, as may also be entailed from the large increase in precision (%8).

[6] All models in this section use the **Answer-Order** linearization method; see Section 5.2 for details.

	UQA			LQA	Prefix
	P	R	F1	Accuracy	Accuracy
Pyatkin et al. (2020)	80.8	86.8	83.7	66.6	49.9
Ours (T5)	87.0	84.3	**85.6**	**73.3**	**57.8**

Table 5: Evaluation results on the QADiscourse test set.

		QA-SRL Baseline			QANom Baseline		
		P	R	F1	P	R	F1
Random-Order	**UA**	74.1	58.6	65.47	67	47	55.25
	LA	61.6	48.7	54.43	48.4	34	39.93
Answer-Order	**UA**	74.7	63.8	**68.82**	65.5	51.9	57.93
	LA	62.5	53.3	**57.56**	46.3	36.7	40.91
All-Permutations	**UA**	63.1	64.8	64.0	57.5	57.5	57.5
	LA	51.0	52.3	51.6	39.3	39.4	39.4
Fixed-Permutations	**UA**	75.2	60.0	66.7	67.6	50.2	57.6
	LA	62.2	49.6	55.2	50.1	37.2	42.7
Linear-Permutations	**UA**	72.5	62.8	67.3	62.3	56.9	**59.5**
	LA	60.9	52.7	56.5	45.3	41.3	**43.2**

Table 6: Output linearization experiment results, comparing different methods for linearizing the set of QAs into output sequence(s). Evaluation metrics are those proposed by Klein et al. (2020).

5.2 The Effect of Output Set Linearization

As stated, the output of the model is parsed into a set of question-answer pairs at post-processing. Thus, the ordering one applies over the linearization of QAs into an output sequence can be arbitrary. It is therefore appealing to examine which ordering schemes facilitate model learning more than others. We investigate whether ordering the QAs according to answer position in the source sentence (**Answer-Order**) makes learning easier for the model, compared to a randomized order (**Random-Order**). Our hypothesis is that teaching the model to "scan" the sentence sequentially in the search for arguments of the predicate would produce a more systematic model with greater coverage.

In contrast to methods that confine the model to a fixed order, one could aim to teach the model to ignore QA ordering altogether. One way to achieve order invariance is to train over various permutations of the QA set rather than one order per instance (Ribeiro et al., 2021). In addition to order-invariance, training on multiple permutation may enhance performance from a strict data-augmentation perspective.

Clearly, including all the permutations of a QA set in the training data would result in an exponential imbalance toward predicates having more QAs. On the other hand, there are reasons to assume that these predicates would be generally harder for the model to learn; in this case, some degree of data imbalance might be beneficial.

We experiment with three permutation-based augmentation methods. The most straight-forward approach is to include all QA permutations of each predicate (**All-Permutations**).[7] One alternative is to avoid data imbalance by sampling (with replacement) a fixed number of k permutations for all predicates (**Fixed-Permutations**; we set $k = 3$). The third method samples $n = |QAs|$ permutations for each predicate, producing a linearly imbalanced training data in which instance frequency is proportional to the number of QA sub-sequences in its output (**Linear-Permutations**).

We train both QA-SRL and QANom baseline models using each of the above mentioned linearization methods, fixing the hyper-parameters for all models. Results are shown in Table 6.[8] Ordering the QAs in the output sequence by the position of answers in the input sentence indeed provides a performance boost for both QA-SRL and QANom baselines. Nonetheless, the **Linear-Permutations** method outperforms **Answer-Order** on QANom, mainly per recall, but not on QA-SRL. We conjecture that this gap as well is related to the data scarcity difference — since QA-SRL have abundant training samples, data augmentation is less effective and has lower priority compared to output's structural consistency. Overall, our experiment indicates that linearization techniques have a substantial effect on predicting a semi-structured task, like QA-prediction with seq2seq models.

[7] To avoid memory overflow, we restrict the number of incorporated permutations by $M = 10$.

[8] We have also applied the permutation-based methods on QADiscourse; however, none of these improved performance over the baseline model.

6 Conclusion

We propose to bundle three QA-based semantic tasks into a congruent conceptual paradigm. We hence train and release new state-of-the-art models for these tasks, based on a unified framework for fine-tuning a seq2seq pre-trained language model. Specifically, joint learning of verbal and nominal QA-driven SRL results in a significant boost for the nominal domain, where training data is limited. In addition, we show the importance of output linearization choices, and propose to sample permutations for augmenting training data while introducing a bias toward richer sequences. Utilizing these models, The QASem tool we release can be used in various downstream scenarios where an explicit account of textual information units is desired.

In future work, beyond incorporating upcoming QA-based semantic tasks into the current seq2seq framework, we plan to test sentence-level modeling for QA-SRL, inline with our proposed QADiscourse model. Further, we plan to extend our joint learning framework and to incorporate QA-SRL and QADiscourse, as well as other QA-formatted tasks, into a single multitask model.

References

Omri Abend and Ari Rappoport. 2017. The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 77–89.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98/COLING '98, page 86–90, USA. Association for Computational Linguistics.

Johannes Bjerva. 2017. Will my auxiliary tagging task help? estimating auxiliary tasks effectivity in multi-task learning. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 216–220, Gothenburg, Sweden. Association for Computational Linguistics.

Cristina Butnariu, Su Nam Kim, Preslav Nakov, Diarmuid O Séaghdha, Stan Szpakowicz, and Tony Veale. 2009. Semeval-2010 task 9: The interpretation of noun compounds using paraphrasing verbs and prepositions. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW-2009)*, pages 100–105.

Jifan Chen and Greg Durrett. 2021. Robust question answering through sub-part alignment. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1251–1263.

Daniel Deutsch, Tania Bedrax-Weiss, and Dan Roth. 2021. Towards question-answering as an automatic metric for evaluating the content quality of a summary. *Transactions of the Association for Computational Linguistics*, 9:774–789.

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.

Yanai Elazar, Victoria Basmov, Yoav Goldberg, and Reut Tsarfaty. 2021. Text-based np enrichment. *arXiv preprint arXiv:2109.12085*.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. 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 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

Nicholas Fitzgerald, Julian Michael, Luheng He, and Luke S. Zettlemoyer. 2018. Large-scale qa-srl parsing. In *ACL*.

Mantas Gavenavicius. 2020. Evaluating and comparing textual summaries using question answering models and reading comprehension datasets. B.S. thesis, University of Twente.

Hardy Hardy and Andreas Vlachos. 2018. Guided neural language generation for abstractive summarization using abstract meaning representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 768–773.

Hangfeng He, Qiang Ning, and Dan Roth. 2020. QuASE: Question-answer driven sentence encoding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8743–8758, Online. Association for Computational Linguistics.

Luheng He, Mike Lewis, and Luke S. Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *EMNLP*.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. Q2:: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870.

Yin Jou Huang and Sadao Kurohashi. 2021. Extractive summarization considering discourse and coreference relations based on heterogeneous graph. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3046–3052.

Robin Jia and Percy Liang. 2016. Data recombination for neural semantic parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12–22, Berlin, Germany. Association for Computational Linguistics.

Hans Kamp, Josef van Genabith, and Uwe Reyle. 2011. Discourse representation theory. In *Handbook of philosophical logic*, pages 125–394. Springer.

Paul Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *LREC*, pages 1989–1993. Citeseer.

Ayal Klein, Jonathan Mamou, Valentina Pyatkin, Daniela Stepanov, Hangfeng He, Dan Roth, Luke Zettlemoyer, and Ido Dagan. 2020. QANom: Question-answer driven SRL for nominalizations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3069–3083, Barcelona, Spain (Online). International Committee on Computational Linguistics.

I-Ta Lee and Dan Goldwasser. 2019. Multi-relational script learning for discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4214–4226.

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.

Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract meaning representation for multi-document summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1178–1190.

Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A Smith. 2015. Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086.

William C Mann and Sandra A Thompson. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. Annotating noun argument structure for NomBank. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Julian Michael, Gabriel Stanovsky, Luheng He, Ido Dagan, and Luke Zettlemoyer. 2018. Crowdsourcing question-answer meaning representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 560–568.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Eleni Miltsakaki, Rashmi Prasad, Aravind K Joshi, and Bonnie L Webber. 2004. The penn discourse treebank. In *LREC*. Citeseer.

Muhidin Mohamed and Mourad Oussalah. 2019. Srl-esa-textsum: A text summarization approach based on semantic role labeling and explicit semantic analysis. *Information Processing & Management*, 56(4):1356–1372.

Ramakanth Pasunuru, Mengwen Liu, Mohit Bansal, Sujith Ravi, and Markus Dreyer. 2021. Efficiently summarizing text and graph encodings of multi-document clusters. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4768–4779.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The penn discourse treebank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Valentina Pyatkin, Ayal Klein, Reut Tsarfaty, and Ido Dagan. 2020. QADiscourse - Discourse Relations as QA Pairs: Representation, Crowdsourcing and Baselines. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2804–2819, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2021. Investigating pretrained language models for graph-to-text generation. In *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*, pages 211–227, Online. Association for Computational Linguistics.

Paul Roit, Ayal Klein, Daniela Stepanov, Jonathan Mamou, Julian Michael, Gabriel Stanovsky, Luke Zettlemoyer, and Ido Dagan. 2020. Controlled crowdsourcing for high-quality QA-SRL annotation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7008–7013, Online. Association for Computational Linguistics.

Fynn Schröder and Chris Biemann. 2020. Estimating the influence of auxiliary tasks for multi-task learning of sequence tagging tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2971–2985, Online. Association for Computational Linguistics.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

Wei Shi and Vera Demberg. 2019. Learning to explicitate connectives with seq2seq network for implicit discourse relation classification. In *Proceedings of the 13th International Conference on Computational Semantics-Long Papers*, pages 188–199.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2300–2305, Austin, Texas. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yanpeng Zhao and Ivan Titov. 2020. Unsupervised transfer of semantic role models from verbal to nominal domain.

A Appendices

A.1 Alternative QA-SRL Linearization Methods

Here we specify in greater detail about experiments we ran assessing alternative linearization methods for QA-SRL and QANom models.

Concerning the input encoding, we experimented with four methods of highlighting the target predicate token within the sentence:

1. Repeating the target word at the end of the sequence
2. Special token before the target
3. Special token after the target
4. Special tokens before and after the target

Method 4. outperformed methods 2. and 3. by a small margin, while method 1. was worse.