IMoJIE: Iterative Memory-Based Joint Open Information Extraction

Keshav Kolluru[1], Samarth Aggarwal[1], Vipul Rathore[1], Mausam[1] and Soumen Chakrabarti[2]
[1] Indian Institute of Technology Delhi
{keshav.kolluru, samarth.aggarwal.2510, rathorevipul28}@gmail.com
mausam@cse.iitd.ac.in
[2] Indian Institute of Technology Bombay
soumen@cse.iitb.ac.in

Abstract

While traditional systems for Open Information Extraction were statistical and rule-based, recently neural models have been introduced for the task. Our work builds upon CopyAttention, a sequence generation OpenIE model (Cui et al., 2018). Our analysis reveals that CopyAttention produces a constant number of extractions per sentence, and its extracted tuples often express redundant information.

We present IMoJIE, an extension to CopyAttention, which produces the next extraction conditioned on all previously extracted tuples. This approach overcomes both shortcomings of CopyAttention, resulting in a variable number of diverse extractions per sentence. We train IMoJIE on training data bootstrapped from extractions of several non-neural systems, which have been automatically filtered to reduce redundancy and noise. IMoJIE outperforms CopyAttention by about 18 F1 pts, and a BERT-based strong baseline by 2 F1 pts, establishing a new state of the art for the task.

1 Introduction

Extracting structured information from unstructured text has been a key research area within NLP. The paradigm of Open Information Extraction (OpenIE) (Banko et al., 2007) uses an open vocabulary to convert natural text to semi-structured representations, by extracting a set of (subject, relation, object) tuples. OpenIE has found wide use in many downstream NLP tasks (Mausam, 2016) like multi-document question answering and summarization (Fan et al., 2019), event schema induction (Balasubramanian et al., 2013) and word embedding generation (Stanovsky et al., 2015).

Traditional OpenIE systems are statistical or rule-based. They are largely unsupervised in nature, or bootstrapped from extractions made by earlier systems. They often consist of several components like POS tagging, and syntactic parsing. To bypass error accumulation in such pipelines, end-to-end neural systems have been proposed recently.

Recent neural OpenIE methods belong to two categories: sequence *labeling*, e.g., RnnOIE (Stanovsky et al., 2018) and sequence *generation*, e.g., CopyAttention (Cui et al., 2018). In principle, generation is more powerful because it can introduce auxiliary words or change word order. However, our analysis of CopyAttention reveals that it suffers from two drawbacks. First, it does not naturally adapt the number of extractions to the length or complexity of the input sentence. Second, it is susceptible to *stuttering*: extraction of multiple triples bearing redundant information.

These limitations arise because its decoder has no explicit mechanism to remember what parts of the sentence have already been 'consumed' or what triples have already been generated. Its decoder uses a fixed-size beam for inference. However, beam search can only ensure that the extractions are not exact duplicates.

In response, we design the first neural OpenIE system that uses sequential decoding of tuples conditioned on previous tuples. We achieve this by adding every generated extraction so far to the encoder. This iterative process stops when the *EndOfExtractions* tag is generated by the decoder, allowing it to produce a variable number of extractions. We name our system **I**terative **M**em**O**ry **J**oint Open **I**nformation **E**xtraction (**IMoJIE**).

CopyAttention uses a bootstrapping strategy, where the extractions from OpenIE-4 (Christensen et al., 2011; Pal and Mausam, 2016) are used as training data. However, we believe that training on extractions of multiple systems is preferable. For example, OpenIE-4 benefits from high precision compared to ClausIE (Del Corro and Gemulla, 2013), which offers high recall. By aggregating extractions from both, IMoJIE could potentially

Sentence	He was appointed Commander of the Order of the British Empire in the 1948 Queen's Birthday Honours and was knighted in the 1953 Coronation Honours .
CopyAttention	(He ; was appointed ; Commander ... Birthday Honours) (He ; was appointed ; Commander ... Birthday Honours and was knighted ... Honours) (Queen 's Birthday Honours ; was knighted ; in the 1953 Coronation Honours) (He ; was appointed ; Commander of the Order of the British Empire in the 1948) (the 1948 ; was knighted ; in the 1953 Coronation Honours)
IMOJIE	(He ; was appointed ; Commander of the Order ... Birthday Honours) (He ; was knighted ; in the 1953 Coronation Honours)

Table 1: IMOJIE vs. CopyAttention. CopyAttention suffers from stuttering, which IMOJIE does not.

Sentence	Greek and Roman pagans , who saw their relations with the gods in political and social terms , scorned the man who constantly trembled with fear at the thought of the gods , as a slave might fear a cruel and capricious master .
OpenIE-4	(the man ; constantly trembled ;)
IMOJIE	(a slave ; might fear ; a cruel and capricious master) (Greek and Roman pagans ; scorned ; the man who ... capricious master) (the man ; constantly trembled ; with fear at the thought of the gods) (Greek and Roman pagans ; saw ; their relations with the gods in political and social terms)

Table 2: IMOJIE vs. OpenIE-4. Pipeline nature of OpenIE-4 can get confused by long convoluted sentences, but IMOJIE responds gracefully.

obtain a better precision-recall balance.

However, simply concatenating extractions from multiple systems does not work well, as it leads to redundancy as well as exaggerated noise in the dataset. We devise an unsupervised **Score-and-Filter** mechanism to automatically select a subset of these extractions that are non-redundant and expected to be of high quality. Our approach scores all extractions with a scoring model, followed by filtering to reduce redundancy.

We compare IMOJIE against several neural and non-neural systems, including our extension of CopyAttention that uses BERT (Devlin et al., 2019) instead of an LSTM at encoding time, which forms a very strong baseline. On the recently proposed CaRB metric, which penalizes redundant extractions (Bhardwaj et al., 2019), IMOJIE outperforms CopyAttention by about 18 pts in F1 and our strong BERT baseline by 2 pts, establishing a new state of the art for OpenIE. We release IMOJIE & all related resources for further research[1]. In summary, our contributions are:

- We propose IMOJIE, a neural OpenIE system that generates the next extraction, fully conditioned on the extractions produced so far. IMOJIE produce a variable number of diverse extractions for a sentence,
- We present an unsupervised aggregation scheme to bootstrap training data by combining extractions from multiple OpenIE systems.
- IMOJIE trained on this data establishes a new SoTA in OpenIE, beating previous systems and also our strong BERT-baseline.

[1] https://github.com/dair-iitd/imojie

2 Related Work

Open Information Extraction (OpenIE) involves extracting (arg1 phrase, relation phrase, arg2 phrase) assertions from a sentence. Traditional open extractors are rule-based or statistical, e.g., Textrunner (Banko et al., 2007), ReVerb (Fader et al., 2011; Etzioni et al., 2011), OLLIE (Mausam et al., 2012), Stanford-IE (Angeli et al., 2015), ClausIE (Del Corro and Gemulla, 2013), OpenIE-4 (Christensen et al., 2011; Pal and Mausam, 2016), OpenIE-5 (Saha et al., 2017, 2018), PropS (Stanovsky et al., 2016), and MinIE (Gashteovski et al., 2017). These use syntactic or semantic parsers combined with rules to extract tuples from sentences.

Recently, to reduce error accumulation in these pipeline systems, neural OpenIE models have been proposed. They belong to one of two paradigms: sequence *labeling* or sequence *generation*.

Sequence Labeling involves tagging each word in the input sentence as belonging to the subject, predicate, object or other. The final extraction is obtained by collecting labeled spans into different fields and constructing a tuple. RnnOIE (Stanovsky et al., 2018) is a labeling system that first identifies the relation words and then uses sequence labelling to get their arguments. It is trained on OIE2016 dataset, which postprocesses SRL data for OpenIE (Stanovsky and Dagan, 2016).

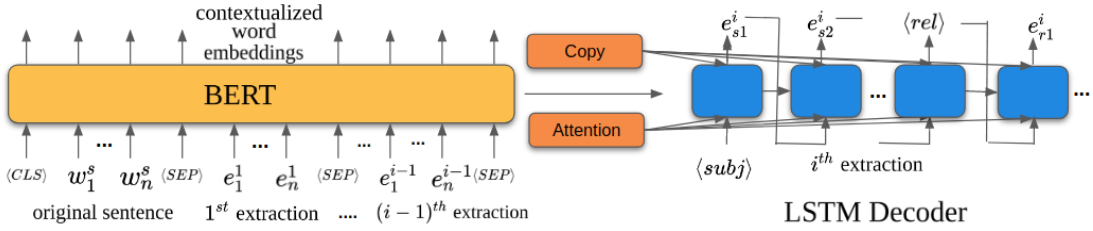

Figure 1: One step of the sequential decoding process, for generating the i^{th} extraction, which takes the original sentence and all extractions numbered $1, \ldots, i-1$, previously generated, as input.

SenseOIE (Roy et al., 2019), improves upon RnnOIE by using the extractions of multiple OpenIE systems as features in a sequence labeling setting. However, their training requires manually annotated gold extractions, which is not scalable for the task. This restricts SenseOIE to train on a dataset of 3,000 sentences. In contrast, our proposed *Score-and-Filter* mechanism is unsupervised and can scale unboundedly. Jiang et al. (2019) is another labeling system that better calibrates extractions across sentences.

SpanOIE (Zhan and Zhao, 2020) uses a span selection model, a variant of the sequence labelling paradigm. Firstly, the predicate module finds the predicate spans in a sentence. Subsequently, the argument module outputs the arguments for this predicate. However, SpanOIE cannot extract nominal relations. Moreover, it bootstraps its training data over a single OpenIE system only. In contrast, IMoJIE overcomes both of these limitations.

Sequence Generation uses a Seq2Seq model to generate output extractions one word at a time. The generated sequence contains field demarcators, which are used to convert the generated flat sequence to a tuple. CopyAttention (Cui et al., 2018) is a neural generator trained over bootstrapped data generated from OpenIE-4 extractions on a large corpus. During inference, it uses beam search to get the predicted extractions. It uses a fixed-size beam, limiting it to output a constant number of extractions per sentence. Moreover, our analysis shows that CopyAttention extractions severely lack in diversity, as illustrated in Table 1.

Sun et al. (2018) propose the *Logician* model, a restricted sequence generation model for extracting tuples from Chinese text. Logician relies on coverage attention and gated-dependency attention, a language-specific heuristic for Chinese. Using coverage attention, the model also tackles generation of multiple extractions while being globally-aware. We compare against Logician's coverage attention as one of the approaches for increasing diversity.

Sequence-labeling based models lack the ability to change the sentence structure or introduce new auxiliary words while uttering predictions. For example, they cannot extract (Trump, is the President of, US) from "US President Trump", since 'is', 'of' are not in the original sentence. On the other hand, sequence-generation models are more general and, in principle, need not suffer from these limitations.

Evaluation: All neural models have shown improvements over the traditional systems using the OIE2016 benchmark. However, recent work shows that the OIE2016 dataset is quite noisy, and that its evaluation does not penalize highly redundant extractions (Léchelle et al., 2018). In our work, we use the latest CaRB benchmark, which crowdsources a new evaluation dataset, and also provides a modified evaluation framework to downscore near-redundant extractions (Bhardwaj et al., 2019).

3 Sequential Decoding

We now describe IMoJIE, our generative approach that can output a variable number of diverse extractions per sentence. The architecture of our model is illustrated in Figure 1. At a high level, the next extraction from a sentence is best determined in context of all other tuples extracted from it so far. Hence, IMoJIE uses a decoding strategy that generates extractions in a sequential fashion, one after another, each one being aware of all the ones generated prior to it.

This kind of sequential decoding is made possible by the use of an *iterative memory*. Each of the generated extractions are added to the memory so that the next iteration of decoding has access to all of the previous extractions. We simulate this iterative memory with the help of BERT encoder, whose input includes the [*CLS*] token and original

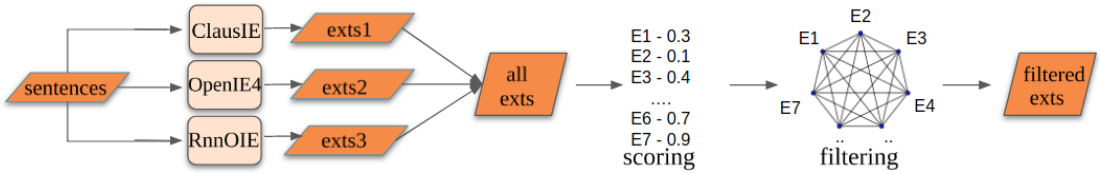

Figure 2: Ranking-Filtering subsystem for combining extractions from multiple open IE systems in an unsupervised fashion. ('Exts'=extractions.)

sentence appended with the decoded extractions so far, punctuated by the separator token [*SEP*] before each extraction.

IMOJIE uses an LSTM decoder, which is initialized with the embedding of [*CLS*] token. The contextualized-embeddings of all the word tokens are used for the Copy (Gu et al., 2016) and Attention (Bahdanau et al., 2015) modules. The decoder generates the tuple one word at a time, producing $\langle rel \rangle$ and $\langle obj \rangle$ tokens to indicate the start of relation and object respectively. The iterative process continues until the *EndOfExtractions* token is generated.

The overall process can be summarized as:

1. Pass the sentence through the Seq2Seq architecture to generate the first extraction.
2. Concatenate the generated extraction with the existing input and pass it again through the Seq2Seq architecture to generate the next extraction.
3. Repeat Step 2 until the *EndOfExtractions* token is generated.

IMOJIE is trained using a cross-entropy loss between the generated output and the gold output.

4 Aggregating Bootstrapped Data

4.1 Single Bootstrapping System

To train generative neural models for the task of OpenIE, we need a set of sentence-extraction pairs. It is ideal to curate such a training dataset via human annotation, but that is impractical, considering the scale of training data required for a neural model. We follow Cui et al. (2018), and use bootstrapping — using extractions from a pre-existing OpenIE system as 'silver'-labeled (as distinct from 'gold'-labeled) instances to train the neural model. We first order all extractions in the decreasing order of confidences output by the original system. We then construct training data in IMOJIE's input-output format, assuming that this is the order in which it should produce its extractions.

4.2 Multiple Bootstrapping Systems

Different OpenIE systems have diverse quality characteristics. For example, the human-estimated (precision, recall) of OpenIE-4 is $(\mathbf{61}, 43)$ while that of ClausIE is $(40, \mathbf{50})$. Thus, by using their combined extractions as the bootstrapping dataset, we might potentially benefit from the high precision of OpenIE-4 and high recall of ClausIE.

However, simply pooling all extractions would not work, because of the following serious hurdles.

No calibration: Confidence scores assigned by different systems are not calibrated to a comparable scale.

Redundant extractions: Beyond exact duplicates, multiple systems produce similar extractions with low marginal utility.

Wrong extractions: Pooling inevitably pollutes the silver data and can amplify incorrect instances, forcing the downstream open IE system to learn poor-quality extractions.

We solve these problems using a **Score-and-Filter** framework, shown in Figure 2.

Scoring: All systems are applied on a given sentence, and the pooled set of extractions are scored such that good (correct, informative) extractions generally achieve higher values compared to bad (incorrect) and redundant ones. In principle, this score may be estimated by the generation score from IMOJIE, trained on a single system. In practice, such a system is likely to consider extractions similar to its bootstrapping training data as good, while disregarding extractions of other systems, even though those extractions may also be of high quality. To mitigate this bias, we use an IMOJIE model, pre-trained on a *random bootstrapping dataset*. The random bootstrapping dataset is generated by picking extractions for each sentence randomly from any one of the bootstrapping systems being aggregated. We assign a score to each extraction in the pool based on the confidence value given to it by this IMOJIE (Random) model.

Filtering: We now filter this set of extractions for

redundancy. Given the set of ranked extractions in the pool, we wish to select that subset of extractions that have the best confidence scores (assigned by the random-boostrap model), while having minimum similarity to the other selected extractions.

We model this goal as the selection of an optimal subgraph from a suitably designed complete weighted graph. Each node in the graph corresponds to one extraction in the pool. Every pair of nodes (u, v) are connected by an edge. Every edge has an associated weight $R(u, v)$ signifying the similarity between the two corresponding extractions. Each node u is assigned a score $f(u)$ equal to the confidence given by the random-bootstrap model.

Given this graph $G = (V, E)$ of all pooled extractions of a sentence, we aim at selecting a subgraph $G' = (V', E')$ with $V' \subseteq V$, such that the most significant ones are selected, whereas the extractions redundant with respect to already-selected ones are discarded. Our objective is

$$\max_{G' \subseteq G} \sum_{i=1}^{|V'|} f(u_i) - \sum_{j=1}^{|V'|-1} \sum_{k=j+1}^{|V'|} R(u_j, u_k), \quad (1)$$

where u_i represents node $i \in V'$. We compute $R(u, v)$ as the ROUGE2 score between the serialized triples represented by nodes u and v. We can intuitively understand the first term as the aggregated sum of significance of all selected triples and second term as the redundancy among these triples.

If G has n nodes, we can pose the above objective as:

$$\max_{\boldsymbol{x} \in \{0,1\}^n} \boldsymbol{x}^\top \boldsymbol{f} - \boldsymbol{x}^\top \boldsymbol{R} \boldsymbol{x}, \quad (2)$$

where $\boldsymbol{f} \in \mathbb{R}^n$ representing the node scores, i.e., $f[i] = f(u_i)$, and $\boldsymbol{R} \in \mathbb{R}^{n \times n}$ is a symmetric matrix with entries $R_{j,k} = \text{ROUGE2}(u_j, u_k)$. $\boldsymbol{x}$ is the decision vector, with $x[i]$ indicating whether a particular node $u_i \in V'$ or not. This is an instance of Quadratic Boolean Programming and is NP-hard, but in our application n is modest enough that this is not a concern. We use the QPBO (Quadratic Pseudo Boolean Optimizer) solver[2] (Rother et al., 2007) to find the optimal $\boldsymbol{x}^*$ and recover V'.

5 Experimental Setup

5.1 Training Data Construction

We obtain our training sentences by scraping Wikipedia, because Wikipedia is a comprehensive source of informative text from diverse domains, rich in entities and relations. Using sentences from Wikipedia ensures that our model is not biased towards data from any single domain.

We run OpenIE-4[3], ClausIE[4] and RnnOIE[5] on these sentences to generate a set of OpenIE tuples for every sentence, which are then ranked and filtered using our Score-and-Filter technique. These tuples are further processed to generate training instances in IMoJIE's input-output format.

Each sentence contributes to multiple (input, output) pairs for the IMoJIE model. The first training instance contains the sentence itself as input and the first tuple as output. For example, ("I ate an apple and an orange.", "I; ate; an apple"). The next training instance, contains the sentence concatenated with previous tuple as input and the next tuple as output ("I ate an apple and an orange. [SEP] I; ate; an apple", "I; ate; an orange"). The final training instance generated from this sentence includes all the extractions appended to the sentence as input and *EndOfExtractions* token as the output. Every sentence gives the seq2seq learner one training instance more than the number of tuples.

While forming these training instances, the tuples are considered in decreasing order of their confidence scores. If some OpenIE system does not provide confidence scores for extracted tuples, then the output order of the tuples may be used.

5.2 Dataset and Evaluation Metrics

We use the CaRB data and evaluation framework (Bhardwaj et al., 2019) to evaluate the systems[6] at different confidence thresholds, yielding a precision-recall curve. We identify three important summary metrics from the P-R curve.

Optimal F1: We find the point in the P-R curve corresponding to the largest F1 value and report that. This is the operating point for getting extractions with the best precision-recall trade-off.

AUC: This is the area under the P-R curve. This metric is useful when the downstream application can use the confidence value of the extraction.

Last F1: This is the F1 score computed at the point of zero confidence. This is of importance when we cannot compute the optimal threshold, due to lack of any gold-extractions for the domain.

[2] https://pypi.org/project/thinqpbo/

[3] https://github.com/knowitall/openie

[4] https://www.mpi-inf.mpg.de/clausie

[5] https://github.com/gabrielStanovsky/supervised-oie

[6] Our reported CaRB scores for OpenIE-4 and OpenIE-5 are slightly different from those reported by Bhardwaj et al. (2019). The authors of CaRB have verified our values.

System	Metric		
	Opt. F1	AUC	Last F1
Stanford-IE	23	13.4	22.9
OllIE	41.1	22.5	40.9
PropS	31.9	12.6	31.8
MinIE	41.9	-*	41.9
OpenIE-4	51.6	29.5	51.5
OpenIE-5	48.5	25.7	48.5
ClausIE	45.1	22.4	45.1
CopyAttention	35.4	20.4	32.8
RNN-OIE	49.2	26.5	49.2
Sense-OIE	17.2	-*	17.2
Span-OIE	47.9	-*	47.9
CopyAttention + BERT	51.6	32.8	49.6
IMoJIE	**53.5**	**33.3**	**53.3**

Table 3: Comparison of various OpenIE systems - non-neural, neural and proposed models. (*) Cannot compute AUC as Sense-OIE, MinIE do not emit confidence values for extractions and released code for Span-OIE does not provision calculation of confidence values. In these cases, we report the Last F1 as the Opt. F1

Many downstream applications of OpenIE, such as text comprehension (Stanovsky et al., 2015) and sentence similarity estimation (Christensen et al., 2014), use *all* the extractions output by the OpenIE system. Last F1 is an important measure for such applications.

5.3 Comparison Systems

We compare IMoJIE against several non-neural baselines, including Stanford-IE, OpenIE-4, OpenIE-5, ClausIE, PropS, MinIE, and OLLIE. We also compare against the sequence labeling baselines of RnnOIE, SenseOIE, and the span selection baseline of SpanOIE. Probably the most closely related baseline to us is the neural generation baseline of CopyAttention. To increase CopyAttention's diversity, we compare against an English version of Logician, which adds coverage attention to a single-decoder model that emits all extractions one after another. We also compare against CopyAttention augmented with diverse beam search (Vijayakumar et al., 2018) — it adds a diversity term to the loss function so that new beams have smaller redundancy with respect to all previous beams.

Finally, because our model is based on BERT, we reimplement CopyAttention with a BERT encoder — this forms a very strong baseline for our task.

5.4 Implementation

We implement IMoJIE in the AllenNLP framework[7] (Gardner et al., 2018) using Pytorch 1.2. We use "BERT-small" model for faster training. Other hyper-parameters include learning rate for BERT, set to 2×10^{-5}, and learning rate, hidden dimension, and word embedding dimension of the decoder LSTM, set to $(10^{-3}, 256, 100)$, respectively.

Since the model or code of CopyAttention (Cui et al., 2018) were not available, we implemented it ourselves. Our implementation closely matches their reported scores, achieving (F1, AUC) of (56.4, 47.7) on the OIE2016 benchmark.

[7] https://github.com/allenai/allennlp

System	Metric		
	Opt. F1	AUC	Last F1
CopyAttention	35.4	20.4	32.8
CoverageAttention	41.8	22.1	41.8
CoverageAttention+BERT	47.9	27.9	47.9
Diverse Beam Search	46.1	26.1	39.6
IMoJIE (w/o BERT)	37.9	19.1	36.6
IMoJIE	**53.2**	**33.1**	**52.4**

Table 4: Models to solve the redundancy issue prevalent in Generative Neural OpenIE systems. All systems are bootstrapped on OpenIE-4.

Bootstrapping Systems	Metric		
	Opt. F1	AUC	Last F1
ClausIE	49.2	31.4	45.5
RnnOIE	51.3	31.1	50.8
OpenIE-4	53.2	33.1	52.4
OpenIE-4+ClausIE	51.5	32.5	47.1
OpenIE-4+RnnOIE	53.1	32.1	53.0
ClausIE+RnnOIE	50.9	32.2	49.8
All	**53.5**	**33.3**	**53.3**

Table 5: IMoJIE trained with different combinations of bootstrapping data from 3 systems - OpenIE-4, ClausIE, RNNOIE. Graph filtering is not used over single datasets.

6 Results and Analysis

6.1 Performance of Existing Systems

How well do the neural systems perform as compared to the rule-based systems?

Using CaRB evaluation, we find that, contrary to previous papers, neural OpenIE systems are not necessarily better than prior non-neural systems (Table 3). Among the systems under consideration, the best non-neural system reached Last F1 of 51.5, whereas the best existing neural model could only reach 49.2. Deeper analysis reveals that CopyAttention produces redundant extractions conveying nearly the same information, which CaRB effectively penalizes. RnnOIE performs much better, however suffers due to its lack of generating auxiliary verbs and implied prepositions. Example, it can only generate (Trump; President; US) instead of (Trump; is President of; US) from the sentence

Filtering	Metric		
	Opt. F1	AUC	Last F1
None	49.7	**34.5**	37.4
Extraction-based	46	29.2	44.9
Sentence-based	49.5	32.7	48.6
Score-And-Filter	**53.5**	33.3	**53.3**

Table 6: Performance of IMOJIE on aggregated dataset **OpenIE-4+ClausIE+RnnOIE**, with different filtering techniques. For comparison, SenseOIE trained on multiple system extractions gives an F1 of 17.2 on CaRB.

"US President Trump...". Moreover, it is trained only on limited number of pseudo-gold extractions, generated by Michael et al. (2018), which does not take advantage of boostrapping techniques.

6.2 Performance of IMOJIE

How does IMOJIE *perform compared to the previous neural and rule-based systems?*

In comparison with existing neural and non-neural systems, IMOJIE trained on aggregated bootstrapped data performs the best. It outperforms OpenIE-4, the best existing OpenIE system, by 1.9 F1 pts, 3.8 pts of AUC, and 1.8 pts of Last-F1. Qualitatively, we find that it makes fewer mistakes than OpenIE-4, probably because OpenIE-4 accumulates errors from upstream parsing modules (see Table 2).

IMOJIE outperforms CopyAttention by large margins – about 18 Optimal F1 pts and 13 AUC pts. Qualitatively, it outputs non-redundant extractions through the use of its iterative memory (see Table 1), and a variable number of extractions owing to the *EndofExtractions* token. It also outperforms CopyAttention with BERT, which is a very strong baseline, by 1.9 Opt. F1 pts, 0.5 AUC and 3.7 Last F1 pts. IMOJIE consistently outperforms CopyAttention with BERT over different bootstrapping datasets (see Table 8).

Figure 3 shows that the precision-recall curve of IMOJIE is consistently above that of existing OpenIE systems, emphasizing that IMOJIE is consistently better than them across the different confidence thresholds. We do find that CopyAttention+BERT outputs slightly higher recall at a significant loss of precision (due to its beam search with constant size), which gives it some benefit in the overall AUC. CaRB evaluation of SpanOIE[8] results in (precision, recall, F1) of (58.9, 40.3, 47.9). SpanOIE sources its training data only from OpenIE-4. In order to be fair, we compare it against

[8] https://github.com/zhanjunlang/Span_OIE

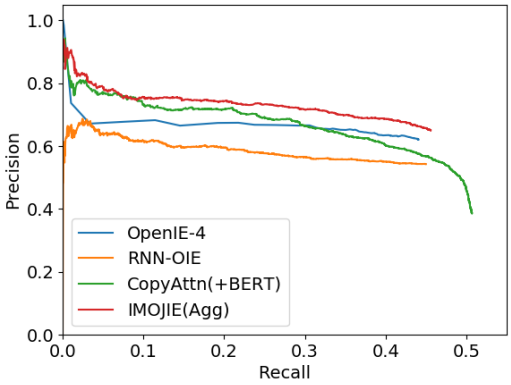

Figure 3: Precision-Recall curve of OpenIE Systems.

IMOJIE trained only on data from OpenIE-4 which evaluates to (60.4, 46.3, 52.4). Hence, IMOJIE outperforms SpanOIE, both in precision and recall.

Attention is typically used to make the model focus on words which are considered important for the task. But the IMOJIE model successfully uses attention to *forget* certain words, those which are already covered. Consider, the sentence "He served as the first prime minister of Australia and became a founding justice of the High Court of Australia". Given the previous extraction (He; served; as the first prime minister of Australia), the BERTs attention layers figure out that the words 'prime' and 'minister' have already been covered, and thus push the decoder to prioritize 'founding' and 'justice'. Appendix D analyzes the attention patterns of the model when generating the intermediate extraction in the above example and shows that IMOJIE gives less attention to already covered words.

6.3 Redundancy

What is the extent of redundancy in IMOJIE *when compared to earlier OpenIE systems?*

We also investigate other approaches to reduce redundancy in CopyAttention, such as Logician's coverage attention (with both an LSTM and a BERT encoder) as well as diverse beam search. Table 4 reports that both these approaches indeed make significant improvements on top of CopyAttention scores. In particular, qualitative analysis of diverse beam search output reveals that the model gives out different words in different tuples in an effort to be diverse, without considering their correctness. Moreover, since this model uses beam search, it still outputs a fixed number of tuples.

This analysis naturally suggested the IMOJIE (w/o BERT) model — an IMOJIE variation

Extractions	Metric		
	MNO	IOU	#Tuples
CopyAttention+BERT	2.805	0.463	3159
IMOJIE	**1.282**	**0.208**	**1620**
Gold	1.927	0.31	2650

Table 7: Measuring redundancy of extractions. MNO stands for Mean Number of Occurrences. IOU stands for Intersection over Union.

that uses an LSTM encoder instead of BERT. Unfortunately, IMOJIE (w/o BERT) is behind the CopyAttention baseline by 12.1 pts in AUC and 4.4 pts in Last F1. We hypothesize that this is because the LSTM encoder is unable to learn how to capture *inter-fact dependencies* adequately — the input sequences are too long for effectively training LSTMs.

This explains our use of Transformers (BERT) instead of the LSTM encoder to obtain the final form of IMOJIE. With a better encoder, IMOJIE is able to perform up to its potential, giving an improvement of **(17.8, 12.7, 19.6)** pts in (Optimal F1, AUC, Last F1) over existing seq2seq OpenIE systems.

We further measure two quantifiable metrics of redundancy:

Mean Number of Occurrences (MNO): The average number of tuples, every output word appears in.

Intersection Over Union (IOU): Cardinality of intersection over cardinality of union of words in the two tuples, averaged over all pairs of tuples.

These measures were calculated after removing stop words from tuples. Higher value of these measures suggest higher redundancy among the extractions. IMOJIE is significantly better than CopyAttention+BERT, the strongest baseline, on both these measures (Table 7). Interestingly, IMOJIE has a lower redundancy than even the gold triples; this is due to imperfect recall.

6.4 The Value of Iterative Memory

To what extent does the IMOJIE *style of generating tuples improve performance, over and above the use of BERT?*

We add BERT to CopyAttention model to generate another baseline for a fair comparison against the IMOJIE model. When trained only on OpenIE-4, IMOJIE continues to outperform CopyAttention+BERT baseline by (1.6, 0.3, 2.8) pts in (Optimal F1, AUC, Last F1), which provides strong evidence that the improvements are not solely by virtue of using a better encoder. We repeat this experiment over different (single) bootstrapping datasets. Table 8 depicts that IMOJIE consistently outperforms CopyAttention+BERT model.

We also note that the order in which the extractions are presented to the model (during training) is indeed important. On training IMOJIE using a randomized-order of extractions, we find a decrease of 1.6 pts in AUC (averaged over 3 runs).

6.5 The value of Score-and-Filter

To what extent does the scoring and filtering approach lead to improvement in performance?

IMOJIE aggregates extractions from multiple systems through the scoring and filtering approach. It uses extractions from OpenIE-4 (190K), ClausIE (202K) and RnnOIE (230K) to generate a set of 215K tuples. Table 6 reports that IMOJIE does not perform well when this aggregation mechanism is turned off. We also try two supervised approaches to aggregation, by utilizing the gold extractions from CaRB's dev set.

- **Extraction Filtering**: For every sentence-tuple pair, we use a binary classifier that decides whether or not to consider that extraction. The input features of the classifier are the *[CLS]*-embeddings generated from BERT after processing the concatenated sentence and extraction. The classifier is trained over tuples from CaRB's dev set.
- **Sentence Filtering**: We use an IMOJIE model (bootstrapped over OpenIE-4), to score all the tuples. Then, a Multilayer Perceptron (MLP) predicts a confidence threshold to perform the filtering. Only extractions with scores greater than this threshold will be considered. The input features of the MLP include the length of sentence, IMOJIE (OpenIE-4) scores, and GPT (Radford et al., 2018) scores of each extraction. This MLP is trained over sentences from CaRB's dev set and the gold optimal confidence threshold calculated by CaRB.

We observe that the Extraction, Sentence Filtering are better than no filtering by by 7.5, 11.2 pts in Last F1, but worse at Opt. F1 and AUC. We hypothesise that this is because the training data for the MLP (640 sentences in CaRB's dev set), is not sufficient and the features given to it are not sufficiently discriminative. Thereby, we see the value of our unsupervised Score-and-Filter that improves

System	Bootstrapping System			
	OpenIE-4	OpenIE-5	ClausIE	RnnOIE
Base	50.7, 29, 50.7	47.4, 25.1, 47.4	45.1, 22.4, 45.1	49.2, 26.5, 49.2
CopyAttention+BERT	51.6, 32.8, 49.6	48.7, **29.4**, 48.0	47.4, 30.2, 43.6	47.9, 30.6, 41.1
IMoJIE	**53.2**, **33.1**, **52.4**	**48.8**, 27.9, **48.7**	**49.2**, **31.4**, **45.5**	**51.3**, **31.1**, **50.8**

Table 8: Evaluating models trained with different bootstrapping systems.

the performance of IMoJIE by (3.8, 15.9) pts in (Optimal F1, Last F1). The 1.2 pt decrease in AUC is due to the fact that the IMoJIE (no filtering) produces many low-precision extractions, that inflates the AUC.

Table 5 suggests that the model trained on all three aggregated datasets perform better than models trained on any of the single/doubly-aggregated datasets. Directly applying the Score-and-Filter method on the test-extractions of RnnOIE+OpenIE-4+ClausIE gives (Optimal F1, AUC, Last F1) of (50.1, 32.4, 49.8). This shows that training the model on the aggregated dataset is important.

Computational Cost: The training times for CopyAttention+BERT, IMoJIE (OpenIE-4) and IMoJIE (including the time taken for Score-and-Filter) are 5 hrs, 13 hrs and 30 hrs respectively. This shows that the performance improvements come with an increased computational cost, and we leave it to future work to improve the computational efficiency of these models.

7 Error Analysis

We randomly selected 50 sentences from the CaRB validation set. We consider only sentences where at least one of its extractions shows the error. We identified four major phenomena contributing to errors in the IMoJIE model:

(1) **Missing information:** 66% of the sentences have at least one of the relations or arguments or both missing in predicted extractions, which are present in gold extractions. This leads to incomplete information.

(2) **Incorrect demarcation:** Extractions in 60% of the sentences have the separator between relation and argument identified at the wrong place.

(3) **Missing conjunction splitting:** In 32% of the sentences, our system fails to separate out extractions by splitting a conjunction. E.g., in the sentence "US 258 and NC 122 parallel the river north ...", IMoJIE predicts just one extraction (US 258 and NC 122; parallel; ...) as opposed to two separate extractions (US 258; parallel; ...) and (NC 122; parallel; ...) as in gold.

(4) **Grammatically incorrect extractions:** 38% sentences have a grammatically incorrect extraction (when serialized into a sentence).

Additionally, we observe 12% sentences still suffering from **redundant** extractions and 4% **miscellaneous** errors.

8 Conclusions and Discussion

We propose IMoJIE for the task of OpenIE. IMoJIE significantly improves upon the existing OpenIE systems in all three metrics, Optimal F1, AUC, and Last F1, establishing a new State Of the Art system. Unlike existing neural OpenIE systems, IMoJIE produces non-redundant as well as a variable number of OpenIE tuples depending on the sentence, by iteratively generating them conditioned on the previous tuples. Additionally, we also contribute a novel technique to combine multiple OpenIE datasets to create a high-quality dataset in a completely unsupervised manner. We release the training data, code, and the pretrained models.[9]

IMoJIE presents a novel way of using attention for text generation. Bahdanau et al. (2015) showed that attending over the input words is important for text generation. See et al. (2017) showed that using a coverage loss to track the attention over the decoded words improves the quality of the generated output. We add to this narrative by showing that deep inter-attention between the input and the partially-decoded words (achieved by adding previous output in the input) creates a better representation for iterative generation of triples. This general observation may be of independent interest beyond OpenIE, such as in text summarization.

Acknowledgements

IIT Delhi authors are supported by IBM AI Horizons Network grant, an IBM SUR award, grants by Google, Bloomberg and 1MG, and a Visvesvaraya faculty award by Govt. of India. We thank IIT Delhi HPC facility for compute resources. Soumen is supported by grants from IBM and Amazon. We would like to thank Arpita Roy for sharing the extractions of SenseOIE with us.

[9] https://github.com/dair-iitd/imojie

References

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging Linguistic Structure for Open Domain Information Extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL), 2015*, pages 344–354.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *International Conference on Learning Representations (ICLR), 2015*.

Niranjan Balasubramanian, Stephen Soderland, Mausam, and Oren Etzioni. 2013. Generating Coherent Event Schemas at Scale. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2013*, pages 1721–1731.

Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *International Joint Conference on Artificial Intelligence (IJCAI), 2007*, volume 7, pages 2670–2676.

Sangnie Bhardwaj, Samarth Aggarwal, and Mausam. 2019. CaRB: A Crowdsourced Benchmark for OpenIE. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019*, pages 6263–6268.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2011. An analysis of open information extraction based on semantic role labeling. In *Proceedings of the sixth international conference on Knowledge capture*, pages 113–120. ACM.

Janara Christensen, Stephen Soderland, Gagan Bansal, et al. 2014. Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 902–912.

Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural open information extraction. In *Proceedings of Association for Computational Linguistics (ACL), 2018*, pages 407–413.

Luciano Del Corro and Rainer Gemulla. 2013. ClausIE: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web (WWW), 2013*, pages 355–366. ACM.

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

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. 2011. Open Information Extraction: The Second Generation. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 3–10. IJCAI/AAAI.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying Relations for Open Information Extraction. In *Proceedings of the Conference of Empirical Methods in Natural Language Processing (EMNLP '11)*, Edinburgh, Scotland, UK.

Angela Fan, Claire Gardent, Chloé Braud, and Antoine Bordes. 2019. Using Local Knowledge Graph Construction to Scale Seq2Seq Models to Multi-Document Inputs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A Deep Semantic Natural Language Processing Platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6.

Kiril Gashteovski, Rainer Gemulla, and Luciano del Corro. 2017. MinIE: minimizing facts in open information extraction. In *Association for Computational Linguistics (ACL), 2017*.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *Proceedings of Association for Computational Linguistics (ACL), 2016*. Association for Computational Linguistics.

Zhengbao Jiang, Pengcheng Yin, and Graham Neubig. 2019. Improving Open Information Extraction via Iterative Rank-Aware Learning. In *Proceedings of the Association for Computational Linguistics (ACL), 2019*.

William Léchelle, Fabrizio Gotti, and Philippe Langlais. 2018. Wire57 : A fine-grained benchmark for open information extraction. In *LAW@ACL*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems (NIPS), 2019*, pages 13–23.

Mausam. 2016. Open information extraction systems and downstream applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI), 2016*, pages 4074–4077. AAAI Press.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics.

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 (NAACL-HLT), 2018, Volume 2 (Short Papers)*, pages 560–568.

Harinder Pal and Mausam. 2016. Demonyms and compound relational nouns in nominal OpenIE. In *Proceedings of the 5th Workshop on Automated Knowledge Base Construction*, pages 35–39.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training (2018).

Carsten Rother, Vladimir Kolmogorov, Victor S. Lempitsky, and Martin Szummer. 2007. Optimizing Binary MRFs via Extended Roof Duality. *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Arpita Roy, Youngja Park, Taesung Lee, and Shimei Pan. 2019. Supervising Unsupervised Open Information Extraction Models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 728–737.

Swarnadeep Saha, Harinder Pal, and Mausam. 2017. Bootstrapping for numerical OpenIE. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 317–323. Association for Computational Linguistics.

Swarnadeep Saha et al. 2018. Open information extraction from conjunctive sentences. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2288–2299.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Association for Computational Linguistics (ACL), 2017*.

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 (EMNLP)*, Austin, Texas. Association for Computational Linguistics.

Gabriel Stanovsky, Jessica Ficler, Ido Dagan, and Yoav Goldberg. 2016. Getting more out of syntax with PropS. *CoRR*, abs/1603.01648.

Gabriel Stanovsky, Mausam, and Ido Dagan. 2015. OpenIE as an intermediate structure for semantic tasks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 303–308.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised Open Information Extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1 (Long Papers)*, pages 885–895.

Mingming Sun, Xu Li, Xin Wang, Miao Fan, Yue Feng, and Ping Li. 2018. Logician: A unified end-to-end neural approach for open-domain information extraction. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 556–564.

Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. In *Proceedings of Association for Computational Linguistics (ACL), 2019*.

Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2018. Diverse Beam Search for Improved Description of Complex Scenes. In *AAAI Conference on Artificial Intelligence, 2018*, pages 7371–7379.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning (ICML), 2015*, pages 2048–2057.

Junlang Zhan and Hai Zhao. 2020. Span Model for Open Information Extraction on Accurate Corpus. In *AAAI Conference on Artificial Intelligence, 2020*, pages 5388–5399.

IMOJIE: Iterative Memory-Based Joint Open Information Extraction (Supplementary Material)

A Performance with varying sentence lengths

In this experiment, we measure the performance of baseline and our models by testing on sentences of varying lengths. We partition the original CaRB test data into 6 datasets with sentences of lengths (9-16 words), (17-24 words), (25-32 words), (33-40 words), (41-48 words) and (49-62 words) respectively. Note that the minimum and maximum sentence lengths are 9 and 62 respectively. We measure the Optimal F1 score of both Copy Attention + BERT and IMOJIE (Bootstrapped on OpenIE-4) on these partitions as depicted in Figure 4.
We observe that the performance deteriorates with increasing sentence length which is expected as well. Also, for each of the partitions, IMOJIE marginally performs better as compared to Copy Attention + BERT.

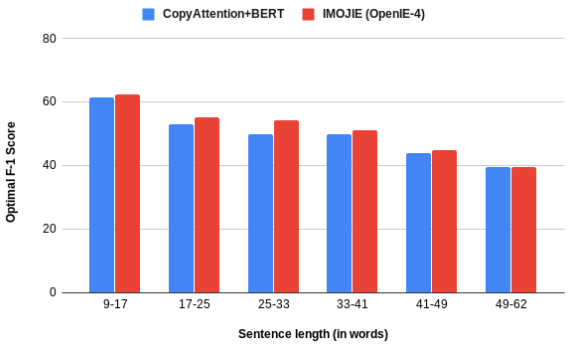

Figure 4: Measuring performance with varying input sentence lengths

B Measuring Performance on Varying Beam Size

We perform inference of the CopyAttention with BERT model on CaRB test set with beam sizes of 1, 3, 5, 7, and 11. We observe in Figure 5 that AUC increases with increasing beam size. A system can surge its AUC by adding several low confidence tuples to its predicted set of tuples. This adds low precision - high recall points to the Precision-Recall curve of the system leading to higher AUC.
On the other hand, Last F1 experiences a drop at very high beam sizes, thereby capturing the decline in performance. Optimal F1 saturates at high beam sizes since its calculation ignores the extractions below the optimal confidence threshold.
This analysis also shows the importance of using Last F1 as a metric for measuring the performance of OpenIE systems.

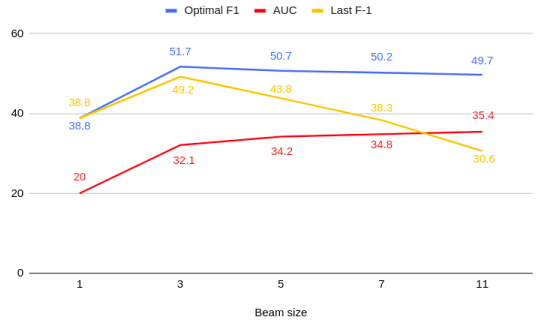

Figure 5: Measuring performance of CopyAttention with BERT model upon changing the beam size

C Evaluation on other datasets

We use sentences from other benchmarks with the CaRB evaluation policy and we find similar improvements, as shown in Table 9. IMOJIE consistently outperforms our strongest baseline, CopyAttention with BERT, over different test sets. This confirms that IMOJIE is domain agnostic.

D Visualizing Attention

Attention has been used in a wide variety of settings to help the model learn to focus on important things (Bahdanau et al., 2015; Xu et al., 2015; Lu et al., 2019). However, the IMOJIE model is able to use attention to understand which words have already been generated, to focus on remaining words. In order to understand how the model achieves this, we visualize the learnt attention weights. There are two attention weights of importance, the learnt attention inside the BERT encoder and the attention between the decoder and encoder. We use BertViz (Vig, 2019) to visualize the attention inside BERT.

We consider the following sentence as the running example - "he served as the first prime minister of australia and became a founding justice of the high court of australia". We visualize the attention after producing the first extraction - "he; served; as the first prime minister of australia". Intuitively, we understand that the model must focus on the words "founding" and "justice" in order to generate the next extraction - "he; became; a founding justice of the high court of australia". In Figure 8 and Figure

Model	Dataset		
	Wire57	Penn	Web
CopyAttention + BERT	45.60, **27.70**, 39.70	18.20, 7.9, 12.40	30.10, **18.00**, 14.60
IMoJIE	**46.20**, 26.60, **46.20**	**20.20**, **8.70**, **15.50**	**30.40**, 15.50, **26.40**

Table 9: Evaluation on other datasets with the CaRB evaluation strategy

9 (where the left-hand column contains the words which are used to attend while right-hand column contains the words which are attended over), we see that the words “prime” and “minister” of the original sentence have high attention over the same words in the first extraction. But the attention for “founding” and “justice” are limited to the original sentence.

Based on these patterns, the decoder is able to give a high attention to the words “founding” and “justice” (as shown in Figure 10), in-order to successfully generate the second extraction ”he; became; a founding justice of the high court of australia”.

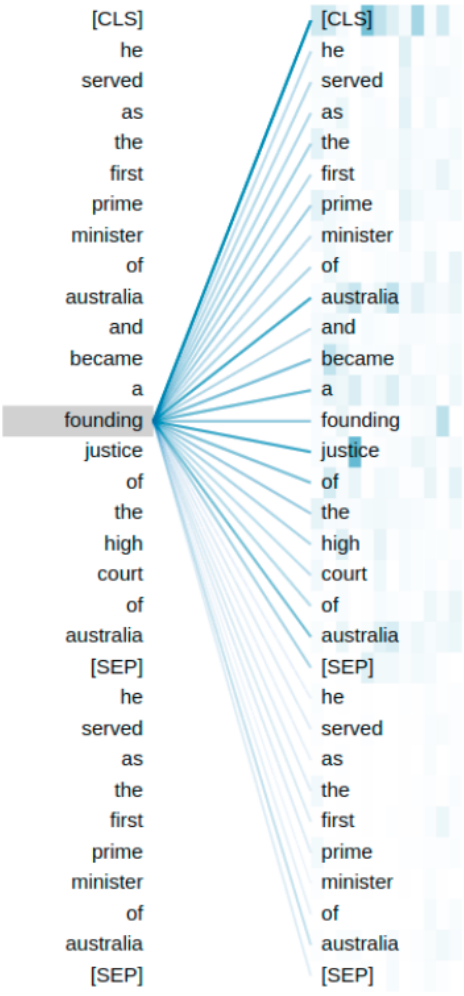

Figure 6: BERT attention for the word ‘founding’

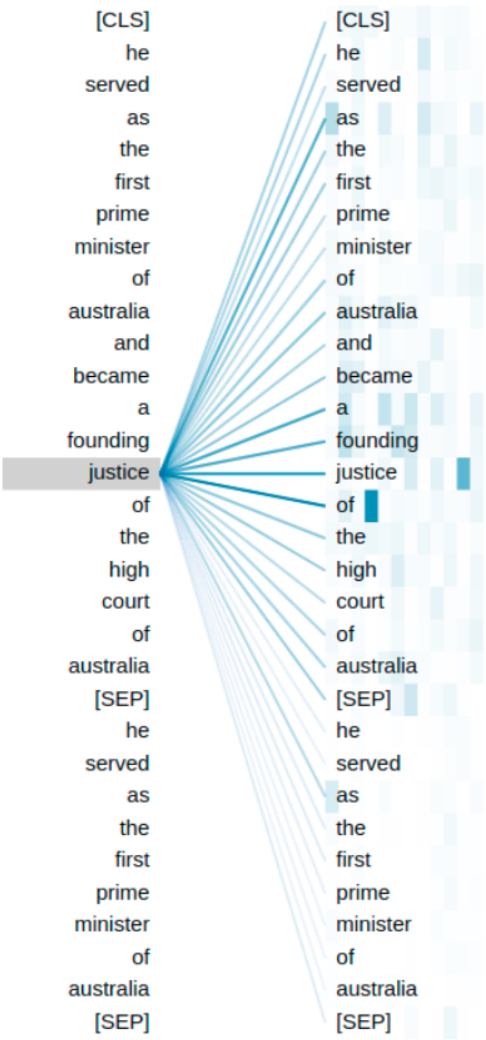

Figure 7: BERT attention for the word ‘justice’

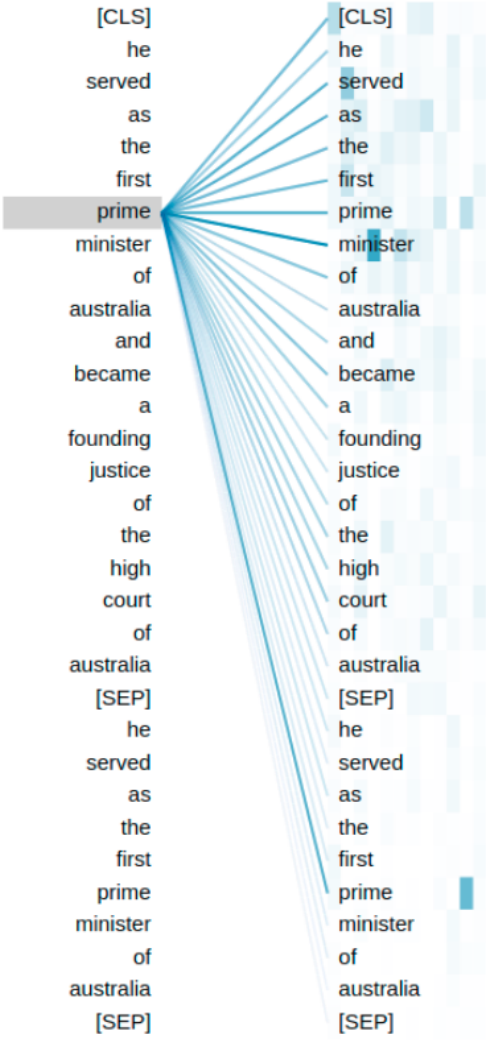

Figure 8: BERT attention for the word ‘prime’

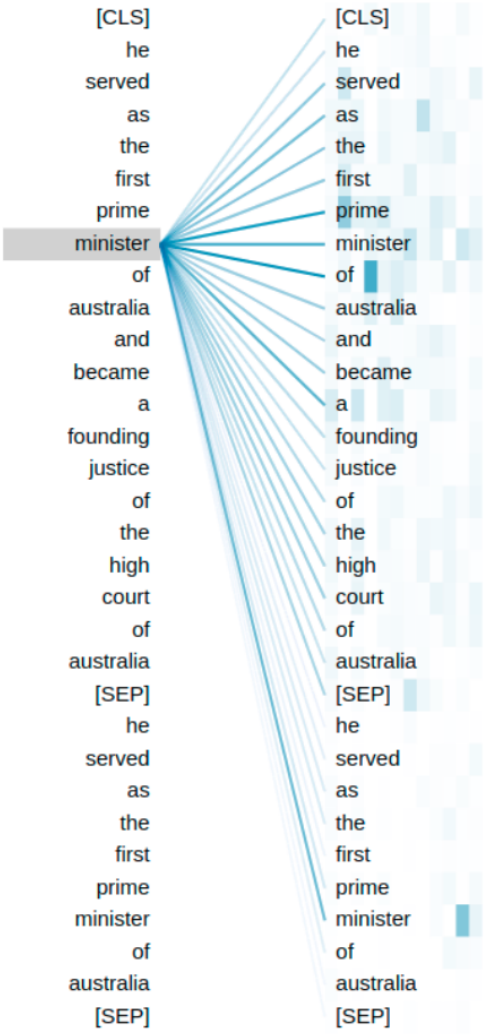

Figure 9: BERT attention for the word ‘minister’

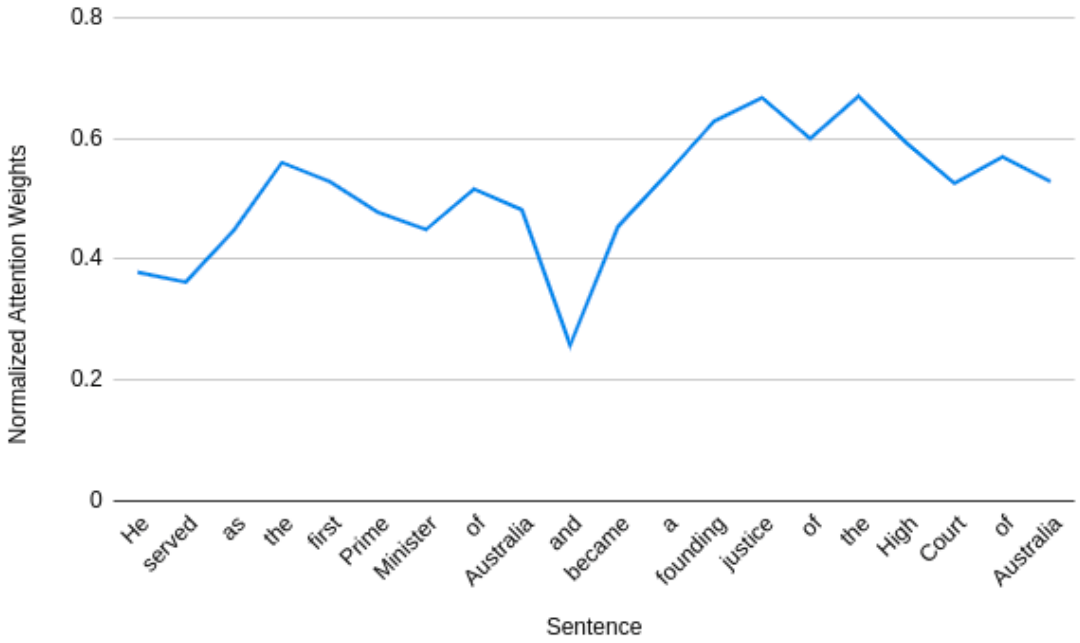

Figure 10: Attention weights for the decoder