Optimizing Rank-based Metrics with Blackbox Differentiation

Supplementary Material

A. Parameters of retrieval experiments

In all experiments we used the ADAM optimizer with a weight decay value of 4×10^{-4} and batch size 128. All experiments ran at most 80 epochs with a learning rate drop by 70% after 35 epochs and a batch memory of length 3. We used higher learning rates for the embedding layer as specified by defaults in Cakir et al. [5].

We used a super-label batch preparation strategy in which we sample a consecutive batches for the same super-label pair, as specified by Cakir et al. [5]. For the In-shop Clothes dataset we used 4 batches per pair of super-labels and 8 samples per class within a batch. In the Online Products dataset we used 10 batches per pair of super-labels along with 4 samples per class within a batch. For CUB200, there are no super-labels and we just sample 4 examples per classes within a batch. These values again follow Cakir et al. [5]. The remaining settings are in Table 1.

	Online Products	In-shop	CUB200
lr	3×10^{-6}	10^{-5}	5×10^{-6}
margin	0.02	0.05	0.02
λ	4	0.2	0.2

Table 1: Hyperparameter values for retrieval experiments.

B. Proofs

Lemma 1. *Let $\{w_k\}$ be a sequence of nonnegative weights and let $r_1, \dots, r_n$ be positive integers. Then*

$$\sum_{k=1}^{\infty} w_k |\{i : r_i \geq k\}| = \sum_{i=1}^{n} W(r_i), \tag{1}$$

where

$$W(k) = \sum_{i=1}^{k} w_i \quad \text{for } k \in \mathbb{N}. \tag{2}$$

Note that the sum on the left hand-side of (1) is finite.

Proposition 2. *Let w_K be nonnegative weights for $K \in \mathbb{N}$ and assume that L_{rec} is given by*

$$L_{rec}(\mathbf{y}, \mathbf{y}^*) = \sum_{K=1}^{\infty} w_K \, L@K(\mathbf{y}, \mathbf{y}^*). \tag{3}$$

Then

$$L_{rec}(\mathbf{y}, \mathbf{y}^*) = \frac{1}{|\operatorname{rel}(\mathbf{y}^*)|} \sum_{i \in \operatorname{rel}(\mathbf{y}^*)} W(r_i), \tag{4}$$

where W is as in (2).

Proof. Taking the complement of the set $\operatorname{rel}(\mathbf{y}^*)$ in the definition of $L@K$, we get

$$L@K(\mathbf{y}, \mathbf{y}^*) = \frac{|\{i \in \operatorname{rel}(\mathbf{y}^*) : r_i \geq K\}|}{|\operatorname{rel}(\mathbf{y}^*)|}, \tag{5}$$

whence (3) reads as

$$L_{\text{rec}}(\mathbf{y}, \mathbf{y}^*) = \frac{1}{|\operatorname{rel}(\mathbf{y}^*)|} \sum_{k=1}^{\infty} w_K |\{i : r_i \geq K\}|.$$

Equation (4) then follows by Lemma 1. □

proof of Lemma 1. Observe that $w_k = W(k) - W(k-1)$ and $W(0) = 0$. Then

$$\begin{aligned}
\sum_{i=1}^{n} W(r_i) &= \sum_{k=1}^{\infty} W(k) |\{i : r_i = k\}| \\
&= \sum_{k=1}^{\infty} W(k) \big|\{i : r_i \geq k\} \setminus \{i : r_i \geq k+1\}\big| \\
&= \sum_{k=1}^{\infty} W(k) |\{i : r_i \geq k\}| \\
&\quad - \sum_{k=1}^{\infty} W(k-1) |\{i : r_i \geq k\}| \\
&= \sum_{k=1}^{\infty} \big(W(k) - W(k-1)\big) |\{i : r_i \geq k\}| \\
&= \sum_{k=1}^{\infty} w_k |\{i : r_i \geq k\}|
\end{aligned}$$

and (1) follows. □

Proof of (20). Let us set $w_k = \log(1 + 1/k)$ for $k \in \mathbb{N}$. Then from Taylor's expansion of log we have the desired $w_k \approx \frac{1}{k}$ and

$$\begin{aligned}
W(k) &= \sum_{i=1}^{k} \log\left(1 + \frac{1}{i}\right) \\
&= \log\left(\prod_{i=1}^{k} \frac{1+i}{i}\right) = \log(1+k).
\end{aligned}$$

If we set

$$w_k = \log\left(1 + \frac{\log\left(1+\frac{1}{k}\right)}{1+\log k}\right), \quad \text{for } k \in \mathbb{N}$$

then, using Taylor's expansions again,

$$w_k \approx \frac{\log\left(1+\frac{1}{k}\right)}{1+\log k} \approx \frac{1}{k\log k}$$

and

$$\begin{aligned} W(k) &= \sum_{i=1}^{k} \log\left(1 + \frac{\log\left(1+\frac{1}{k}\right)}{1+\log k}\right) \\ &= \log\left(\prod_{i=1}^{k} \frac{1+\log(1+i)}{1+\log i}\right) \\ &= \log\bigl(1+\log(1+k)\bigr). \end{aligned}$$

The conclusion then follows by Proposition 2. □

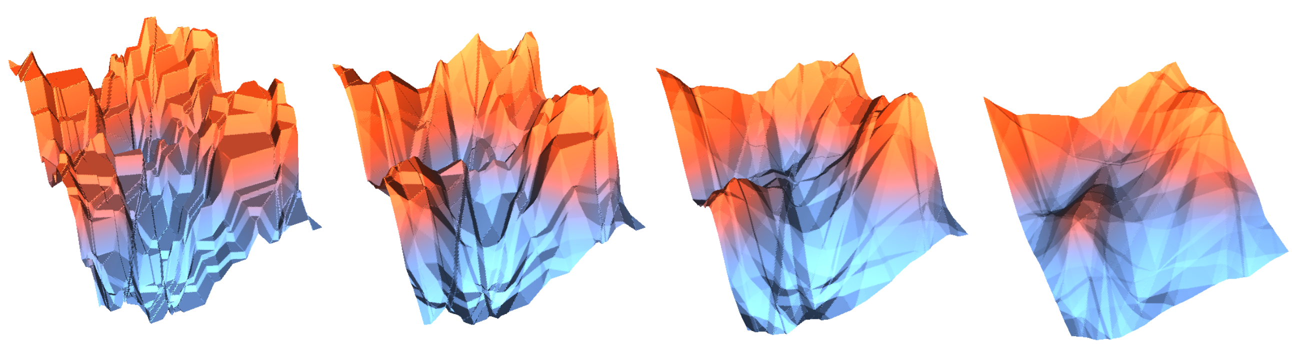
C. Ranking surrogates visualization

For the interested reader, we additionally present visualizations of smoothing effects introduced by different approaches for direct optimization of rank-based metrics. We display the behaviour of our approach using blackbox differentiation [60], of FastAP [4], and of SoDeep [10].

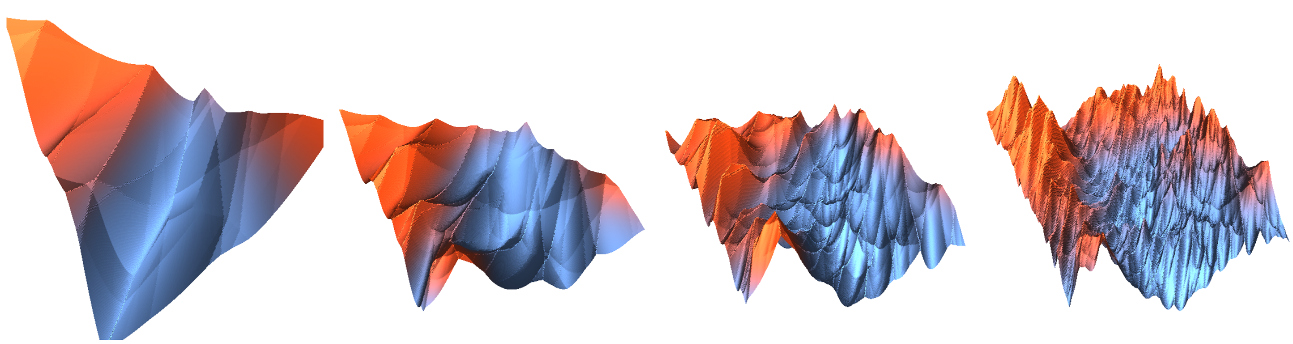
In the following, we fix a 20-dimensional score vector $w \in \mathbb{R}^{20}$ and a loss function L which is a (random but fixed) linear combination of the ranks of w. We plot a (random but fixed) two-dimensional section of $\mathbb{R}^{20}$ of the loss landscape $L(w)$. In Fig. 2a we see the true piecewise constant function. In Fig. 2b, Fig. 2c and Fig. 2d the ranking is replaced by interpolated ranking [60], FastAP soft-binning ranking [4] and by pretrained SoDeep LSTM [10], respectively. In Fig. 1a and Fig. 1b the evolution of the loss landscape with respect to parameters is displayed for the blackbox ranking and FastAP.

Acknowledgement

We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Marin Vlastelica and Claudio Michaelis. We acknowledge the support from the German Federal Ministry of Education and Research (BMBF) through the Tübingen AI Center (FKZ: 01IS18039B). Claudio Michaelis was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via grant EC 479/1-1 and the Collaborative Research Center (Projektnummer 276693517 – SFB 1233: Robust Vision).

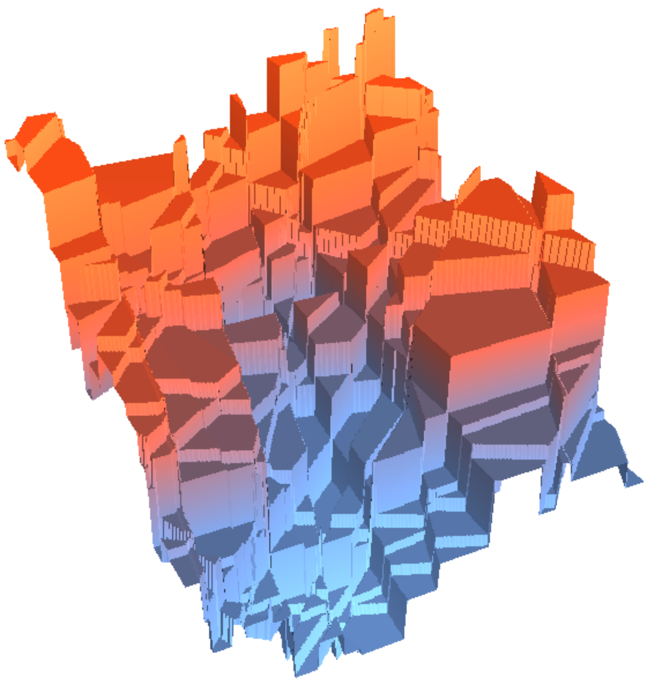


(a) Ranking interpolation by [60] for $\lambda = 0.2, 0.5, 1.0, 2.0$.

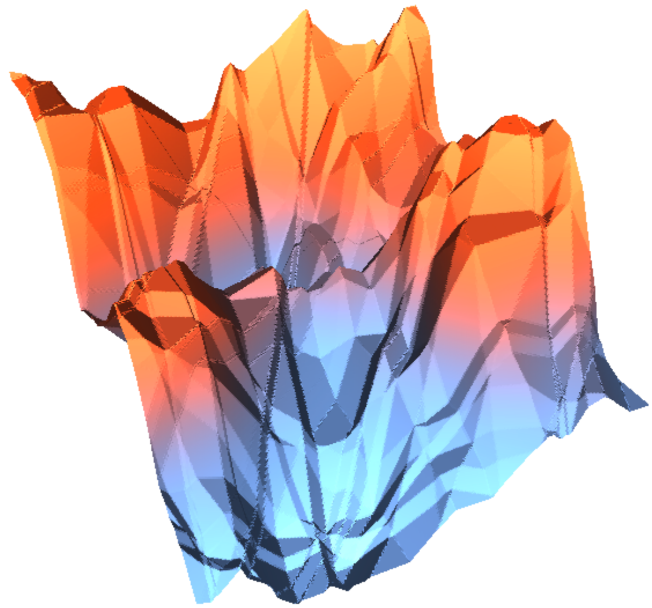


(b) FastAp [4] with bin counts $5, 10, 20, 40$.

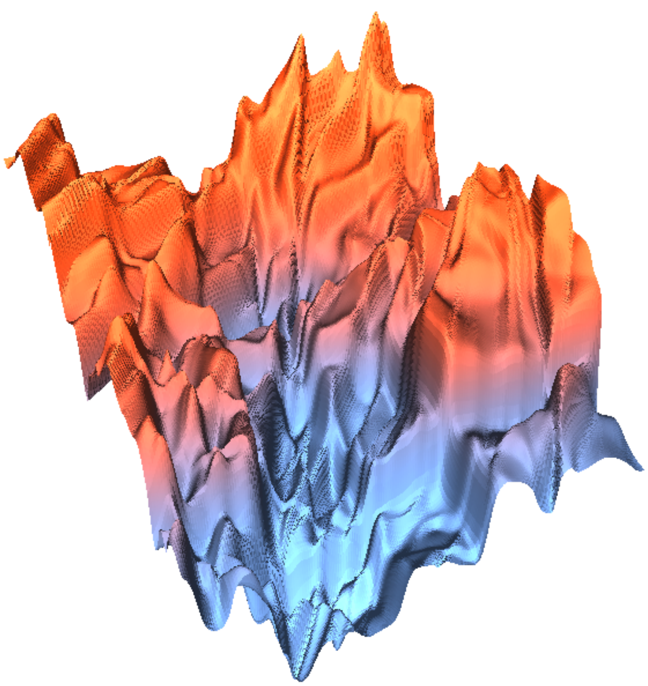
Figure 1: Evolution of the ranking-surrogate landscapes with respect to their parameters.



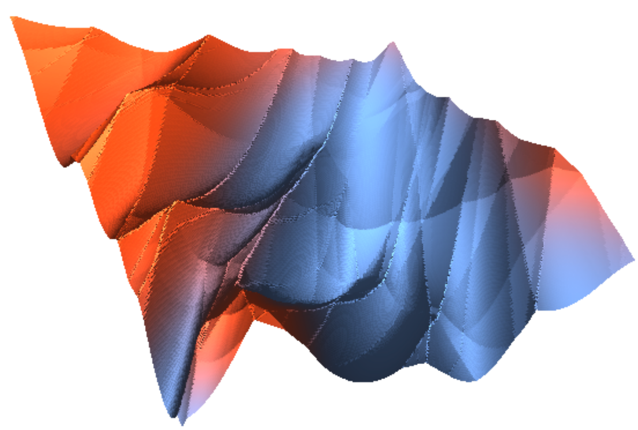
(a) Original piecewise constant landscape



(b) Piecewise linear interpolation scheme of [60] with $\lambda = 0.5$



(c) SoDeep LSTM-based ranking surrogate [10]



(d) FastAP [4] soft-binning with 10 bins.

Figure 2: Visual comparison of various differentiable proxies for piecewise constant function.