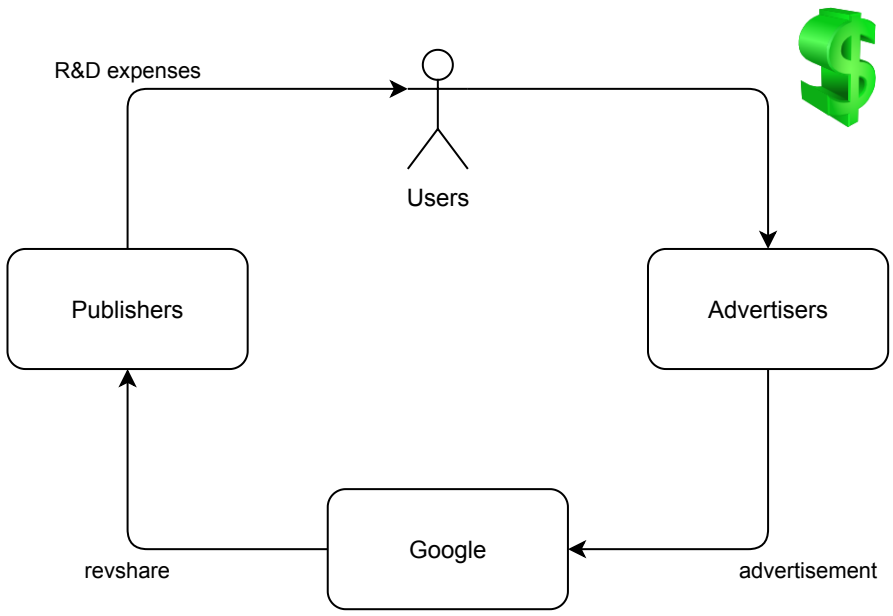
Resume

Hi,

My name is Dima and I am a software engineer currently working on Google Mobile Ads, specializing in all things machine learning. We build all sort of models for, just as an example, predicting the probabilities of clicks, installs, events, and the value to be received by advertisers.

To give you some perspective, the cashflow Google incorporated itself into originates with users and forms a loop:



What can we say about the user population P and its dynamics? Assuming that you run an advertising campaign with the budget proportional to your revenue, e.g. spending a dollar for every 5 dollars you receive, we arrive at

$$\frac{dP}{dt} = (\alpha - \delta)P,$$

where α is the acquisition rate, δ is the churn rate and P is the effective population. This formula can be improved substantially to introduce the population ceiling and reflect the real dynamics we observe in advertising campaigns to build up the flow and induce positive feedback, this is achieved by constructing models that model users → advertisers (maximization of the inflow) and publishers → users (maximization of publisher's retaining power) edges, both segments constituted my main interest in the recent years.

I wouldn't say I am a machine learning expert, I primarily construct neural networks on our internal AdBrain training infrastructure (with snippets in tensorflow), computing quantization ranges for training on TPUs, and serving on the serving stack developed and maintained by a separate team. I start with the regular structure: embeddings, ReLUs, the appropriate cap (Linear/Sigmoid/Poisson), adding SoftMax activations, multiple heads, setting stop gradients, gradient clipping, hyperparameter optimizations, all the usual jazz. So, I'm actually just writing the models themselves and taking measurements of the vital business metrics. I very rarely deal with custom loss functions, and I usually consult "experts" (we have separate Data Science folks) to help me design and build very advanced models. It would be best to think of me as of an engineer who can get the lowest hanging fruit rather quickly and then switch to another area once yields are low than of a guy who can work on the same thing for 10 years, slowly and steadily gaining the remaining 5% of accuracy through extremely advanced mathematical apparatus and reading/publishing an endless stream of scientific articles, although I'm very much comfortable working alongside such folks. I am fully capable of constructing a very large class of various neural networks, not just for prediction, explore/exploit problems, and picking the best choice out of many available options, but also for measuring the attrition rates, user growth, determining advertiser competitors, user clusters, modeling events that are substantially delayed. Many of the problems I deal with are quite complex, to the point where it takes multiple iterations to get to a satisfactory quality of predictions or estimates.

I trained gmob's pCTR model for several years (you probably know gmob under the name AdMob, which also encompasses other stacks from Google's past acquisitions), along with the pCTR calibration to control the monetary damage that can occur during CPC-to-CPM conversion. The company division I'm in makes more than a billion dollars a week (all numbers are publicly available on the company's earning report), you can imagine what sort of scale we operate on, training set worth petabytes of data, billions of weights in the model, endlessly growing number of advertisers and publishers.

What exactly sort of problems do I usually deal with? Obviously, figuring out the right features (including sensitive user features), their construction, evaluation, and production launches. There is a lot can be done here to make the inclusion process to be completely automatic, leaving opportunity for mere engineers to quickly construct any evaluate any feature they can think of. Measuring what

advertiser-user clusters we have and maximizing the flowing spend to the mutual benefits of all involved parties. Measuring the epistemic uncertainty of just started campaigns to address cold start problems, when you need to sample from the clustered space in proportion to user-dollars in each cluster, to ensure that each advertiser-user cluster is picked up by the primary value-measuring models. Automatic handling of available choices, such as choosing the best ad format and ad rendering parameters among many possible (as drawn by designers) to optimize for revenue and retention metrics, there are multiple approaches that we use for these explore-exploit challenges. I've been constructing all sort of anomaly detection and metric measurements dashboards.

I also handle customer escalations, root-causing and fixing the faulty components. I became fairly proficient at ensuring the safe mode of operation and what it takes to keep things running smoothly.

Languages and libraries? I've been pretty much through all of them, there were periods of my life when I used Assembly, C++, Top Speed Clarion, Clipper Summer 87, Borland Pascal, C#, F#, Java, Clojure, Scala, Haskell, PHP, Javascript, Go, etc... At Google I actively use Google C++ (don't confuse it with the actual C++, all our libraries are custom-made and we have all sort of restrictions and limitations on the language features we are allowed to use), Python for modelling and colabs (Jupyter Notebook). My current favorite? Probably, typescript, which allows for creation of ubiquitous tfjs models that can train and run everywhere, on desktops, in browsers and on mobile devices. In such circumstances you cannot serve big models since on-device chip does't yield the throughput we get on TPUs, though they are powerful enough for many practical tasks.

Well, what else? I'm not looking for any career changes, nor do I care about promotions, I'd rather just do modeling and write code, in that order. Google is an excellent company to work for; with many truly great engineers working with me, so I feel very hesitant to leave. On the other hand, it's been 6.5 years working on the same team for the same company, and I'm a bit curious what I could be doing in different settings, especially if it is accompanied with a non-trivial increase in pay with comparable perks and benefits.

I worked for Microsoft prior to joining Google, was training decision trees and truncation neural net for the Bing local ranking team. And prior to that I was working on the Point of Sales/Axapta ERP, which at this point I remember very vaguely about, it's being well over 10 years ago.

You can get in touch with me via Linked-In or directly at Dmitry.Belikov@gmail.com.

Cheers,
Dima