Multileave Gradient Descent for Fast Online Learning to Rank

Fred Stoney, Jagroop Sarkaria, Ben Barrett, Shuche Wang

Problem

The learning rate for current ranking systems is too slow

- Leads to needing large amounts of data to improve ranking
- Slow to adjust to a single user’s preferences
- Offline ranking requires lots of labels and data entries which is costly
- Slow interleaving methods (comparison model for different rankers) that worked linearly and had complexity N(N-1)

Evaluation Methods/Previous work

Offline Learning:

- Offline Learning models- labelled datasets are expensive, large, and inaccurate.
- User Studies - not repeatable and do not scale.

Online Learning:

- A/B testing - are less sensitive so they take longer to train
- Balanced Interleave - can result in preference for a ranker regardless of user's choices and are highly sensetive
- Probabilistic Interleave - runs risk of showing poor rankings to users

Duel Bandit Gradient Descent model is what used these evaluation methods

Background

Team Draft Interleave - Rankers draft documents -> interleave documents -> user input determines better ranker. N * (N-1) complexity.

Team Draft Multileave - Same method but able to compare against many rankers at once - better performance

Dueling bandit gradient descent - current ranking system is compared to a slight variant, and use gradient descent to adjust parameters in favor of the winner.

Multileave Bandit gradient descent - create n candidate rankers, and use multileave comparison, then adjust parameters in favor of winner.

Multileave Gradient Descent Algorithm

Extends Dueling Bandit Gradient Descent to use multiple candidate rankers

Multiple rankers = More likely to find better ranker

Should take less time to converge than DBGD

Disadvantage: quality of list presented to user may decrease

Algorithm

1. Receive query from user
2. Generate the current best ranking
3. Generate n candidate rankings
4. Multileave results together
5. Get feedback from user (clicks)
6. If best ranking didn't "win", update

Algorithm 2 Multileave Gradient Descent (MGD).

Input: n, α, δ, $\mathbf{w}_0^0$, $update(\mathbf{w}, \alpha, \{\mathbf{b}\}, \{\mathbf{u}\})$
for $t \leftarrow 1..\infty$ **do**
    $q_t \leftarrow receive\_query(t)$ *// obtain a query from a user*
    $\mathbf{l}_0 \leftarrow generate\_list(\mathbf{w}_t^0, q_t)$ *// ranking of current best*
    for $i \leftarrow 1...n$ **do**
        $\mathbf{u}_t^i \leftarrow sample\_unit\_vector()$
        $\mathbf{w}_t^i \leftarrow \mathbf{w}_t^0 + \delta \mathbf{u}_t^i$ *// create a candidate ranker*
        $\mathbf{l}_t^i \leftarrow generate\_list(\mathbf{w}_t^i, q_t)$ *// exploratory ranking*
    $\mathbf{m}_t, \mathbf{t}_t \leftarrow TDM\_multileave(\mathbf{l}_t)$ *// multileaving and teams*
    $\mathbf{c}_t \leftarrow receive\_clicks(\mathbf{m}_t)$ *// show multileaving to the user*
    $\mathbf{b}_t \leftarrow TDM\_infer(\mathbf{t}_t, \mathbf{c}_t)$ *// set of winning candidates*
    if $\mathbf{w}_t^0 \in \mathbf{b}_t$ **then**
        $\mathbf{w}_{t+1}^0 \leftarrow \mathbf{w}_t^0$ *// if current best among winners, no update*
    else
        $\mathbf{w}_{t+1}^0 \leftarrow update(\mathbf{w}_t^0, \alpha, \mathbf{b}_t, \mathbf{u}_t)$ *// Algorithm 3 or 4*

Multileave Gradient Descent

Set of winning rankers based on clicks

If current ranker is in winners, don't update

MGD-W: Move towards random winner

MGD-M: Move toward mean of winners

Algorithm 3 MGD update function: winner takes all (MGD-W).

Input: $\mathbf{w}, \alpha, \mathbf{b}, \mathbf{u}$ *// in* $\mathbf{b}$ *are only winners*
$\mathbf{b}^j \leftarrow pick\_random\_uniformly(\mathbf{b})$
return $\mathbf{w} + \alpha \mathbf{u}^j$

Algorithm 4 MGD update function: mean winner (MGD-M).

Input: $\mathbf{w}, \alpha, \mathbf{b}, \mathbf{u}$ *// in* $\mathbf{b}$ *are only winners*
return $\mathbf{w} + \alpha \frac{1}{|\mathbf{u}|} \sum_{\mathbf{b}^j \in \mathbf{b}} \mathbf{u}^j$

Experiment

- **Static Dataset**

  9 datasets:

  HP2003 NP2003 TD2003 HP2004 NP2004 TD2004 MQ2007 MQ2008 OHSUMED

  1. Queries
  2. Manual relevance assessments
  3. Documents represented as feature vectors

Experiment

- **Simulating Clicks Models**

  Cascade click model (CCM)

Table 1: Instantiations of CCM [6] as used in our experiments.

		$P(click = 1 \mid R)$			$P(stop = 1 \mid R)$		
Relevance grade R		0	1	2	0	1	2
perfect	(per)	0.0	0.5	1.0	0.0	0.0	0.0
navigational	(nav)	0.05	0.5	0.95	0.2	0.5	0.9
informational	(inf)	0.4	0.7	0.9	0.1	0.3	0.5
almost random	(a.ra)	0.4	0.5	0.6	0.5	0.5	0.5

P(click=1|R): users decide whether it warrants a click

P(stop=1|R): whether users' information be satisfied

1. Perfect: Clicks on all highly relevant and only on relevant documents.

2. Navigational: Clicks on a single highly relevant document.

3. Informational: Clicks on several documents, less dependent on their relevance.

Experiment Runs

- Number of candidates *n* in {1,2,6,9,20}.
- Baseline: DBGD algorithm
- Initialization Parameters:

Candidates generation $\alpha = 0.01$

Updating rate for DBGD: $\delta = 1$

Updating rate for MGD: $\alpha = 0.03$

- Evaluation:

$$NDCG = \sum_{i=1}^{\kappa} \frac{2^{rel(\mathbf{r}[i])-1}}{log_2(i+1)} iNDCG^{-1}.$$

RESULTS AND ANALYSIS

Learning Speed:

- Offline performance of **Both *MGD-M-n*** and ***MGD-W-n*** improves monotonically with the increase of the number of candidates *n.*
- Systems with more candidates learn much faster.
- In ***perfect*** feedback, there is less of an effect as there is less to gain over an already well performing baseline.
- When the noise in user feedback increases, the advantage of ***MGD*** over ***DBGD*** becomes stronger.

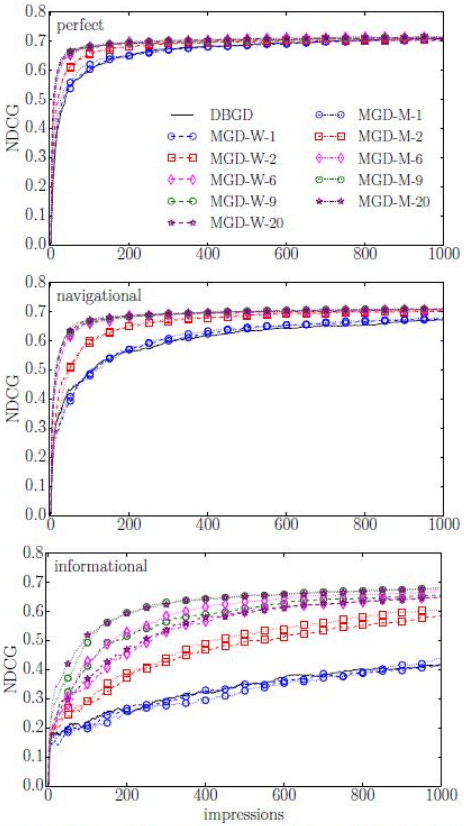

Figure 1: Offline performance (NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for the *perfect*, *navigational* and *informational* click model.

Learning Speed: Offline Score

Table 2: Offline score (NDCG) after 1,000 query impressions of each of the algorithms for the 3 instantiations of the CCM (see Table 1). Bold values indicate maximum performance per dataset and click model. Statistically significant improvements (losses) over the DBGD baseline are indicated by △ ($p < 0.05$) and ▲ ($p < 0.01$) (▽ and ▼). We show the standard deviation between brackets.

		HP2003	NP2003	TD2003	HP2004	NP2004	TD2004	MQ2007	MQ2008	OHSUMED
perfect	DBGD	0.766 (0.06)	0.710 (0.05)	0.299 (0.09)	0.730 (0.07)	0.715 (0.08)	0.303 (0.03)	0.381 (0.03)	0.476 (0.04)	0.443 (0.05)
	MGD-W-2	0.771 (0.06)	0.705 (0.05)	0.314 (0.08)	0.731 (0.06)	0.726 (0.07)	0.306 (0.03)	0.392 (0.02) ▲	0.480 (0.04)	0.445 (0.05)
	MGD-W-4	0.771 (0.06)	0.712 (0.05)	0.318 (0.08)	0.742 (0.06)	**0.732** (0.07)	0.310 (0.04)	0.396 (0.02) ▲	0.481 (0.04)	**0.447** (0.05)
	MGD-W-6	0.778 (0.06)	0.712 (0.05)	0.314 (0.08)	0.745 (0.06)	0.725 (0.07)	0.308 (0.04)	0.398 (0.02) ▲	0.479 (0.04)	0.444 (0.05)
	MGD-W-9	0.774 (0.06)	0.713 (0.05)	0.314 (0.07)	0.744 (0.06)	0.725 (0.07)	0.311 (0.04)	0.400 (0.02) ▲	0.481 (0.04)	0.430 (0.04) ▽
	MGD-W-20	0.776 (0.06)	0.710 (0.05)	0.314 (0.07)	**0.749** (0.06) △	0.726 (0.07)	0.308 (0.04)	0.396 (0.02) ▲	0.480 (0.04)	0.438 (0.05)
	MGD-M-2	0.771 (0.07)	0.712 (0.05)	0.312 (0.08)	0.743 (0.06)	0.730 (0.08)	0.311 (0.04)	0.392 (0.02) ▲	0.480 (0.04)	0.443 (0.05)
	MGD-M-4	0.777 (0.07)	0.711 (0.05)	0.317 (0.07)	0.742 (0.07)	0.729 (0.07)	0.315 (0.04) ▲	0.400 (0.02) ▲	0.482 (0.04)	**0.447** (0.05)
	MGD-M-6	0.779 (0.06)	**0.716** (0.04)	0.320 (0.07)	0.747 (0.06) △	0.725 (0.07)	0.312 (0.04) △	0.402 (0.02) ▲	0.481 (0.04)	**0.447** (0.05)
	MGD-M-9	**0.780** (0.06)	0.714 (0.05)	**0.322** (0.07) △	0.747 (0.06) △	0.726 (0.07)	0.311 (0.04)	0.406 (0.02) ▲	**0.484** (0.04)	0.437 (0.04)
	MGD-M-20	0.777 (0.06)	0.714 (0.05)	0.321 (0.07) △	0.747 (0.06) △	0.724 (0.08)	**0.316** (0.04) ▲	**0.408** (0.02) ▲	**0.484** (0.04)	0.446 (0.04)
navigational	DBGD	0.725 (0.07)	0.672 (0.06)	0.281 (0.09)	0.676 (0.08)	0.693 (0.08)	0.281 (0.03)	0.370 (0.03)	0.460 (0.04)	0.433 (0.06)
	MGD-W-2	0.766 (0.06) ▲	0.702 (0.05) ▲	0.306 (0.09) △	0.732 (0.06) ▲	0.715 (0.08) △	0.303 (0.03) ▲	0.372 (0.03)	0.466 (0.04)	**0.438** (0.05)
	MGD-W-4	0.769 (0.06) ▲	0.708 (0.05) ▲	0.314 (0.08) ▲	0.735 (0.06) ▲	0.720 (0.08) ▲	**0.307** (0.04) ▲	0.380 (0.02) ▲	0.469 (0.04)	0.437 (0.04)
	MGD-W-6	**0.772** (0.06) ▲	0.705 (0.05) ▲	0.312 (0.08) ▲	0.738 (0.06) ▲	0.721 (0.08) ▲	0.304 (0.04) ▲	0.382 (0.02) ▲	0.468 (0.04)	0.431 (0.05)
	MGD-W-9	0.771 (0.06) ▲	0.708 (0.05) ▲	0.304 (0.07) △	0.738 (0.06) ▲	**0.725** (0.08) ▲	0.304 (0.04) ▲	0.388 (0.02) ▲	0.470 (0.04) △	0.431 (0.05)
	MGD-W-20	0.771 (0.06) ▲	0.710 (0.05) ▲	0.314 (0.07) ▲	0.738 (0.06) ▲	0.721 (0.07) ▲	0.304 (0.04) ▲	0.386 (0.03) ▲	0.470 (0.04)	0.432 (0.05)
	MGD-M-2	0.766 (0.06) ▲	0.703 (0.06) ▲	0.302 (0.08)	0.726 (0.06) ▲	0.717 (0.08) △	0.301 (0.04) ▲	0.376 (0.03)	0.467 (0.04)	0.435 (0.05)
	MGD-M-4	0.768 (0.06) ▲	0.705 (0.05) ▲	0.312 (0.08) ▲	0.738 (0.06) ▲	0.721 (0.07) ▲	0.305 (0.04) ▲	0.385 (0.02) ▲	0.468 (0.04)	0.435 (0.04)
	MGD-M-6	**0.772** (0.06) ▲	0.707 (0.05) ▲	0.309 (0.07) ▲	0.736 (0.07) ▲	0.723 (0.08) ▲	0.305 (0.04) ▲	0.387 (0.03) ▲	**0.473** (0.04) ▲	0.437 (0.05)
	MGD-M-9	0.771 (0.06) ▲	0.710 (0.05) ▲	0.317 (0.08) ▲	**0.741** (0.06) ▲	0.724 (0.07) ▲	0.302 (0.04) ▲	**0.391** (0.02) ▲	0.472 (0.04) △	0.436 (0.04)
	MGD-M-20	0.769 (0.06) ▲	**0.711** (0.04) ▲	**0.318** (0.08) ▲	**0.741** (0.06) ▲	0.720 (0.07) ▲	0.306 (0.04) ▲	0.390 (0.02) ▲	0.472 (0.04) △	0.437 (0.05)
informational	DBGD	0.460 (0.22)	0.418 (0.19)	0.167 (0.10)	0.401 (0.21)	0.489 (0.19)	0.197 (0.08)	0.323 (0.05)	0.419 (0.05)	0.407 (0.05)
	MGD-W-2	0.677 (0.11) ▲	0.585 (0.13) ▲	0.223 (0.10) ▲	0.625 (0.11) ▲	0.629 (0.11) ▲	0.233 (0.07) ▲	0.338 (0.04) △	0.427 (0.05)	0.418 (0.05)
	MGD-W-4	0.722 (0.07) ▲	0.636 (0.09) ▲	0.253 (0.09) ▲	0.667 (0.10) ▲	0.662 (0.09) ▲	0.258 (0.05) ▲	0.343 (0.04) ▲	0.440 (0.04) ▲	0.422 (0.05) △
	MGD-W-6	0.727 (0.06) ▲	0.652 (0.07) ▲	0.249 (0.09) ▲	0.674 (0.09) ▲	0.669 (0.10) ▲	0.272 (0.04) ▲	0.344 (0.04) ▲	0.441 (0.04) ▲	0.423 (0.05) △
	MGD-W-9	0.727 (0.06) ▲	0.656 (0.07) ▲	0.264 (0.10) ▲	0.679 (0.07) ▲	0.670 (0.09) ▲	0.259 (0.05) ▲	0.339 (0.04) ▲	0.434 (0.04) △	0.418 (0.06)
	MGD-W-20	0.729 (0.06) ▲	0.649 (0.06) ▲	0.252 (0.09) ▲	0.675 (0.09) ▲	0.664 (0.10) ▲	0.260 (0.05) ▲	0.341 (0.04) ▲	0.434 (0.05) △	0.415 (0.05)
	MGD-M-2	0.696 (0.09) ▲	0.606 (0.10) ▲	0.241 (0.09) ▲	0.634 (0.12) ▲	0.645 (0.12) ▲	0.246 (0.07) ▲	0.333 (0.04)	0.430 (0.05)	0.421 (0.05) △
	MGD-M-4	0.736 (0.06) ▲	0.659 (0.07) ▲	0.276 (0.09) ▲	0.685 (0.08) ▲	0.682 (0.09) ▲	0.271 (0.04) ▲	0.350 (0.03) ▲	0.443 (0.04) ▲	0.427 (0.05) ▲
	MGD-M-6	0.742 (0.06) ▲	0.667 (0.06) ▲	0.275 (0.09) ▲	0.692 (0.07) ▲	0.687 (0.09) ▲	0.278 (0.04) ▲	0.351 (0.04) ▲	0.448 (0.04) ▲	0.427 (0.05) ▲
	MGD-M-9	0.745 (0.06) ▲	**0.681** (0.06) ▲	0.283 (0.08) ▲	**0.710** (0.08) ▲	0.698 (0.08) ▲	0.284 (0.04) ▲	**0.361** (0.03) ▲	**0.454** (0.04) ▲	0.425 (0.05) ▲
	MGD-M-20	**0.752** (0.06) ▲	0.677 (0.07) ▲	**0.295** (0.08) ▲	0.703 (0.07) ▲	**0.703** (0.08) ▲	**0.291** (0.04) ▲	0.356 (0.03) ▲	0.452 (0.04) ▲	**0.430** (0.05) ▲

- ***Perfect*** feedback converged performance does not change much
- ***MGD*** improves more over the baseline with the more noisy feedback and the more candidates.

Learning Speed: Online Score

Table 3: Online score (discounted cumulative NDCG, see Section 5.4) of each of the algorithms for the 3 instantiations of the CCM (see Table 1). Bold values indicate maximum performance per dataset and click model. Statistically significant improvements (losses) over the DBGD baseline are indicated by $^{\triangle}$ ($p < 0.05$) and $^{\blacktriangle}$ ($p < 0.01$) ($^{\triangledown}$ and $^{\blacktriangledown}$). We show the standard deviation between brackets.

		HP2003	NP2003	TD2003	HP2004	NP2004	TD2004	MQ2007	MQ2008	OHSUMED
perfect	**DBGD**	95.88 (23.04)	97.79 (7.19)	36.28 (16.18)	97.93 (19.02)	102.92 (8.81)	**42.92** (14.73)	60.11 (4.49)	78.17 (4.54)	70.43 (3.85)
	MGD-W-2	110.77 (5.91) ▲	101.71 (5.87) ▲	41.19 (4.84) ▲	100.92 (6.67)	106.69 (6.17) ▲	38.15 (3.32) ▼	60.49 (3.23)	78.79 (3.97)	72.58 (3.58) ▲
	MGD-W-4	112.96 (5.14) ▲	103.42 (5.94) ▲	42.36 (3.91) ▲	104.44 (5.27) ▲	108.94 (5.48) ▲	38.50 (2.66) ▼	61.33 (3.47) △	78.52 (4.96)	72.73 (3.14) ▲
	MGD-W-6	113.37 (5.13) ▲	104.13 (5.42) ▲	43.00 (3.94) ▲	104.79 (5.46) ▲	110.02 (5.39) ▲	38.13 (2.43) ▼	61.22 (3.30) △	78.76 (4.01)	72.73 (3.48) ▲
	MGD-W-9	114.66 (4.57) ▲	**105.79** (5.88) ▲	43.53 (3.63) ▲	**107.22** (4.80) ▲	110.27 (5.69) ▲	38.39 (2.35) ▼	60.62 (3.42)	78.12 (3.93)	70.32 (3.11)
	MGD-W-20	**116.25** (4.21) ▲	104.96 (5.23) ▲	44.43 (4.19) ▲	106.23 (5.22) ▲	109.94 (5.91) ▲	39.79 (2.42) ▽	60.28 (3.35)	78.07 (3.87)	72.42 (3.58) ▲
	MGD-M-2	111.23 (5.59) ▲	101.42 (6.54) ▲	41.26 (4.37) ▲	101.67 (7.35) △	108.24 (6.08) ▲	37.51 (2.86) ▼	61.43 (3.54) △	79.08 (4.29)	72.74 (4.00) ▲
	MGD-M-4	113.91 (4.86) ▲	103.48 (5.61) ▲	42.64 (4.39) ▲	103.58 (5.62) ▲	109.70 (5.79) ▲	38.39 (2.43) ▼	**61.85** (3.10) ▲	**79.11** (4.41)	**72.84** (3.63) ▲
	MGD-M-6	113.46 (4.66) ▲	104.25 (5.02) ▲	43.22 (3.88) ▲	105.31 (5.16) ▲	109.80 (5.66) ▲	38.68 (2.47) ▼	61.00 (3.06)	78.97 (3.91)	72.78 (3.16) ▲
	MGD-M-9	115.81 (4.21) ▲	105.09 (4.71) ▲	44.02 (4.06) ▲	106.79 (5.36) ▲	**110.88** (5.94) ▲	38.38 (2.11) ▼	60.64 (2.75)	77.88 (3.86)	70.97 (3.12)
	MGD-M-20	115.67 (4.99) ▲	104.75 (4.79) ▲	**44.50** (3.71) ▲	107.05 (4.71) ▲	110.51 (5.69) ▲	40.18 (2.31) ▽	61.57 (3.10) ▲	78.69 (3.73)	72.51 (3.24) ▲
navigational	**DBGD**	78.79 (24.72)	85.83 (13.70)	32.21 (15.15)	80.61 (20.58)	90.92 (13.27)	37.46 (13.46)	58.07 (4.96)	76.04 (5.21)	66.99 (5.71)
	MGD-W-2	105.53 (8.45) ▲	95.55 (8.07) ▲	38.54 (5.71) ▲	92.59 (10.96) ▲	100.96 (9.07) ▲	35.72 (4.74)	59.24 (4.34) △	77.18 (4.95)	71.34 (4.56) ▲
	MGD-W-4	110.07 (6.55) ▲	100.22 (6.74) ▲	41.58 (4.81) ▲	100.37 (7.05) ▲	105.81 (6.72) ▲	37.90 (2.68)	59.57 (4.11) △	78.18 (4.52) ▲	72.00 (3.47) ▲
	MGD-W-6	109.97 (5.47) ▲	101.36 (5.14) ▲	41.66 (4.39) ▲	101.00 (6.46) ▲	107.07 (5.95) ▲	38.21 (3.00)	60.18 (3.28) ▲	77.88 (4.35) ▲	72.62 (3.67) ▲
	MGD-W-9	**113.17** (5.33) ▲	101.71 (5.19) ▲	43.03 (4.53) ▲	102.54 (5.67) ▲	106.63 (6.02) ▲	39.04 (2.86)	**60.71** (3.54) ▲	77.91 (4.49) ▲	72.69 (3.84) ▲
	MGD-W-20	112.18 (4.70) ▲	101.85 (5.68) ▲	42.24 (4.60) ▲	102.82 (5.80) ▲	107.02 (6.20) ▲	39.28 (2.91)	60.55 (3.14) ▲	77.77 (4.34) ▲	72.39 (3.37) ▲
	MGD-M-2	106.27 (8.78) ▲	94.34 (8.44) ▲	39.21 (5.48) ▲	94.70 (9.55) ▲	103.34 (8.49) ▲	36.40 (4.00)	59.65 (4.32) ▲	77.72 (4.73) ▲	71.04 (4.52) ▲
	MGD-M-4	109.44 (6.21) ▲	99.22 (6.87) ▲	41.02 (4.33) ▲	99.01 (7.37) ▲	105.82 (6.92) ▲	38.09 (3.21)	60.25 (3.59) ▲	78.16 (4.53) ▲	72.49 (3.81) ▲
	MGD-M-6	110.70 (5.77) ▲	100.56 (5.54) ▲	42.45 (4.57) ▲	102.04 (6.37) ▲	106.04 (6.51) ▲	38.28 (3.48)	60.31 (3.30) ▲	77.84 (4.89) ▲	**72.79** (3.57) ▲
	MGD-M-9	112.30 (5.35) ▲	**102.95** (5.62) ▲	42.74 (4.55) ▲	**103.96** (6.55) ▲	**107.41** (5.94) ▲	**39.55** (3.08)	60.36 (3.29) ▲	77.89 (3.94) ▲	72.71 (3.97) ▲
	MGD-M-20	111.38 (5.37) ▲	102.23 (5.45) ▲	**43.10** (4.34) ▲	102.88 (5.79) ▲	106.78 (5.16) ▲	38.79 (2.76)	60.21 (3.56) ▲	**78.23** (4.13) ▲	72.53 (3.11) ▲
informational	**DBGD**	48.23 (27.66)	50.42 (19.83)	22.46 (11.82)	43.36 (21.88)	59.58 (22.88)	27.76 (11.89)	55.60 (6.00)	71.94 (5.56)	62.99 (7.22)
	MGD-W-2	72.76 (18.10) ▲	64.09 (18.01) ▲	25.94 (7.55) ▲	62.61 (18.11) ▲	69.48 (16.80) ▲	26.52 (6.26)	55.83 (4.98)	73.33 (5.09) △	66.39 (5.93) ▲
	MGD-W-4	78.49 (16.93) ▲	73.02 (14.67) ▲	29.34 (6.15) ▲	70.73 (14.81) ▲	78.97 (13.16) ▲	28.21 (5.60)	56.04 (4.11)	74.31 (4.93) ▲	67.70 (4.44) ▲
	MGD-W-6	81.96 (14.73) ▲	73.66 (12.21) ▲	30.84 (6.17) ▲	70.90 (12.56) ▲	81.44 (11.32) ▲	29.12 (4.57)	56.24 (4.18)	**74.86** (4.79) ▲	67.75 (4.55) ▲
	MGD-W-9	85.14 (10.63) ▲	77.57 (10.24) ▲	30.26 (5.61) ▲	73.60 (11.66) ▲	82.86 (11.52) ▲	28.45 (4.36)	56.09 (3.61)	73.64 (4.71) △	68.48 (4.52) ▲
	MGD-W-20	84.44 (11.41) ▲	74.65 (9.53) ▲	29.91 (4.96) ▲	69.06 (13.01) ▲	81.78 (10.73) ▲	28.56 (4.28)	56.18 (3.46)	73.09 (4.87)	67.51 (4.63) ▲
	MGD-M-2	70.70 (18.56) ▲	66.03 (18.38) ▲	27.33 (6.76) ▲	61.78 (20.42) ▲	71.30 (18.17) ▲	27.29 (6.20)	56.28 (4.55)	73.35 (5.68) △	67.15 (5.63) ▲
	MGD-M-4	84.38 (14.30) ▲	76.41 (12.63) ▲	28.98 (5.88) ▲	72.82 (14.05) ▲	82.12 (11.01) ▲	28.44 (4.77)	56.58 (4.35)	74.45 (4.87) ▲	**68.75** (4.49) ▲
	MGD-M-6	85.51 (11.36) ▲	76.37 (11.58) ▲	31.55 (6.33) ▲	73.52 (13.17) ▲	83.05 (10.23) ▲	28.59 (4.16)	**56.88** (4.11)	74.26 (5.07) ▲	67.97 (4.47) ▲
	MGD-M-9	**87.84** (8.50) ▲	**81.04** (8.74) ▲	**32.45** (4.90) ▲	75.81 (10.21) ▲	**86.05** (7.79) ▲	**30.21** (3.74) △	55.91 (3.67)	74.50 (4.44) ▲	68.65 (4.51) ▲
	MGD-M-20	86.73 (7.24) ▲	80.99 (7.72) ▲	32.07 (4.23) ▲	**76.72** (9.64) ▲	82.92 (8.21) ▲	29.68 (3.32)	56.57 (3.48)	73.99 (4.61) ▲	68.50 (3.49) ▲

Learning Speed: Online Score

- ***MGD*** outperforms ***DBGD*** more when the noise in the feedback increases.
- In ***Perfect*** feedback: online performance for one dataset actually decreases compared to the baseline. This implies that while adding candidates increases offline performance, in the absence of feedback noise it may harm online performance.

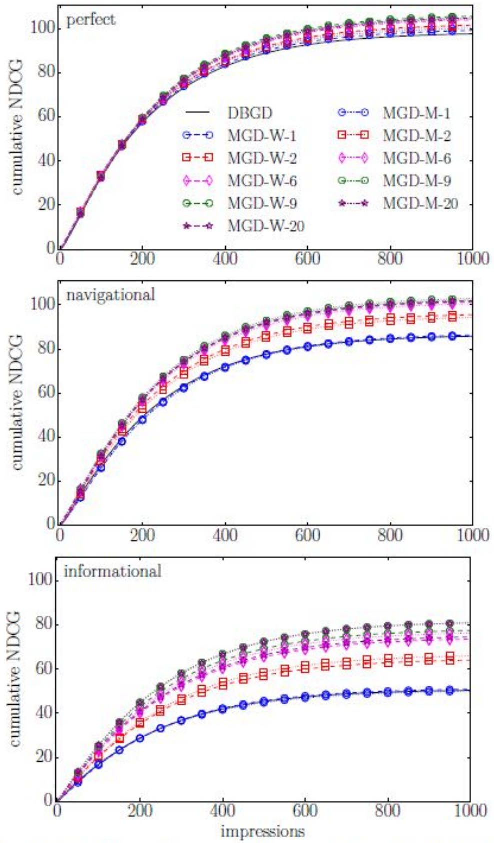

Figure 2: Online performance (discounted cumulative NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for *perfect*, *navigational* and *informational* click model instantiations.

Convergence

- **Both *MGD*** algorithms seem to converge to the same optimum but DBGD requires many more queries than MGD to do so.

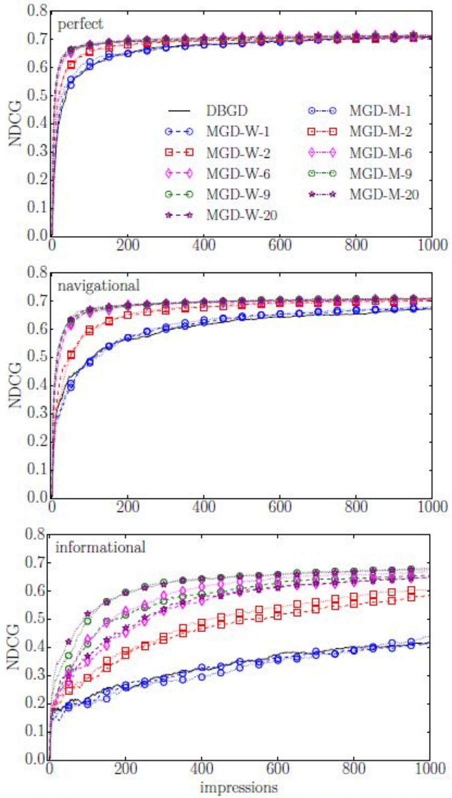

Figure 1: Offline performance (NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for the *perfect*, *navigational* and *informational* click model.

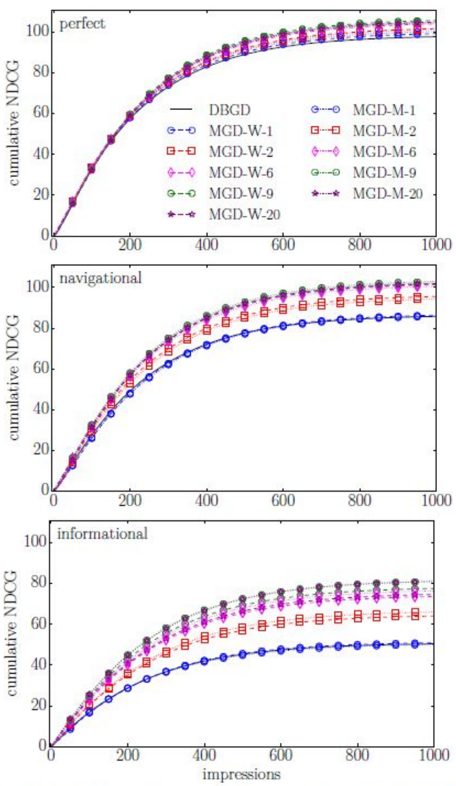

Figure 2: Online performance (discounted cumulative NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for *perfect*, *navigational* and *informational* click model instantiations.

Comparing outcome interpretations

How **MGD-W** and **MGD-M** compare to each other?

Offline Performance Comparison:

- No big difference between ***MGD-W*** and ***MGD-M*** for the ***perfect*** and ***navigational*** click models.
- ***MGD-M*** consistently outperforms ***MGD-W*** in the ***informational*** click models (***noiser*** feedback).

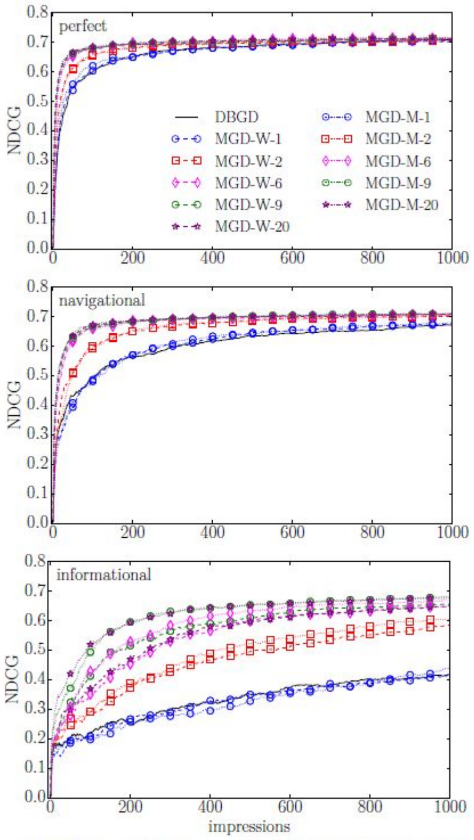

Figure 1: Offline performance (NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for the *perfect*, *navigational* and *informational* click model.

Comparing outcome interpretations

Online Performance Comparison:

- ***MGD-M*** also usually outperforms ***MGD-W.***
- When there is more noise in the feedback this effect (***MGD-M*** outperforms ***MGD-W.***) is more strong.
- Generally, ***MGD-M*** has lower standard deviation than ***MGD-W*** indicating that it is more stable.

Conclusion:

In general both ***MGD*** methods outperform ***DBGD***, ***MGD-M*** is better at handling high noise levels, making it more effective than ***MGD-W*** overall. The advantage of ***MGD-M*** over ***MGD-W*** comes from both the update direction and a smaller update size.

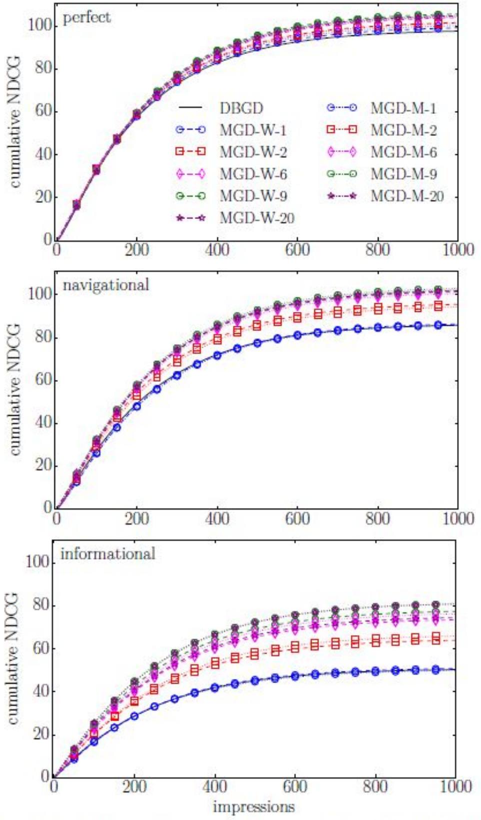

Figure 2: Online performance (discounted cumulative NDCG) on MGD-W and MGD-M with varying number of candidates compared to DBGD on *NP2003* dataset for *perfect*, *navigational* and *informational* click model instantiations.

Number of candidates

- **Both** offline and online performance increase with the number of candidates when noise is present.
- This effect appears to be limited by the length of the result list shown to users.

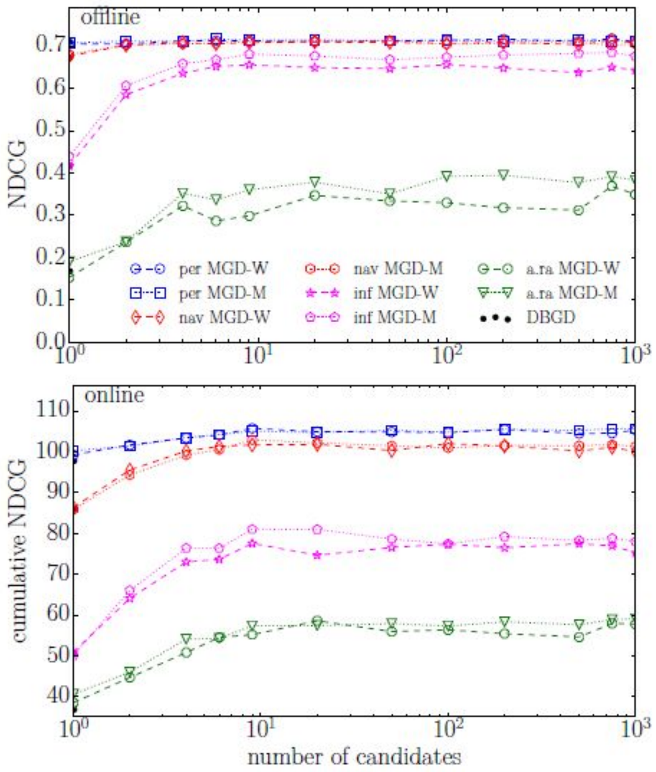

Figure 4: Sweep over the number of candidates in terms of offline and online performance for MGD-M and MGD-W after 1,000 impressions. Performed on *NP2003* using *all four* instantiations of the click model. DBGD is displayed by the black dots on the left axis. Note the log scale on the horizontal axis.

Learning rate

- ***DBGD*** and ***MGD*** have different optimal learning rates and that **MGD** can greatly outperform ***DBGD***, both offline and online, when the learning rate is chosen appropriately.

- $\alpha = 0.03$ be chosen as the learning rate in this paper.

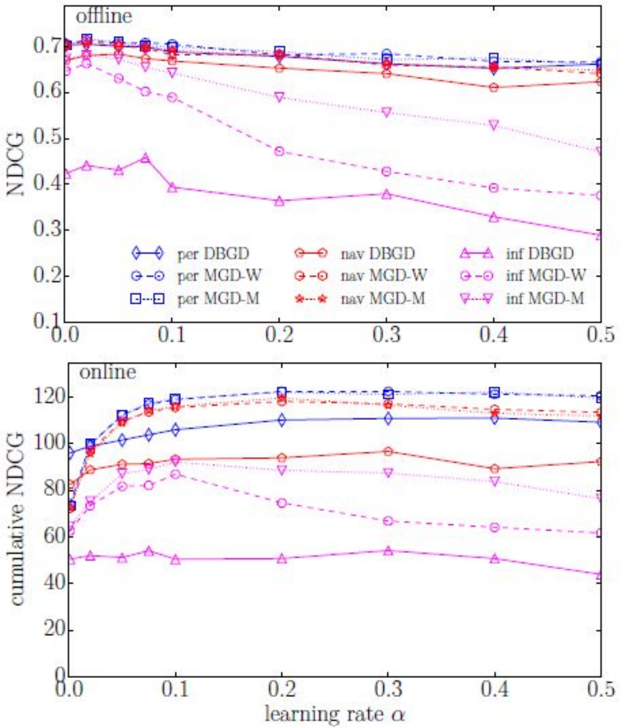

Figure 5: Sweep over learning rate values in terms of offline and online performance after 1,000 impressions for MGD-M and MGD-W with 9 candidates and DBGD. Performed on *NP2003* using three different click models with varying degrees of noise.

Conclusion

- An extension of DBGD: MGD
- Less user interaction data is required to find good ranker.
- Learning better rankers much faster and more advantages with noisy feedback.

Thanks !

Q&A