Vowpal Wabbit 2016

Kai-Wei Chang, Paul Mineiro, John Langford

http://hunch.net/~vw/

git clone
git://github.com/JohnLangford/vowpal_wabbit.git

What is Vowpal Wabbit

1. Large Scale linear regression (*)

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)
3. Active Learning (*)

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)
3. Active Learning (*)
4. Learning Reduction (*)

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)
3. Active Learning (*)
4. Learning Reduction (*)
5. Contextual Bandit Learning

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)
3. Active Learning (*)
4. Learning Reduction (*)
5. Contextual Bandit Learning
6. Logarithmic Time Classification

(*) Previous Tutorials online

What is Vowpal Wabbit

1. Large Scale linear regression (*)
2. Online Learning (*)
3. Active Learning (*)
4. Learning Reduction (*)
5. Contextual Bandit Learning
6. Logarithmic Time Classification
7. Joint Prediction

(*) Previous Tutorials online

Community

1. BSD license.

Community

1. BSD license.
2. Mailing list >500, Github >1K forks, >1K, >1K issues, >100 contributors

Community

1. BSD license.
2. Mailing list >500, Github >1K forks, >1K, >1K issues, >100 contributors
3. The official strawman for large scale logistic regression @ NIPS :-)

Community

1. BSD license.
2. Mailing list >500, Github >1K forks, >1K, >1K issues, >100 contributors
3. The official strawman for large scale logistic regression @ NIPS :-)
4. amazon AMERICAN EXPRESS AOL Baidu 百度 GraphLab facebook FTI CONSULTING IBM Microsoft YAHOO! Яндекс

Surface details

1. Automated test suite, github repository.

Surface details

1. Automated test suite, github repository.
2. VW supports all I/O modes: executable, library, port, daemon, service (see next).

Surface details

1. Automated test suite, github repository.
2. VW supports all I/O modes: executable, library, port, daemon, service (see next).
3. VW has a reasonable++ input format: sparse, dense, namespaces, etc... + JSON format

Surface details

1. Automated test suite, github repository.
2. VW supports all I/O modes: executable, library, port, daemon, service (see next).
3. VW has a reasonable++ input format: sparse, dense, namespaces, etc... + JSON format
4. Mostly C++, but bindings in other languages of varying maturity (python, C#, Java good).

Surface details

1. Automated test suite, github repository.
2. VW supports all I/O modes: executable, library, port, daemon, service (see next).
3. VW has a reasonable++ input format: sparse, dense, namespaces, etc... + JSON format
4. Mostly C++, but bindings in other languages of varying maturity (python, C#, Java good).
5. A substantial user base + developer base. Thanks to many who have helped.

An example

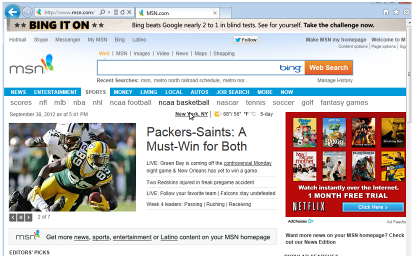
wget `http://hunch.net/~jl/VW_raw.tar.gz`

`vw -c rcv1.train.raw.txt -b 22 --ngram 2 --skips 4 -l 0.25 --binary` provides stellar performance in 12 seconds.

Next

1. Contextual Bandit Learning (John Langford)
2. Logarithmic Time Classification (Paul Mineiro)
3. Joint Prediction (Kai-Wei Chang)

Suppose you want to make decisions


http://www.msn.com/
MSN.com
★★ BING IT ON ★★
Hotmail Skype Messenger My MSN Bing Latino
Follow
Make MSN my homepage Welcome to M
Content options | Page options
Web | MSN | Images | Video | News | Maps | Shopping
msn
bing Web Search
Recent Searches: msn, metro north railroad schedule, metro nor...
Manage History
NEWS ENTERTAINMENT SPORTS MONEY LIVING LOCAL AUTOS JOB SEARCH MORE NOW
scores nfl mlb nba nhl ncaa football ncaa basketball nascar tennis soccer golf fantasy games
September 30, 2012 as of 5:41 PM
New York, NY 68°/ 56° °F °C 5-day
2 of 7
NETFLIX
Click Here ›
AdChoices
Ad Feedb
EDITORS' PICKS

Bing beats Google nearly 2 to 1 in blind tests. See for yourself. Take the challenge now.

Packers-Saints: A Must-Win for Both

LIVE: Green Bay is coming off the controversial Monday night game & New Orleans has yet to win a game.

Two Redskins injured in freak pregame accident

LIVE: Follow your favorite team | Falcons stay undefeated

Week 4 leaders: Passing | Rushing | Receiving

Watch instantly over the Internet. 1 MONTH FREE TRIAL

Get more news, sports, entertainment or Latino content on your MSN homepage

Want more news on your MSN homepage? Check out our News Edition

Repeatedly:

1. A user comes to Microsoft (with history of previous visits, IP address, data related to an account)
2. Microsoft chooses information to present (urls, ads, news stories)
3. The user reacts to the presented information (clicks on something, clicks, comes back and clicks again,...)

Microsoft wants to interactively choose content and use the

The Contextual Bandit Setting

For $t = 1, \ldots, T$:

1. The world produces some context $x \in X$
2. The learner chooses an action $a \in A$
3. The world reacts with reward $r_a \in [0, 1]$

Goal: Learn a good policy for choosing actions given context.

How do you test things?

Use format:
action:cost:probability | features
Example:
1:1:0.5 | tuesday year million short compan vehicl line stat financ commit exchang plan corp subsid credit issu debt pay gold bureau prelimin refin billion telephon time draw basic relat file spokesm reut secur acquir form prospect period interview regist toront resourc barrick ontario qualif bln prospectus convertibl vinc borg arequip

...

How do you train?

Training a deterministic policy

How do you train?

Training a deterministic policy

```
vw –cb 2 –cb_type dr rcv1.train.txt.gz -c
vw –cb 2 –cb_type ips rcv1.train.txt.gz -c
```

How do you train?

Training a deterministic policy

```
vw –cb 2 –cb_type dr rcv1.train.txt.gz -c
vw –cb 2 –cb_type ips rcv1.train.txt.gz -c
```

Training an exploration policy

```
vw –cb_explore 2 –epsilon 0.2 rcv1.train.txt -c
vw –cb_explore –cover 1 rcv1.train.txt -c
vw –cb_explore –bag 5 rcv1.train.txt -c
```

How do you train?

Training a deterministic policy

```
vw –cb 2 –cb_type dr rcv1.train.txt.gz -c
vw –cb 2 –cb_type ips rcv1.train.txt.gz -c
```

Training an exploration policy

```
vw –cb_explore 2 –epsilon 0.2 rcv1.train.txt -c
vw –cb_explore –cover 1 rcv1.train.txt -c
vw –cb_explore –bag 5 rcv1.train.txt -c
```

Datasets with Action Dependent Features (adf) work. Use `–cb_adf` or `–cb_explore_adf`

How do you evaluate exploration algorithms?

Method 1: With a supervised multiclass dataset

```
vw –cbify 2 –epsilon 0.2 rcv1.train.multiclass -c
vw –cbify 2 –cover 1 rcv1.train.multiclass -c
vw –cbify 2 –bag 5 rcv1.train.multiclass -c
```

How do you evaluate exploration algorithms?

Method 1: With a supervised multiclass dataset

```
vw –cbify 2 –epsilon 0.2 rcv1.train.multiclass -c
vw –cbify 2 –cover 1 rcv1.train.multiclass -c
vw –cbify 2 –bag 5 rcv1.train.multiclass -c
```

Method 2: With a CB dataset

```
vw –explore_eval –multiplier 0.1 –epsilon 0.2
  rcv1.train.multiclass_adf -c
```

–multiplier: smaller value means less bias (towards data collection policy) but higher variance.

Consider the decision service

http://aka.ms/mwt
Deploy a decision service system using VW in your Azure account.
Two apis: GetAction() and ReportReward().
Talk via JSON or use a client library.