Learning and predicting at scale

Nicolas Le Roux

Scientific Program Manager - R&D

2014-08-28

What does Criteo do?

- We buy advertising space on websites
- We display ads for our partners
- We get paid if the user clicks on the ad

How do we buy advertising space?

- Real-time bidding (RTB): it is an auction
- Second-price
- Optimal strategy: bid the expected gain
- Expected gain = CPC * CTR

What to do once we win the display?

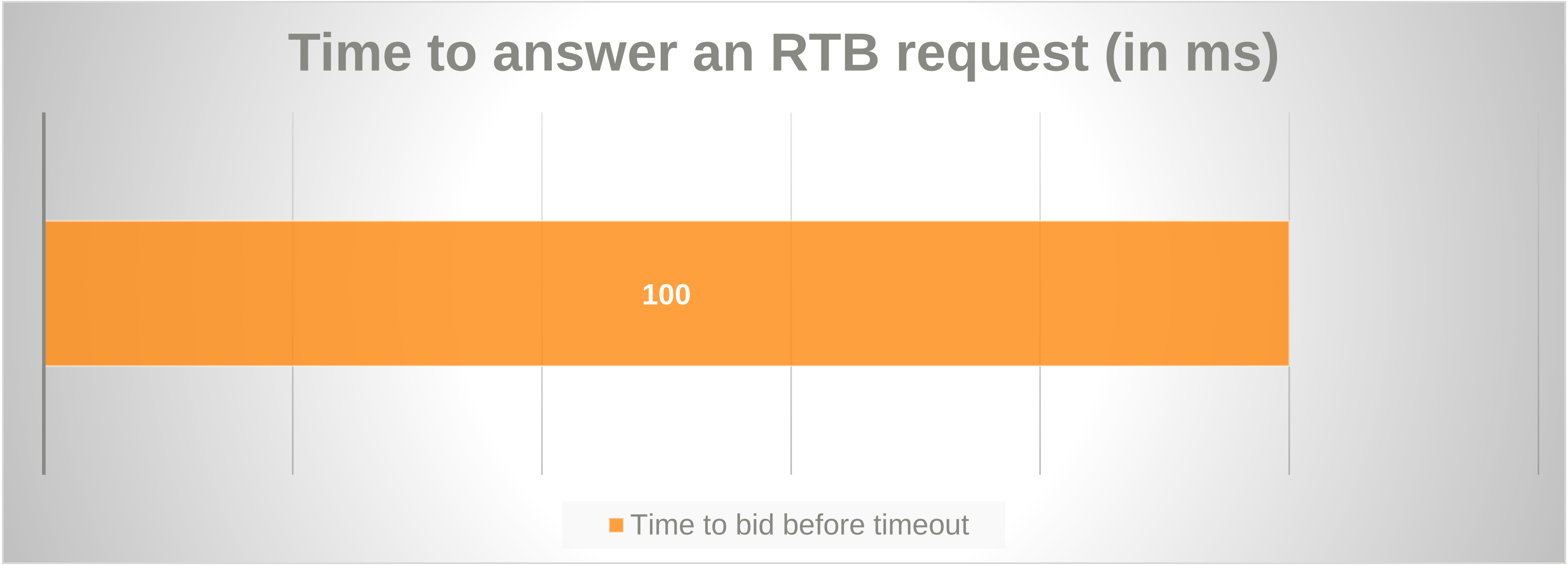
- Choose the best products
- Choose the color, the font and the layout
- Generate the banner

Real-time constraints at Criteo

- More than 2 billion banners displayed per day

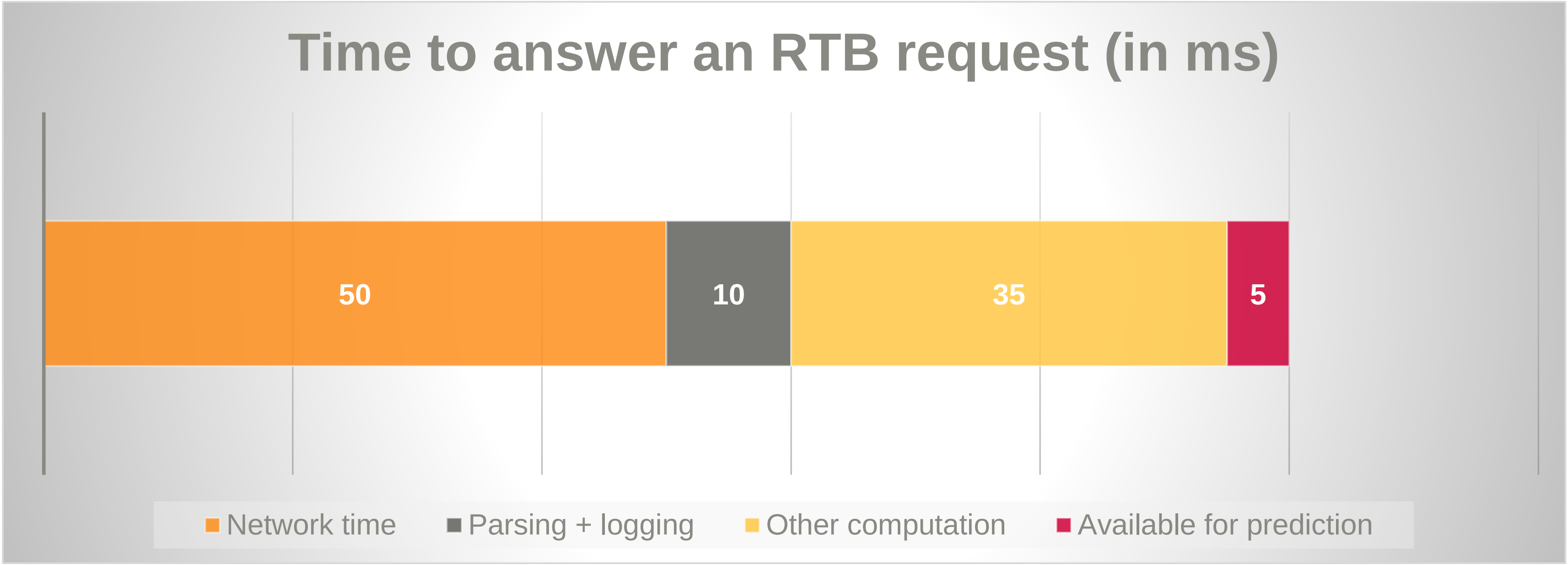
Real-time constraints at Criteo

- More than 2 billion banners displayed per day



Real-time constraints at Criteo

- More than 2 billion banners displayed per day



Predicting the CTR

- $P(click = 1|x) = \sigma(\theta^T x)$

Predicting the CTR

- $P(click = 1|x) = \sigma(\theta^T x)$
- x: features containing historical and contextual information
- θ: parameters of the model

Predicting the CTR

- $P(click = 1|x) = \sigma(\theta^T x)$

- x: [TimeSinceLastVisit, CurrentURL]

- θ: parameters of the model

A large number of parameters

- x: [TimeSinceLastVisit, CurrentURL]
- One parameter per modality

A large number of parameters

- x: [TimeSinceLastVisit, CurrentURL]
- One parameter per modality
  - TimeSinceLastVisit
    - Less than 10 seconds: [1, 0, 0]
    - Between 10 seconds and 5 minutes: [0, 1, 0]
    - More than 5 minutes: [0, 0, 1]

A large number of parameters

- x: [TimeSinceLastVisit, CurrentURL]
- One parameter per modality
  - TimeSinceLastVisit
    - Less than 10 seconds: [1, 0, 0]
    - Between 10 seconds and 5 minutes: [0, 1, 0]
    - More than 5 minutes: [0, 0, 1]
  - CurrentURL
    - *lemonde.fr : [1, 0, 0, ..., 0]*
    - *facebook.com : [0, 1, 0, ..., 0]*
    - *maisonetjardin.fr : [0, 0, 1, ..., 0]*
- x: [More Than 5 minutes, facebook.com] $= [0, 0, 1, 0, 1, 0, \ldots, 0]$

Modeling higher-order information

- A linear model cannot represent higher order information

Modeling higher-order information

- A linear model cannot represent higher order information
- E.g. CurrentUrl = "disney.com" and Advertiser = "Guns4Life"

Modeling higher-order information

- A linear model cannot represent higher order information
- E.g. CurrentUrl = "disney.com" and Advertiser = "Guns4Life"
- We model these by creating "cross-features"

Modeling higher-order information

- A linear model cannot represent higher order information
- E.g. CurrentUrl = "disney.com" and Advertiser = "Guns4Life"
- We model these by creating "cross-features"
  - CurrentUrl has p_1 modalities, Advertiser has p_2 modalities
  - The cross-feature has $p_1 p_2$ modalities

Hashing

- This model has estimation and computational issues

Hashing

- This model has estimation and computational issues
- Choose $h\colon \mathbb{N} \rightarrow \{1, \ldots, p\}$

Hashing

- This model has estimation and computational issues
- Choose $h\colon \mathbb{N} \to \{1, \dots, p\}$
- Replace $x_i = 1$ with $x_{h(i)} = 1$

Hashing

- This model has estimation and computational issues
- Choose $h\colon \mathbb{N} \to \{1, \dots, p\}$
- Replace $x_i = 1$ with $x_{h(i)} = 1$
- The original x is projected to $\mathbb{R}^p$

Hashing

- This model has estimation and computational issues
- Choose $h\colon \mathbb{N} \rightarrow \{1, \ldots, p\}$
- Replace $x_i = 1$ with $x_{h(i)} = 1$
- The original x is projected to $\mathbb{R}^p$
- $P(click = 1|x) = \sigma(\theta^T \tilde{x})$

Dealing with collisions

- There are many pairs (i_1, i_2) such that $h(i_1) = h(i_2)$
- These two features will become indistinguishable

Dealing with collisions

- There are many pairs (i_1, i_2) such that $h(i_1) = h(i_2)$
- These two features will become indistinguishable
- First solution: increase p

Dealing with collisions

- There are many pairs (i_1, i_2) such that $h(i_1) = h(i_2)$
- These two features will become indistinguishable
- First solution: increase p
- Second solution: do feature selection.

Feature selection

- About 40 original features

Feature selection

- About 40 original features
- 780 level-2 cross-features
- 9880 level-3 cross-features

Feature selection

- About 40 original features
- 780 level-2 cross-features
- 9880 level-3 cross-features
- Each feature contains many modalities

Feature selection at scale

- Greedy methods are too slow to be run to convergence
- They do provide a good initial set of features
- Group sparsity methods can refine the original set

Feature selection at scale

- Greedy methods are too slow to be run to convergence
- They do provide a good initial set of features
- Group sparsity methods can refine the original set
- Selecting the features is a way of learning the kernel.

Learning the parameters

- $n = 10^9$, $p = 10^7$
- Theory tells us that stochastic gradient methods should be used

Learning the parameters - The actual situation

- Tens of models are trained several times a day
- Warm starts favor batch methods
- Not all points are equal
- Stochastic methods are harder to parallelize.

Criteo's optimizer

- Batch optimizer
- Distributed computation of the gradients (10^7 examples/s)
- Update computation on a single node

Dealing with spurious clicks

- Some clicks do not bring sales
- Some clicks are pure fake
- We can build a fraud detection system

Dealing with spurious clicks

- Some clicks do not bring sales
- Some clicks are pure fake
- We can build a fraud detection system
- What is the real issue here?

Dealing with spurious clicks

- The real goal is to only buy clicks which bring sales

Dealing with spurious clicks

- The real goal is to only buy clicks which bring sales

- We have labeled data

Dealing with spurious clicks

- The real goal is to only buy clicks which bring sales
- We have labeled data
- Let's build a sale prediction model.

Summary for the prediction models

- Our models rarely use state-of-the-art techniques
- But they still work surprisingly well
- A production environment also adds constraints

Summary for the prediction models

- Our models rarely use state-of-the-art techniques
- But they still work surprisingly well
- A production environment also adds constraints
- Which latest developments can we incorporate and benefit from?

Product recommendation

- We have the list of products seen
- A catalog can contain 10^5 products
- How do we choose the right products to show?

Product recommendation

- We have the list of products seen
- A catalog can contain 10^5 products
- How do we choose the right products to show?
- In less than 20ms

Two-stage approach

- Stage 1: Product preselection based on:
  - Popularity
  - Browsing history of the user

Two-stage approach

- Stage 1: Product preselection based on:
  - Popularity
  - Browsing history of the user
- Stage 2: Exact scoring using a prediction model

Major hurdles for product recommendation

- Products come and go
- There might be a sale (Black Friday)
- Complementary vs. similar products

Other challenges

- Multiple products in a banner
- Interaction between products and layout
- Different timeframes for different products

A first recipe for success

- There are many sources of success/failure
- It is often suboptimal to focus on one
- The first step to address each source is often manual

Conclusion

- Prediction is at the core of our business
- Huge engineering constraints
- Different bottlenecks than in academia
- Build from the ground up, not the other way around

This is the end

Thank you!

Questions?