# Adaptive Beam Search Decoding for Discrete Keyphrase Generation

## Supplementary Materials

Paper ID: 8780

## A. Comparison to fully-parallelized decoding method

In our main paper, we propose two decoding methods: one is a fully parallelized decoding method, and the other is a semi-parallelized decoding method. Considering that the first word is not completely different, we choose the latter for adaptive beam search decoding. To further illustrate the effectiveness of our decoding method, we compare the two decoding methods, and $F_1@5$ and $F_1@M$ results on five datasets: Inspec (Hulth 2003), Krapivin (Krapivin and Marchese 2009), NUS (Nguyen and Kan 2007), SemEval (Kim et al. 2010) and KP20k (Krapivin and Marchese 2009), are reported in tables 1 to 4.

| Model | Inspec | Krapivin | NUS | SemEval | KP20k |
|---|---|---|---|---|---|
| ExHiRD | 0.235 | 0.286 | — | 0.284 | 0.311 |
| AdaGM(fully) | <u>0.301</u> | <u>0.347</u> | <u>0.427</u> | <u>0.337</u> | <u>0.373</u> |
| AdaGM(semi) | **0.305** | **0.363** | **0.442** | **0.343** | **0.388** |

Table 1: Results of present keyphrases $F@5$ on five datasets. The best results are shown in bold, and the second best results are underlined.

| Model | Inspec | Krapivin | NUS | SemEval | KP20k |
|---|---|---|---|---|---|
| ExHiRD | 0.291 | **0.347** | — | 0.335 | **0.374** |
| AdaGM(fully) | <u>0.332</u> | <u>0.339</u> | <u>0.433</u> | **0.346** | 0.337 |
| AdaGM(semi) | **0.348** | 0.323 | **0.438** | <u>0.337</u> | <u>0.345</u> |

Table 2: Results of present keyphrases $F@M$ on five datasets. The best results are shown in bold, and the second best results are underlined.

It can be seen from the 4 tables that the decoding method we adopted is better than the fully parallelized decoding method. As for the present keyphrases, our decoding method improves the effect of $F1@5$ from 0.347 to 0.363, and on the absent keyphrases, the $F_1@5$ score is doubled (from 0.022 to 0.043). Furthermore, the fully parallelized decoding method (the first word is completely different) is better than the ExHiRD (Chen et al. 2020) model, which demonstrates the effectiveness of our reset state mechanism.



| Model | Inspec | Krapivin | NUS | SemEval | KP20k |
|---|---|---|---|---|---|
| ExHiRD | <u>0.011</u> | 0.022 | — | 0.017 | 0.016 |
| AdaGM(fully) | 0.010 | <u>0.026</u> | <u>0.021</u> | <u>0.026</u> | <u>0.022</u> |
| AdaGM(semi) | **0.016** | **0.050** | **0.037** | **0.032** | **0.043** |

Table 3: Results of absent keyphrases $F@5$ on five datasets. The best results are shown in bold, and the second best results are underlined.

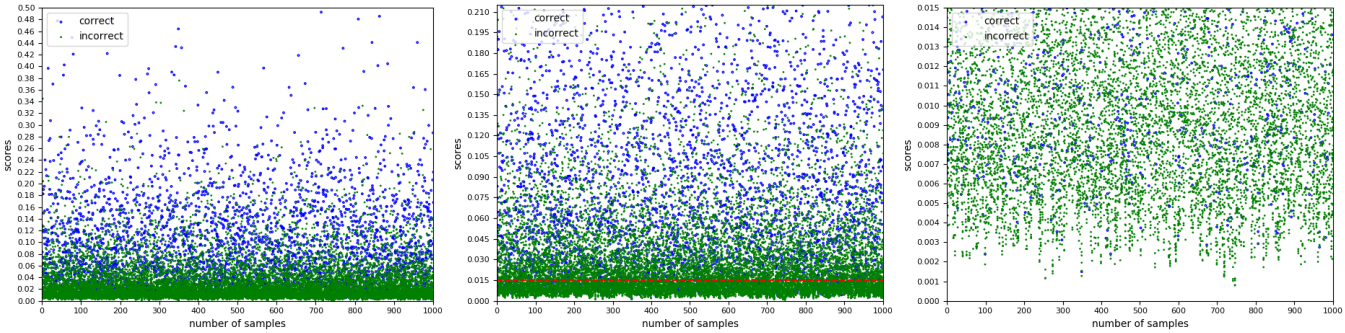
| Model | Inspec | Krapivin | NUS | SemEval | KP20k |
|---|---|---|---|---|---|
| ExHiRD | <u>0.022</u> | 0.043 | — | 0.025 | 0.032 |
| AdaGM(fully) | 0.018 | <u>0.049</u> | <u>0.033</u> | <u>0.036</u> | <u>0.042</u> |
| AdaGM(semi) | **0.024** | **0.076** | **0.059** | **0.039** | **0.071** |

Table 4: Results of absent keyphrases $F@M$ on five datasets. The best results are shown in bold, and the second best results are underlined.

## B. Visualization of first words' scores with reset state mechanism

Here we give an impressive visualization of the first words' scores after applying our novel reset state mechanism. To be more specific, we randomly sample 1,000 documents on the KP20k validation set (Krapivin and Marchese 2009) and count the beam scores of the correct and incorrect first words after inference. The results are reported in Figure 1.

In an overview, it can be seen that: (1)The score of first words is mainly concentrated within $[0.0, 0.2]$, while samples in other intervals (i.e., $[0.2, 1.0]$) are almost correct ones as shown in Figure 1(a). (2)Zooming into score interval $[0.0, 0.2]$ (i.e., Figure 1(b)), we can find that there is a clear diving line at score=0.02, above which are mainly correct samples and incorrect ones are below it. However, it should be pointed out that 0.015 is still the best choice even score=0.02 may be better intuitively. Further exploration is




(a) Score distribution of all first words. (b) Score distribution between 0 and 0.2. (c) Score distribution between 0 and 0.015.

Figure 1: Score distribution of first words on 1000 documents.

| Threshold | Inspec | | Krapivin | | NUS | | SemEval | | KP20k | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M |
| $\alpha$=0.015(AdaGM) | **0.305** | **0.348** | **0.363** | 0.323 | **0.442** | **0.438** | **0.343** | 0.337 | **0.388** | 0.345 |
| $\alpha$=0.016 | 0.303 | 0.345 | 0.356 | 0.329 | 0.436 | 0.428 | **0.343** | **0.338** | **0.388** | 0.349 |
| $\alpha$=0.017 | 0.303 | 0.343 | 0.358 | 0.333 | 0.439 | 0.432 | 0.341 | 0.335 | 0.387 | 0.353 |
| $\alpha$=0.018 | 0.303 | 0.341 | 0.357 | 0.334 | 0.436 | 0.430 | 0.339 | 0.335 | 0.387 | 0.356 |
| $\alpha$=0.019 | 0.301 | 0.338 | 0.357 | **0.337** | 0.435 | 0.431 | 0.339 | 0.336 | 0.386 | **0.359** |

Table 5: Results of generated present keyphrases according to different threshold $\alpha$ on five datasets. The best results are bold.

| Threshold | Inspec | | Krapivin | | NUS | | SemEval | | KP20k | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M | $F_1$@5 | $F_1$@M |
| $\alpha$=0.015(AdaGM) | **0.016** | **0.024** | **0.050** | **0.076** | **0.037** | **0.059** | **0.032** | 0.039 | **0.043** | **0.071** |
| $\alpha$=0.016 | 0.010 | 0.017 | 0.039 | 0.062 | 0.031 | 0.050 | 0.030 | 0.039 | 0.029 | 0.054 |
| $\alpha$=0.017 | 0.010 | 0.017 | 0.037 | 0.061 | 0.029 | 0.045 | 0.030 | 0.040 | 0.028 | 0.053 |
| $\alpha$=0.018 | 0.010 | 0.018 | 0.037 | 0.060 | 0.029 | 0.045 | 0.030 | 0.040 | 0.028 | 0.052 |
| $\alpha$=0.019 | 0.009 | 0.017 | 0.036 | 0.060 | 0.027 | 0.044 | 0.030 | **0.041** | 0.027 | 0.051 |

Table 6: Results of generated absent keyphrases according to different threshold $\alpha$ on five datasets. The best results are bold.

**Document:** particle based non photorealistic volume visualization. non photorealistic techniques are usually applied to produce stylistic renderings . in visualization , these techniques are often able to simplify data , producing clearer images than traditional visualization methods . we investigate the use of particle systems for visualizing volume datasets using non photorealistic techniques . in our volumeflies framework , user selectable rules affect particles to produce a variety of illustrative styles in a unified way . the techniques presented do not require the generation of explicit intermediary surfaces .
**Keyphrases**: **visualization**; **particle** systems; **non** photorealistic rendering; **volume** rendering.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | **non**: | **particle**: | **volume**: | **visualization**: | image: | scientific: | 3d: |
| | 0.291 | 0.134 | 0.107 | 0.050 | 0.013 | 0.012 | 0.011 |
| Ours: | graphics: | surface: | data: | computer: | level: | user: | selectable: |
| {first word:score} | 0.009 | 0.008 | 0.008 | 0.007 | 0.006 | 0.006 | 0.005 |
| | multi: | volumeflies: | illustrative: | parallel: | human: | | |
| | 0.005 | 0.005 | 0.004 | 0.0048 | 0.004 | | |

Figure 2: Example of generated first word score.

**Document:** a new fuzzy rule based classification system for word sense disambiguation. word sense disambiguation (wsd) can be thought of as the most challenging task in the process of machine translation. various supervised and unsupervised learning methods have already been proposed for this purpose. in this paper , we propose a new efficient fuzzy classification system in order to be applied for wsd. in order to optimize the generalization accuracy, we use rule weight as a simple mechanism to tune the classifier and propose a new learning method to iteratively adjust the weight of fuzzy rules. through computer simulations on twa data as a standard corpus, the proposed scheme shows a uniformly good behavior and achieves results which are comparable or better than other classification systems, proposed in the past.
**Keyphrases**: **classification**; **word** sense disambiguation; **machine** translation; **generalization** accuracy; **rule** weight; **fuzzy** systems.

| | | | | | | |
|---|---|---|---|---|---|---|
| | **word**: | **fuzzy**: | **machine**: | **rule**: | **classification**: | **generalization**: |
| | 0.279 | 0.252 | 0.067 | 0.066 | 0.065 | 0.024 |
| Ours: | learning: | unsupervised: | supervised: | twa: | natural: | artificial: |
| {first word:score} | 0.012 | 0.011 | 0.008 | 0.008 | 0.007 | 0.006 |
| | data: | information: | weight: | support: | efficient: | pattern: |
| | 0.004 | 0.004 | 0.004 | 0.004 | 0.003 | 0.003 |

Figure 3: Example of generated first word score.

given below. (3)Taking a closer look at the score distribution within $[0, 0.015]$, which can be referred to Figure 1(c), the number of correct samples below threshold 0.015 is negligible to incorrect ones. It indicates the scores of incorrect samples are seriously suppressed to much lower score (e.g., close to 0.0) than the correct ones after applying our reset state mechanism.

We give a further study about the different performance when threshold in the range of 0.015-0.02 with the interval 0.001 on all test datasets as reported in Table 5 and 6. The results illustrate the rationality of our choice of threshold 0.015.

We also give two examples in Figure 2 and Figure 3. In the last column of Figure 2 and Figure 3, bolding means that the generated first word is correct, and we mark its score in red. It can be seen that our model is indeed able to distinguish the score of the correct first word from the incorrect first word.

## References


Chen, W.; Chan, H. P.; Li, P.; and King, I. 2020. Exclusive Hierarchical Decoding for Deep Keyphrase Generation. In ACL, 1095–1105.

Hulth, A. 2003. Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In EMNLP, 216–223.

Kim, S. N.; Medelyan, O.; Kan, M.-Y.; and Baldwin, T. 2010. SemEval-2010 Task 5 : Automatic Keyphrase Extraction from Scientific Articles. In Proceedings of the 5th International Workshop on Semantic Evaluation, 21–26.

Krapivin, M.; and Marchese, M. 2009. Large Dataset for Keyphrase Extraction. Technical Report DISI-09-055, DISI, Trento, Italy .

Nguyen, T. D.; and Kan, M.-Y. 2007. Keyphrase Extraction in Scientific Publications. In ICADL.