Assessing The Factual Accuracy of Generated Text

Ben Goodrich*
Vinay Rao*
Peter J. Liu
Mohammad Saleh
bgoodrich@google.com
vinaysrao@google.com
peterjliu@google.com
msaleh@google.com
Google Brain

ABSTRACT

We propose a model-based metric to estimate the factual accuracy of generated text that is complementary to typical scoring schemes like ROUGE (Recall-Oriented Understudy for Gisting Evaluation) and BLEU (Bilingual Evaluation Understudy). We introduce and release a new large-scale dataset based on Wikipedia and Wikidata to train relation classifiers and end-to-end fact extraction models. The end-to-end models are shown to be able to extract complete sets of facts from datasets with full pages of text. We then analyse multiple models that estimate factual accuracy on a Wikipedia text summarization task, and show their efficacy compared to ROUGE and other model-free variants by conducting a human evaluation study.

KEYWORDS

datasets, neural networks, fact extraction, deep learning, metric, end-to-end

ACM Reference Format:
Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing The Factual Accuracy of Generated Text. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3292500.3330955

1 INTRODUCTION

Recently, there has been wide empirical success in text summarization [15, 21, 27], machine translation [1, 36, 39], dialogue response generation [12, 28, 29], and other text generation tasks. For evaluation, these models generally rely on metrics like ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [13], BLEU (Bilingual Evaluation Understudy) [23] and perplexity [3] that measure locally constrained n-gram overlap. In this paper, we propose an automatic metric for evaluating the factual accuracy of generated text.

A fact f is defined to be a relation tuple *(subject, relation, object)*, where *subject* has a binary *relation* to *object* and can be assumed to have been inferred from text or a knowledge base, e.g. *Barack Hussein Obama II (born August 4, 1961) is an American politician who served as the 44th President of the United States from January 20, 2009 to January 20, 2017* implies a set of facts such as *(Barack Obama, president of, United States), (Barack Obama, born on, August 4 1961).*

In this paper, we limit our scope to the task of evaluating text summarization. To evaluate a text summarization model, we compare the ground-truth summary text, T and the generated summary, G. Let $f_t, f_g \in F$, and $F_T, F_G \subset F$ where F is a set of relation tuples.

$$F_T = \{f_t \mid f_t \textit{ is inferred from ground-truth } T\}$$
$$F_G = \{f_g \mid f_g \textit{ is inferred from generated-text } G\}$$

The models used in the metric we propose do not make use of world knowledge (e.g. knowledge base) during inference, and to account for that we filter F_T and F_G by only considering claims made in G that can either be verified or refuted by statements in T. Concretely, if $f_t = (subj_t, rel_t, obj_t) \in F_T$ and $f_g = (subj_g, rel_g, obj_g) \in F_G$

$$F_{T'} = \{f_t \mid \exists f_g \text{ and } subj_t = subj_g, rel_t = rel_g\}$$
$$F_{G'} = \{f_g \mid \exists f_t \text{ and } subj_g = subj_t, rel_g = rel_t\}$$

We can then define factual accuracy $fact_{acc}$ as the *precision* between $F_{T'}$ and $F_{G'}$.

$$fact_{acc} = \frac{|F_{T'} \cap F_{G'}|}{|F_{G'}|} \tag{1}$$

For example, consider ground-truth summary T: *Brad Pitt was born in 1963* and generated summary G: *Brad Pitt was born in 1961.* Then, F_T = *{(Brad Pitt, born-in, 1963)}*, F_G = *{(Brad Pitt, born-in, 1961)}*. The metric $fact_{acc} = 0$ indicates there is no factual consistency between the two summaries, whereas another metric like ROUGE-1 (1-gram overlap) measures 0.83. A real example is highlighted in Table 1 where the summarization model commits such a mistake. It is important to be able to measure these mistakes accurately to aid in training factually accurate summarization models.

Extracting fact tuples from text has been previously studied in methods like OpenIE (Open Information Extraction) [2]. OpenIE extracts triplets with an unspecified schema, and the relation is usually the text linking the two entities. However, it does not leverage information from a knowledge base and leads to outputs that are hard to compare. For example, *Person was born in that town* ⇒

*Both authors contributed equally to this research.

KDD '19, August 4–8, 2019, Anchorage, AK, USA

ACM ISBN 978-1-4503-6201-6/19/08.
https://doi.org/10.1145/3292500.3330955

Target	Peter Duryea (July 14, 1939 – March 24, 2013) was an American actor. He is best known for appearing in a pilot episode of Star Trek: The Original Series, "The Cage" (1964), most of which was reused in "The Menagerie" (1966), as Lieutenant Tyler. His father, Dan Duryea (1907 – 1968), was also an actor.
Output	Peter Duryea (April 23, 1907 – March 24, 2013) was an American actor. He is best known for his role as Lt. Jose Tyler in the original Star Trek pilot, "The Cage"

Table 1: Example of factual inaccuracy noted in a summarization model [15]. In this example, the summarization model uses the subject (Peter Duryea)'s father, Dan Duryea's birthdate.

(Person, born in, town). But *That town is the birthplace of Person* $\Rightarrow$ *(Town, is the birthplace of, Person)*.

We standardize comparison by studying structured approaches to relation tuple extraction where the schema is fixed. We compare two approaches for fact extraction. One is a two-step process that first involves recognizing all the named entities in a sentence, and then classifying the relation for every pair of entities in the sentence [14, 32]. Our other approach is to use an end-to-end model with a Transformer-based architecture [36] that is trained to output structured fact tuples. These models are described in Section 4. We create a new dataset for fact extraction using distant supervision [17] on Wikipedia text by cross-referencing facts from the Wikidata knowledge base [37]. To the best of our knowledge, this dataset is bigger and contains more relations and domains than previously used datasets for relation or fact tuple extraction.

Our main contributions are:

(1) We introduce model-based metrics to analyze the factual accuracy of generated text (Sec 4). We compare them against model-free metrics listed in Sec 5.
(2) To train fact tuple extraction models, we release code (as part of the Tensor2Tensor[1] framework along with the model weights[2]) and a large dataset (Sec 3) based on Wikidata and Wikipedia at https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact.
(3) We show that a Transformer-based end-to-end fact extraction model is able to perform structured prediction of relation tuples, avoiding the need to split the process into multiple steps (named entity recognition, coreference resolution and relation classification). It is able to extract complete sets of facts from full pages of text in one pass.
(4) We conduct experiments to compare our proposed metric against human evaluation of factual accuracy of generated text (Sec 8.1) and show that model-based metrics are better correlated with human judgment when compared to traditional metrics like ROUGE.

Our models work under some limitations that are discussed in Sec 9.1, and then Sec 9.2 discusses future work and ways to make our models more robust.

2 RELATED WORK AND MOTIVATION

Many evaluation metrics have been proposed for text generation tasks like BLEU [23] and METEOR [10] for machine translation and ROUGE [13], Basic Elements [8] & Pyramid [22] for text summarization. In Steinberger and Jezek [33], the authors explain the different kinds of evaluation we can perform for summarization. They are broadly classified as extrinsic metrics that are specific to tasks (e.g. in summarizing a person, whether the date of birth has been included) and intrinsic metrics like grammaticality, coherency and non-redundancy that are based on the analysis of the summary. ROUGE, BLEU, sentence level F1 measures, etc are intrinsic content based metrics. Zhang et al. [40] and other related works study ways to estimate the trustworthiness of answers to a question. With the recent shift towards using neural abstractive methods for text summarization and other text generation tasks, we believe that it is important to assess the factual accuracy of generated text. Wiseman et al. [38] have also studied some extractive evaluative methods to assess the quality of generated text. This includes a Relation Generator, which predicts the relation between entities to assess the factual correctness of generated records. However, we introduce a much larger dataset and enable training end-to-end models that can extract fact triplets from text. We additionally perform detailed analysis of the fact extraction models.

Typical fact extraction pipelines are a multistage process consisting of part-of-speech tagging, named entity recognition [5, 7, 9] that produces entities $\{e_i\}$ and then relation classification that predicts a relation r_k for every pair of entities (e_i, e_j). OpenIE [2] predicts a relation by linking the text connecting e_i and e_j. Because it does not have a fixed schema, logical reasoning on its outputs are not possible. Mohamed et al. [20] extend this to start with a fixed schema that can grow with more training, yet retain a consistent output surface form.

In this paper, we consider fact classification models with fixed schema. This idea has been studied in many previous works including Surdeanu et al. [34], which considered datasets that have multiple relation labels for an entity pair, which each may have multiple instances in the input text. This was modeled as a graphical model over latent variables. Riedel et al. [26] treated relation extraction as reasoning with matrix-factorization, and could work with surface-form texts and knowledge-base embeddings simultaneously. However, both of these works had datasets with very few types of relations, and were shown to work over limited domains. Recently, neural networks have been used for classifying relations. Lin et al. [14] used attention over multiple instances for the same entity pair to predict relations. Sorokin and Gurevych [32] proposed to predict multiple relations in a sentence by using all the entity pairs and relation labels in the sentence as contextual input. We propose a simpler model where we classify relations between all the entity pairs in a sentence, without any additional context.

[1] https://github.com/tensorflow/tensor2tensor
[2] https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact

We also make use of our proposed dataset that is bigger, more diverse and has more relation types. Our dataset also has article-level information that can be used to train models like in Section 4.2. Since using two-step processes may be affected by compounding of errors across the models, some end-to-end approaches [18, 19] have been proposed, where the models extract entities and relations in one pass through the model. However, the method used in Miwa and Sasaki [19] required designing hand-crafted features and task-specific algorithms. Miwa and Bansal [18] has a two-phase model that first extracts entity candidates and then predicts relations based on the parsed tree-structure of the sentence. We instead propose a sequence-to-sequence model that is able to output fact tuples directly, and does not require any feature engineering.

We found that the abstractive summarization models such as those described in Liu et al. [15] may generate sentences with factual inaccuracies (e.g. incorrect month in date of birth, wrong city in the state, etc.). Cao et al. [4] found that 30% of summaries generated by a state-of-the-art summarization model contained factual inaccuracies. We found by running a large-scale experiment as described in Section 8.1, that the summarization model had factual inaccuracy rate of approximately 17%. We believe that this is because such mistakes are not heavily penalized by cross-entropy or n-gram based model losses and metrics.

As further motivation, we synthesized factually inaccurate samples by making simple corruptions to Wikipedia lead sections. We replaced mentions of dates (day and month only), locations or people with other entities of the same type in the text. For example, *Barack was born on August 4, 1961 in Honolulu. He married Michelle on October 3, 1992 in Chicago.* becomes *Barack was born on October 3, 1961 in Chicago. He married Michelle on August 4, 1992 in Honolulu.*. Table 2 shows that model-free metrics such as ROUGE and OpenIE-based tuple comparison do not reflect the decline in factual accuracy due to such corruption as much as the model-based metrics do.

Model	Accuracy
ROUGE-1	97.08
ROUGE-2	94.06
ROUGE-L	96.02
OpenIE	87.26
Binary Relation Classifier	46.75
Relation Classifier	59.30
E2E	65.44
E2E-Reduced*	57.10
Expected Accuracy**	30.97

Table 2: Factual accuracy predicted by different metrics on synthesized samples. Binary Classifier is described in Sec 4.3, Classifier in Sec 4.1 and E2E is the end-to-end model described in Sec 4.2. E2E-Reduced* is a model where sentences where no entities are detected are filtered out from the input text. The Expected Accuracy is calculated as the ratio of number of corrupted facts to the total number of facts in the article.**

3 DATASET

We create a dataset for fact extraction using distant supervision that is based entirely on the English Wikipedia corpus and the Wikidata knowledge base W_{KB} [37]. Our distant supervisor is very similar to the one proposed by Mintz et al. [17]. Although the inputs and labels for the classifier and end-to-end model are slightly different, we start by running an NER and co-reference resolution system[4] on each Wikipedia article. The topic of that article is considered as the subject e_s. The other entities e_j found in the article are considered objects. For every pair (e_s, e_j), we say they are related if there is a relation r_k such that the triplet (e_s, r_k, e_j) is found in W_{KB}. We add this triplet to a set of positive examples E_p. If no such relation exists between e_s and e_j, we add the triplet (e_s, r_0, e_j) (r_0 denotes no-relation) to a set of negative examples E_n.

4 MODEL-BASED METRICS

In this section we describe models that can extract fact tuples from text and how we use them to define the factual accuracy metric as defined in Eq 1. Given some input text X, we then extract claims made in X as fact tuples.

4.1 Named Entity Recognition (NER) + Relation Classifier

This approach consists of two steps, where we first recognize all the named entities e_i from X and then classify relations between entity pairs (e_i, e_j).

4.1.1 Named Entity Recognition. Entities are real-world objects like people, locations, organizations etc that can be identified by a proper name[3]. Entities can be identified with named-entity recognition (NER) systems like Chiu and Nichols [5], Finkel et al. [7], Lample et al. [9] that take in X and produce the set $\{e_i\}$. NER is followed by co-reference resolution[4] [6, 11, 24, 25]. Publicly available NER and co-reference systems include Stanford's CoreNLP[5] and NLTK[6].

4.1.2 Relation Classifier. For every pair (e_i, e_j), $e_i \neq e_j$ we consider all sentences S_l in X that contain both entities. The input to the classifier is then each of these sentences S_l. Because a sentence may contain multiple entities, we also add a prefix $SUBJ$ for e_i and OBJ to e_j as a hint. For example, *X = Person1 was born in City1* becomes S_l = *SUBJ{ Person1 } was born in OBJ{ City1 }*. Unlike Sorokin and Gurevych [32], our classifier does not require additional context. Let s_i be a token in the input sentence S_l after NER, and r^k denote the kth relation. Our classifier takes in input tokens s_i that are first embedded onto a latent space, and then a stack of Transformer encoder-only layers process the whole sequence. A subsequent max-pooling layer selects one of these outputs that is then converted to a probability estimate of relations by a sigmoid operation. The exact series of operations can be viewed as:

$$w_{1:n} = embed(s_{1:n})$$
$$h_{1:n} = transformer\_encoder(w_{1:n})$$
$$h_i = max_i(h_{1:n}); h_i \in \Re^k$$
$$p(r^k) = \frac{1}{1 + e^{-h_i^k}} = sigmoid(h_i^k)$$

[3] https://en.wikipedia.org/wiki/Named_entity
[4] While we use an NER and co-reference resolution system that is not available to the public, the dataset we release (Section 3) has the positions of all the recognized and resolved entities that we use for training our classifier.
[5] http://stanfordnlp.github.io/CoreNLP/coref.html
[6] https://www.nltk.org/

Figure 1a also shows the architecture of this model.

4.1.3 Dataset preparation. For every triplet f in $E_p \cup E_n$, we have sentence(s)[7] S_l in the article that may describe the relation between e_s and e_j. S_l is processed so that subject and object are prefixed with "*SUBJ*" and "*OBJ*" as a hint to the model (Section 4.1). This leads to a dataset with 2.9 million positive examples and 34 million negative examples totaling to 45GiB on disk.

4.1.4 $fact_{acc}$ with the Relation Classifier. The classifier predicts a relation r_k for each entity pair (e_i, e_j). We extract such triplets from the ground-truth T and generated text G, and use the definition from eq 1 to calculate the factual accuracy.

4.2 End-to-End Extraction

We propose an end-to-end fact extraction model to avoid compounding of errors across components in multi-stage approaches like Section 4.1 [16]. This model also does not require any feature engineering or context. The input to the model is text X of any length (sentence/paragraph/article) and the *subject* entity e_s prefixed to X. All the inputs tokens in $[e_s; X]$ are first embedded onto a latent space. A Transformer model consisting of a stack of encoder layers followed by decoder layers produces an output sequence of arbitrary length. A softmax operation is applied to every output token to define a distribution at every timestep. Figure 1b shows the architecture of this model. To encourage the model to have structured outputs, we train the model with labels that are a sequence of fact tuples. For example, if X = " *Person1 was born in Country1. He was a painter*", then the label, Y, for that input is "*Person1* ⟨t⟩ *born in* ⟨t⟩ *Country1* ⟨f⟩ *Person1* ⟨t⟩ *profession* ⟨t⟩ *painter* ⟨*end*⟩", where ⟨t⟩ separates tokens within the fact f_i and ⟨f⟩ separates facts. For prediction, we perform a beam search over all the output timesteps, and continue decoding until ⟨*end*⟩ is predicted. A length-penalty α controls the length of this prediction as in [39].

4.2.1 Dataset preparation. If the input article text is X, every triplet f_p in E_p (we ignore the negative examples for end-to-end models because no relations between entity pairs is implied by no output by the model) is appended to the article's label L. L will then contain a series of tokens that describe facts, with seperators between them. For example: $e_s\langle t\rangle r_1\langle t\rangle e_1\langle f\rangle e_s\langle t\rangle r_2\langle t\rangle e_2\langle f\rangle$... We also prepend the input text X with e_s ($[e_s; X]$) as a hint to the model for generating facts about e_s. This leads to a dataset with 2.5 million examples totaling to 1.5GiB on disk. [8]

4.2.2 $fact_{acc}$ with the End-to-End model. The End-to-End model is able to produce a sequence of fact tuples in the form, $subj_1$ ⟨t⟩ rel_1 ⟨t⟩ obj_1 ⟨f⟩ $subj_1$ ⟨t⟩ rel_2 ⟨t⟩ obj_2. It is trained to output relations from a fixed schema based on WikiData. Consider an output from this model, *Barack Obama*⟨t⟩*P69*⟨t⟩*Harvard*. *P69* denotes 'educated at'[9]. These tuples are extracted from T and G to fit into the metric defined in eq 1.

[7] There may be more than one sentence in the article that have mentions of the subject and object entity pair.

[8] This dataset is made available at https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact

[9] https://www.wikidata.org/wiki/Property:P69

4.3 NER + Binary Relation Classifier

Similar to the typical relation classifier detailed in Sec 4.1, we define a classifier that predicts whether a pair of entities (e_i, e_j) are related to each other through any relation. This allows for verifying that entities are related in both the ground-truth T and generated text G, while being flexible enough to allow for any relation types. We also note that two entities can be related to each other in multiple ways. The inputs to this model are the same as Sec 4.1, but the model is expected to output rel as

$$rel = \begin{cases} 1\text{: } e_i \text{ and } e_j \text{ are related} \\ 0\text{: otherwise} \end{cases}$$

4.3.1 Dataset preparation. Data for this model is generated with the same procedure detailed in Sec 4.1.3. The only difference is the way we define the label rel. We consider entities e_i and e_j to be related if there is a relation r_k such that (e_i, r_k, e_j) is found in W_{KB}.

4.3.2 $fact_{acc}$ with the Binary Relation Classifier. The model predicts rel for each entity pair (e_i, e_j), and we are able to extract a set of tuples of the form (e_i, rel, e_j) from both T and G. To use eq 1 to define the factual accuracy, we filter the set by considering only entity pairs (e_i, e_j) that are found in both T and G to then compare the predicted label rel between them.

5 MODEL-FREE METRICS

We describe model-free automatic metrics in this section. Unlike model-based metrics, they are not susceptible to changes in training data, and might be considered easier to interpret or understand.

5.1 ROUGE

ROUGE [13] has been used as an automatic metric to judge the quality of generated text, and has shown to correlate well with human judgment of overall linguistic quality of the text.

5.2 OpenIE

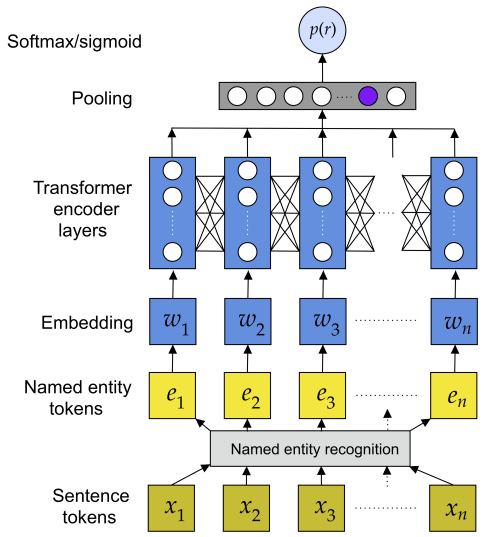
OpenIE [2] is a tool that can extract relation tuples from text, without a specified schema. We use it to extract sets of relation tuples from T and G, and then compute the precision like in eq 1.

6 MODEL EXPERIMENTS

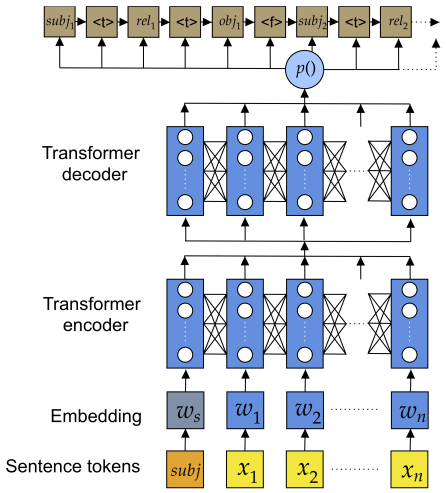
In this section, we describe the methods we used to train and evaluate our relation extraction models. All of our proposed classifiers and end-to-end models have 6 Transformer layers and 1 embedding layer, with number of neurons (hidden layer size) set to 512. In the Transformer-based models, we use 8 attention heads. Our models are trained using the AdaFactor [30] optimizer. We use the publicly available Tensor2Tensor [35][10] framework for our experiments and will be releasing our code extensions as part of that framework. On our proposed dataset, the classifiers are trained for 50,000 iterations with batch-size of 1024 and the end-to-end models are trained for 50,000 iterations with batch-size of 256.

We evaluate classifiers and end-to-end models on our dataset. These results are presented in Table 3. The end-to-end model is learning to recognize entities, resolving entity co-references, and reason

[10] https://github.com/tensorflow/tensor2tensor



(a) Classifier



(b) Transformer encoder-decoder

Figure 1: Fact extraction model architectures

about their relation in one pass through the model. To the best of our knowledge, we are not aware of other end-to-end structured relation extraction models and therefore do not include a comparison against other approaches. Some examples of extracting facts on our dataset are shown in A.2, where we include a comparison to OpenIE's triplet extraction.

We calculate precision and recall in the above experiments by matching ground-truth fact tuples exactly. This implies that the end-to-end model is not only learning to identify entities and resolve co-references, but also predict structured output, and its outputs can be used for reasoning. Their performance is competitive against

Model	P	R	F1
Binary Classifier*	59.60	75.13	66.47
Relation Classifier	63.49	68.64	65.96
E2E	71.67	56.21	63.01
E2E-Reduced**	**72.16**	61.03	66.13

Table 3: Performance (precision(P), recall(R), F1) of models on our proposed dataset grouped by classifiers and then end-to-end models. Binary Classifier is described in Sec 4.3, Classifier in Sec 4.1 and E2E is the end-to-end model described in Sec 4.2. The Binary Classifier* only considers the existence of a relation, and might not be directly comparable to the other models' performance. E2E-Reduced is a model where sentences where no entities are detected are filtered out from the input text. The best model is marked in bold. We consider precision(P) as the measure that matches best with the definition of $fact_{acc}$**

relation classifiers while having a simple training and inference routine.

For each model, we sort and select the ten most frequent relation types that appear in our test sets. The $F1$ measure on these relations for classifiers are shown in Table 4, and end-to-end models are shown in Table 5.

Relation	P	R	F1
No relation	0.9830	0.9817	0.9824
Country of citizenship	0.6446	0.9394	0.7646
Date of birth	0.9330	0.9850	0.9582
Country	0.6049	0.9484	0.7386
Located in territory	0.6260	0.8118	0.7069
Instance of	0.5097	0.7015	0.5904
Place of birth	0.6430	0.7436	0.6897
Member of sports team	0.5179	0.9248	0.6640
Occupation	0.5934	0.7770	0.6729
Date of death	0.9163	0.9875	0.9506

Table 4: Precision (P), Recall (R) and F1 measure of the relation classifier (Section 4.1) on our test sets on ten most frequent relations.

Relation	P	R	F1
Country of citizenship	0.8247	0.8359	0.8302
Instance of	0.7212	0.6676	0.6934
Date of birth	0.9342	0.9798	0.9564
Country	0.8387	0.8267	0.8327
Cast member	0.5889	0.4910	0.5355
Place of birth	0.7012	0.7348	0.7176
Located in the administrative territorial entity	0.7293	0.7700	0.7491
Member of sports team	0.7045	0.7027	0.7036
Occupation	0.5911	0.5774	0.5842
Educated at	0.5432	0.7278	0.6221

Table 5: Precision (P), Recall (R) and F1 measure of our end-to-end model (Section 4.2) on our test sets on ten most frequent relations.

7 ERROR ANALYSIS OF MODEL PREDICTIONS

Distant supervision [17] is a way to create training data by using weak signals. In our dataset, we assign a relation label r_k for every entity pair (e_i, e_j) in the input text X if the relation tuple (e_i, r_k, e_j) exists in the Wikidata knowledge base W_{KB}. However, the sentence S_l containing (e_i, e_j) may not necessarily entail r_k. This leads to inaccurate estimates of the true-positive rate for our fact extraction models. We evaluate the effect of this distant supervision by gathering the set of facts extracted from our models that are marked false-positive by the distant supervision scheme. We present a pair of input text (Wikipedia articles) and facts extracted by our models to human evaluators, and ask them to mark a fact to be *True* only if the relation tuple $(subject, relation, object)$ is implied by the input text. We asked two evaluators to score facts marked false-positive from a random set of 30 Wikipedia articles. We consider the fact to be true if both evaluators agree. We present the results in Table 6, where we can see the rate of false-positive facts that were marked true by the evaluators. This suggests that the end-to-end models could benefit by a better labeling scheme.

Model	% True-positives
End-to-end	**77.8**
Relation Classifier	46.6

Table 6: Percentage of true facts that were inaccurately labelled wrong by the distant supervisor. The End-to-end model is the best model from Section 4.2 and Classifier is the best from 4.1(Transformer-Sigmoid). The End-to-end model (in bold) predicts facts that are likelier to be true.

8 EVALUATION OF $fact_{acc}$ AS A METRIC

In this section, we show the effectiveness of our proposed metric on judging the factual accuracy of generated text. We use the text summarization model proposed in [15] to generate lead sections of Wikipedia articles using the dataset and model in that paper, and compare the generated summary against the real lead section. In the following section, we describe the methodology used to compare human judgment of factual accuracy and how we compare our metric against that baseline.

8.1 Human Evaluation

Every claim made in the generated text G can be considered to belong to one of three categories: supported by a sentence in ground-truth T, refuted by T or cannot be verified by T. The evaluators were asked to only consider claims that are either supported or refuted by T. This ensures that no external knowledge is used in comparing T and G, and ignores all claims that cannot be verified by T. Four evaluators were asked to rate 30 examples of generated text G and then give it a score of 1-5 with 5 being highest factual accuracy. A special case is where the generated text has no verifiable claims. In this case, they were asked to give it a score of 1. Figure 2 shows the interface a human evaluator uses in our experiment.

We conduct the same experiment on two sets of data: first is a random sampling from summaries generated for Actors. We consider this an easier subset because we expect our fact extraction models to do well on this subset due to the summaries and Wikipedia lead sections generally containing relationships our models perform well on (see tables 4 and 5). We present these results in Table 7. We analyzed the inter-rater agreement on the scores given to each example, and found that Krippendorff's alpha (allows for ordinal rankings) was 0.6897. The second is a random sampling from all categories in Wikipedia. The results are presented in Table 8. The inter-rater agreement on this sample was found to be 0.7530.

We see that our end-to-end model (Section 4.2) has the best correlation on both subsets, indicating that it generalizes better to generated text. This may also be because the classifier suffers from a compounding of errors, where it is unable to predict relations if the NER system fails to recognize entities.

Metric	Correlation with human scores
ROUGE-1	0.583
ROUGE-2	0.639
ROUGE-L	0.634
OpenIE	0.258
$fact_{acc}$-Binary Classifier	0.596
$fact_{acc}$-Relation Classifier	0.523
$fact_{acc}$-E2E	0.645
$fact_{acc}$-E2E-Reduced	**0.668**

Table 7: Spearman correlation of different metrics with human evaluation of factual accuracy on the 'Actors' subset of summaries. ROUGE and OpenIE are described in Sec 5, and the model-based $fact_{acc}$ metrics are described in Sec 4. The best metric is shown in bold.

Metric	Correlation with human scores
ROUGE-1	0.384
ROUGE-2	0.435
ROUGE-L	0.339
OpenIE	0.128
$fact_{acc}$-Binary Classifier	0.200
$fact_{acc}$-Relation Classifier	0.250
$fact_{acc}$-E2E	0.314
$fact_{acc}$-E2E-Reduced	**0.453**

Table 8: Spearman correlation of different metrics with human evaluation of factual accuracy on a random subset of summaries. ROUGE and OpenIE are described in Sec 5, and the model-based $fact_{acc}$ metrics are described in Sec 4. The best metric is shown in bold.

9 CONCLUSION

9.1 Limitations

The dataset we create only makes use of sentences found in Wikipedia, and facts found in WikiData. This means that our models are biased to sentences structured to the neutral tone set in Wikipedia, and towards popular types of facts expressed in WikiData such as date of birth, profession, etc. Other sources of text may have more complex structures and styles of writing that may make it hard for our

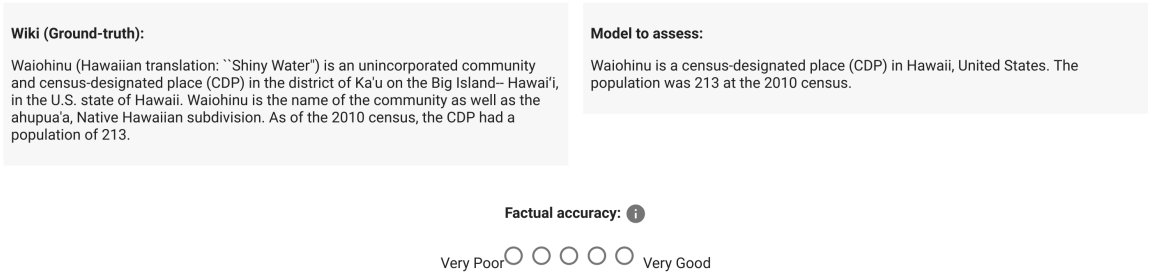


Figure 2: A screenshot of the interface presented to human evaluators to judge the factual accuracies of generated text. The ground-truth text is shown on the left, with the model generated text on the right. The evaluator is then asked to rate the factual accuracy of the generated text on a five point scale of 'Very Poor' to 'Very Good'

models to adapt to easily. An simple example of this is negating a binary relationship with 'not', and different ways of expressing the same idea such as 'wife/husband' instead of 'spouse'. WikiData is an incomplete knowledge base, and this also leads to many sentences that in reality imply a fact to be marked containing no facts. This is a very typical problem faced by any work using distant supervision, and is combated with methods like active learning [31].

It should be noted that ROUGE and to the best of our knowledge, most other automatic metrics, are also susceptible to changes in linguistic style and structure. However, elaborate labeling and bigger datasets will allow for our models to learn to overcome these challenges.

9.2 Discussion and future work

We have shown that our proposed metric is able to indicate the factual accuracy of generated text, and agrees with human judgment on our datasets. By leveraging a new dataset for both relation classification and end-to-end fact extraction, we also showed that classifiers and end-to-end models with straightforward architectures are able to perform competitive fact extraction.

Our end-to-end model avoids compounding of errors over subcomponents typically used in other fact-extraction pipelines. We will release the code and datasets used to train this model, so that the proposed metric can be used to standardize comparison. We are in the process of building a bigger dataset that will contain multiple text domains, stronger human supervision and a larger collection of relation tuples that will help overcome many of the limitations discussed in the previous section (9.1). We encourage further development and use of this metric for automating the assessment of factual accuracy of generated text, and the development of better end-to-end models with structured outputs for fact extraction.

REFERENCES

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

[2] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2670–2676.

[3] Peter F. Brown, Vincent J. Della Pietra, Robert L. Mercer, Stephen A. Della Pietra, and Jennifer C. Lai. 1992. An Estimate of an Upper Bound for the Entropy of English. *Computational Linguistics* 18, 1 (March 1992), 31–40.

[4] Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2017. Faithful to the Original: Fact Aware Neural Abstractive Summarization. *CoRR* abs/1711.04434 (2017). arXiv:1711.04434 http://arxiv.org/abs/1711.04434

[5] Jason Chiu and Eric Nichols. 2016. Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics* 4 (2016), 357–370.

[6] Kevin Clark and Christopher D. Manning. 2016. Improving Coreference Resolution by Learning Entity-Level Distributed Representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 643–653. https://doi.org/10.18653/v1/P16-1061

[7] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*. 363–370.

[8] Eduard Hovy, Chin-Yew Lin, Liang Zhou, and Junichi Fukumoto. 2006. Automated Summarization Evaluation with Basic Elements. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA).

[9] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 260–270. https://doi.org/10.18653/v1/N16-1030

[10] Alon Lavie and Abhaya Agarwal. 2007. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Stroudsburg, PA, USA, 228–231.

[11] Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. StanfordâĂŹs Multi-Pass Sieve Coreference Resolution System at the CoNLL-2011 Shared Task. In *In Proceedings of the CoNLL-2011 Shared Task*.

[12] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial Learning for Neural Dialogue Generation. In *Conference on Empirical Methods in Natural Language Processing*. 2157–2169.

[13] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, Stan Szpakowicz Marie-Francine Moens (Ed.). Association for Computational Linguistics, Barcelona, Spain, 74–81.

[14] Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural Relation Extraction with Selective Attention over Instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. 2124–2133.

[15] Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Łukasz Kaiser, and Noam Shazeer. 2018. Generating Wikipedia by Summarizing Long Sequences. In *Proceedings of the 2018 International Conference on Learning Representations*.

[16] Andrew Mccallum and David Jensen. 2003. A Note on the Unification of Information Extraction and Data Mining using Conditional-Probability, Relational Models. In *In Proceedings of the IJCAI-2003 Workshop on Learning Statistical Models from Relational Data*.

[17] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant Supervision for Relation Extraction Without Labeled Data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. 1003–1011.

[18] Makoto Miwa and Mohit Bansal. 2016. End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1105–1116. https://doi.org/10.18653/v1/P16-1105

[19] Makoto Miwa and Yutaka Sasaki. 2014. Modeling Joint Entity and Relation Extraction with Table Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. 1858–1869.

[20] Thahir P. Mohamed, Estevam R. Hruschka, Jr., and Tom M. Mitchell. 2011. Discovering Relations Between Noun Categories. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1447–1455.

[21] Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, ÃĞaglar GülÃğehre, and Bing Xiang. 2016. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of the 2016 SIGNLL Conference on Computational Natural Language Learning*.

[22] Ani Nenkova and Rebecca J. Passonneau. 2004. Evaluating Content Selection in Summarization: The Pyramid Method.. In *Proceedings of the 2005 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 145–152.

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, 311–318. https://doi.org/10.3115/1073083.1073135

[24] Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A Multi-pass Sieve for Coreference Resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, 492–501.

[25] Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. 2013. The Life and Death of Discourse Entities: Identifying Singleton Mentions. In *Proceedings of the 2013 North American Chapter of the Association for Computational Linguistics*.

[26] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation Extraction with Matrix Factorization and Universal Schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 74–84.

[27] Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A Neural Attention Model for Abstractive Sentence Summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

[28] Iulian Vlad Serban, Tim Klinger, Gerald Tesauro, Kartik Talamadupula, Bowen Zhou, Yoshua Bengio, and Aaron C. Courville. 2017. Multiresolution Recurrent Neural Networks: An Application to Dialogue Response Generation. In *Proceedings of the 2017 AAAI Conference on Artificial Intelligence*. 3288–3294.

[29] Iulian Vlad Serban, Chinnadhurai Sankar, Mathieu Germain, Saizheng Zhang, Zhouhan Lin, Sandeep Subramanian, Taesup Kim, Michael Pieper, Sarath Chandar, Nan Rosemary Ke, Sai Mudumba, Alexandre de Brébisson, Jose Sotelo, Dendi Suhubdy, Vincent Michalski, Alexandre Nguyen, Joelle Pineau, and Yoshua Bengio. 2017. A Deep Reinforcement Learning Chatbot. *CoRR* abs/1709.02349 (2017). arXiv:1709.02349 http://arxiv.org/abs/1709.02349

[30] Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive Learning Rates with Sublinear Memory Cost. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*. 4603–4611.

[31] Yanyao Shen, Hyokun Yun, Zachary C. Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep Active Learning for Named Entity Recognition. *CoRR* abs/1707.05928 (2017). arXiv:1707.05928 http://arxiv.org/abs/1707.05928

[32] Daniil Sorokin and Iryna Gurevych. 2017. Context-Aware Representations for Knowledge Base Relation Extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 1784–1789.

[33] Josef Steinberger and Karel Jezek. 2009. Evaluation Measures for Text Summarization. *Computing and Informatics* 28 (2009), 251–275.

[34] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance Multi-label Learning for Relation Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, Stroudsburg, PA, USA, 455–465.

[35] Ashish Vaswani, Samy Bengio, Eugene Brevdo, François Chollet, Aidan N. Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, Ryan Sepassi, Noam Shazeer, and Jakob Uszkoreit. 2018. Tensor2Tensor for Neural Machine Translation. *arXiv preprint* arXiv:1803.07416 (2018). http://arxiv.org/abs/1803.07416

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). 5998–6008.

[37] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM* 57 (2014), 78–85. Issue 10.

[38] Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in Data-to-Document Generation. *CoRR* abs/1707.08052 (2017). arXiv:1707.08052 http://arxiv.org/abs/1707.08052

[39] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *CoRR* abs/1609.08144 (2016). arXiv:1609.08144 http://arxiv.org/abs/1609.08144

[40] Hengtong Zhang, Yaliang Li, Fenglong Ma, Jing Gao, and Lu Su. 2018. TextTruth: An Unsupervised Approach to Discover Trustworthy Information from Multi-Sourced Text Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. ACM, New York, NY, USA, 2729–2737. https://doi.org/10.1145/3219819.3219977

A APPENDIX

A.1 Reproducibility

We release code to train our fact extraction models as part of the Tensor2Tensor framework[11] along with trained model weights at https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact. A large fact extraction dataset (Sec 3) based on Wikidata and Wikipedia is made available [12]. To train our end-to-end and classifier models for fact extraction, we use the hyper-parameter set "transformer_base" defined in the Tensor2Tensor framework[13]. We further release code to use our end-to-end models as a fact extractor and calculate the factual accuracy metric at https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact.

A.2 Fact extraction example

We include an example of facts extracted from text using our models where we compare it against OpenIE's [2] triplet extraction in Table 9. This example illustrates the advantage of using structured approaches to fact extraction. OpenIE yields many triplets that mostly cannot be used for reasoning.

[11] https://github.com/tensorflow/tensor2tensor
[12] https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikifact
[13] https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/models/transformer.py

Input	Christopher Simon (born 5 June 1963) is an Australian actor and producer. Born in Sydney, Australia. He produced the film Miss You Already directed by Catherine Hardwicke. Simon is also a producer of such films as The Sweeney (2012 film) directed by Nick Love, Pusher, I, Anna, Still Life, Me and Me Dad, Boogie Woogie, The Proposition, Beyond the Ocean, The Trouble with Men and Women. He also produced short films by Joe Wright such as The End and Nick Love's Love Story. Simon's various television acting roles include Eddie in The Long Firm, Pedro in Gimme Gimme Gimme, Michael Hassan in The Bill, Lee Andersen in Casualty, Abdel in Lovejoy Samir in Ultimate Force, Da Souza in Lynda La Plante's Supply and Demand, Nathan Morgan in Wire In The Blood and he appeared in Lenny Henry in Pieces. Film acting roles include Room To Rent, The Delivery and O Jerusalem. Simon has acted in such plays as 12 Angry Men and Taking Sides both directed by Harold Pinter in London's west end, The Kitchen directed by Stephen Daldry at the Royal Court, the Amnesty award winning one man show When The Bulbull Stopped Singing for which he was nominated for the Acting Excellence Award (Best Actor) at the Edinburgh Festival Fringe, which premiered at the Traverse theatre and toured to Iran, New York and Jordan. Other theatre roles include Welcome to Ramallah, which toured York and London, at the Arcola and the Theatre Royal York, The Present at the Royal Court and the Bush, and Poor Superman at the Hampstead and the Traverse.
Targets	(Christopher Simon, date of birth, June 5 1963), (Christopher Simon, country of citizenship, Australian), (Christopher Simon, place of birth, Sydney)
OpenIE	(Abdel, is in, Ultimate Force), (Casualty, Abdel in, Ultimate Force), (Nathan Morgan, is In, Blood), (Lee Andersen, is in, Casualty), (Da Souza, is in, Lynda La Plante 's Supply), (Simon 's various television acting roles, include, Eddie), (Simon, is producer of, films as Sweeney directed by Nick Love), (Simon, is also producer of, such films as Sweeney), (Simon 's television roles, include, Eddie in Firm), (Simon, is producer of, such films), (Simon 's various television roles, include, Eddie), (Eddie, is in, Long Firm), (Simon, is producer of, such films as Sweeney), (Simon, is producer of, such films as Sweeney directed by Nick Love), (Michael Hassan, is in, Bill), (Bill, Andersen in, Casualty), (You, Already directed by, Catherine Hardwicke), (Simon, is producer of, films), (Simon, has, various television acting roles), (Simon 's television acting roles, include, Eddie), (Abdel, is in, Lovejoy Samir), (Simon 's television roles, include, Eddie in Long Firm), (Simon 's television acting roles, include, Eddie in Firm), (Simon, is producer of, films as Sweeney directed), (Simon 's various television acting roles, include, Eddie in Firm), (Simon, is also producer of, films as Sweeney), (Simon 's various television acting roles, include, Eddie in Long Firm), (Simon, is, producer), (Rent, To Room is, Delivery), (Simon 's television roles, include, Eddie), (Simon, is also producer of, films as Sweeney directed), (Lynda La Plante, in, Supply), (Pedro, is in, Gim), ...
Seq2Seq	(Christopher Simon, date of birth, June 5 1963), (Christopher Simon, country of citizenship, Australian), (Christopher Simon, place of birth, Sydney), (Christopher Simon, occupation, Actor)
Classifier	(Christopher Simon, date of birth, June 5 1963), (Christopher Simon, country of citizenship, Australian)

Table 9: Comparison of fact tuples extracted from this example, using OpenIE, our end-to-end model (Section 4.2), and our classifier (Section 4.1). A triplet consists of $(subject, relation, object)$**.**