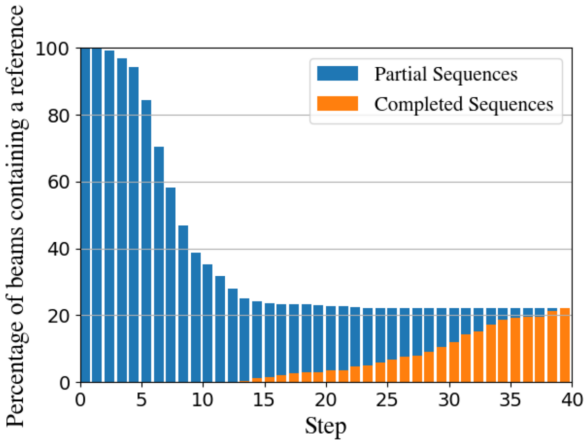

Figure 1: The percentage of beams that contain a reference sentence after each step of beam search. A beam size of 10 was used to decode the model proposed in Dusek and Jurcıcek (2016). Results are for the E2E validation dataset. The orange bars indicate the number of completed references within the beam.

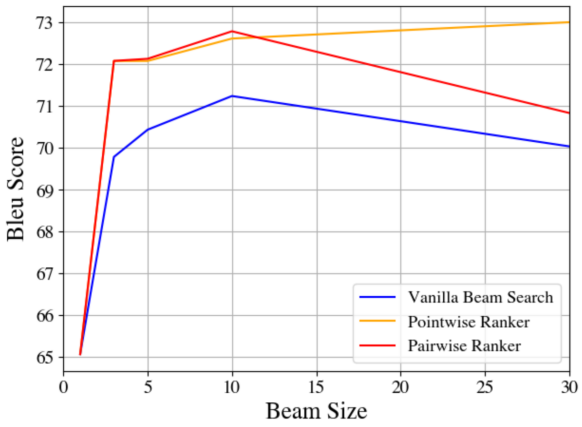

Figure 2: Comparison between the performance of the pointwise and pairwise rankers when used as rerankers on the E2E validation set.

1 Appendices

1.1 Fallout experiment with larger beam size

Section 1 contains a graph which indicates the step at which the reference sentences drop out of the beam (for a beam size of 3). Figure 1 indicates the same results for a larger beam size of 10.

The figure indicates that the number of references that were contained in the final beam was higher for a beam size 10. For the early iterations of the decoding the number of references that fell out of the beam was far lower for a beam size of 10. A larger beam size meant that the beam contained more hypotheses and so has more chances to match against a reference.

However the shape of the graphs is very similar. The majority of references that fell out did so relatively early in the process. 54% of references fell out by step 7, increasing to 79% by step 9. At step 21 the final last reference fell out of the beam despite the fact that the beam contained partially references up to step 40.

1.2 Pointwise vs Pairwise rerankers

This paper required a method of ranking completed hypotheses from worst to best. During preliminary experiments we implemented rerankers based on the Pairwise and Pointwise strategies from the Information Retrieval field. See Section 3.2 for more details.

To evaluate the performance of the different rankers we applied each of the rankers as a reranker of the final beam of a vanilla beam search over the E2E validation set. The BLEU scores for each of the rerankers were calculated for each beam size. The results are shown in Figure 2.

We can see that there was very little difference in performance for the two methods of reranking for the beam sizes up to 10. However, for beam size 30 the pointwise reranker significantly outperforms the pairwise reranker. The larger the beam size the greater the number of hypotheses that the reranker can pick as top and hence the greater the impact of the reranker.

The pointwise reranker requires O(k) runs of the reranker to produce a total ordering. On the other hand the Copeland method to produce a total ordering from the pairwise comparisons requires $O(k^2)$ number of pairwise Comparisons.

These factors lead us to choose the Pointwise ranker over the pairwise ranker for the experiments in the results section.

1.3 Numerical results

This section will present the numerical results for the E2E and WebNLG datasets so that they can be more readily compared in future works.The results are given in Table 1 and Table 2.

1.4 Hyper-parameters

Throughout this paper a number of hyperparamaters were introduced. The values used for each of the models in this paper are sumarised bellow.

Beam size	Vanilla	Rerank	TGEN	LN	LN+Rerank	BM	LN+BM
1	64.72	64.72	64.72	64.72	64.72	64.72	64.72
3	64.47	64.94	65.33	65.93	65.26	65.73	**66.06**
5	64.69	65.36	65.47*	**66.40**	66.19*	**66.40**	66.27
10	64.78*	65.67*	65.58	66.47*	66.19	**66.71***	66.61
30	63.65	65.25	65.44	65.58	66.05	**66.40**	66.25*

Table 1: BLEU scores for each of the different systems on the E2E testset. *indicates the beam size which scored highest on the respective validation sets.**bold** indicates the highest scoring system for each beam size.

Beam size	Vanilla	Rerank	TGEN	LN	LN+Rerank	BM	LN+BM
1	42.14	42.14	42.14	42.14	42.14	42.14	42.14
3	42.10*	47.93*	47.28*	47.02	47.37	**48.38**	47.78
5	41.77	48.13	47.41	47.49	47.54*	47.92*	**48.39**
10	41.33	47.33	46.50	47.11*	47.70	47.66	**48.21**
30	41.20	47.42	46.61	47.18	47.81	47.41	**48.44***

Table 2: BLEU scores for each of the different systems on the WebNLG testset. *indicates the beam size which scored highest on the respective validation sets. **bold** indicates the highest scoring system for each beam size.

Note that the search for these values was far from exhaustive so there is a good chance that the results of this paper could be improved upon through a better optimisation procedure.

In Section 3.3, the beam is split into two sections bottom and rest. For all beam manipulation models the bottom of the beam was set to the bottom (ie lowest scoring) quarter of the beam. We also say a large beam is used to generate the data for training the beam. For all experiments in this paper we use a beam size of 50.

A key hyperparameter for performance of the incremental beam manipulation was the steps of the beam search at which the beam was manipulated. This hyperparameter varied for the 4 seperate beam manipulation models. The values are summarised as follows:

- E2E-Incremental Beam Manipulation on top of vanilla beam search: 5,10,15,20 and final.
- E2E-Incremental Beam Manipulation on top of length normalised beam search: 5,7 and 10.
- WebNLG-Incremental Beam Manipulation on top of vanilla beam search: 4,12 and final.
- WebNLG-Incremental Beam Manipulation on top of length normalised beam search: 5 and 12.

It is worth noting that manipulating the final step is the same as reranking the beam according to the ranker used in beam manipulation (i.e. no rollouts are performed).