News recommendation systems at BBC

Felix Mercer Moss
Senior Data Scientist

INRA, Copenhagen, Denmark
18.09.2019

BBC datalab

Today

1. What is unique about recommendations for BBC?
2. BBC Mundo case study
3. What we have learned and are our next steps

What is unique about recommendations for BBC?

The BBC makes a lot of content

- Been delivering content since 1922
- Reaches > 90% of adult UK population every week.
- 426 million adults reached worldwide every week.
- > 50,000 content brands available.
- 1000s new items produced every day.
- News published in > 40 languages

Surfacing content

- Inform, educate & entertain
- Political non-partisanship
- Maintain a trusted voice and reputation as authoritative source of information

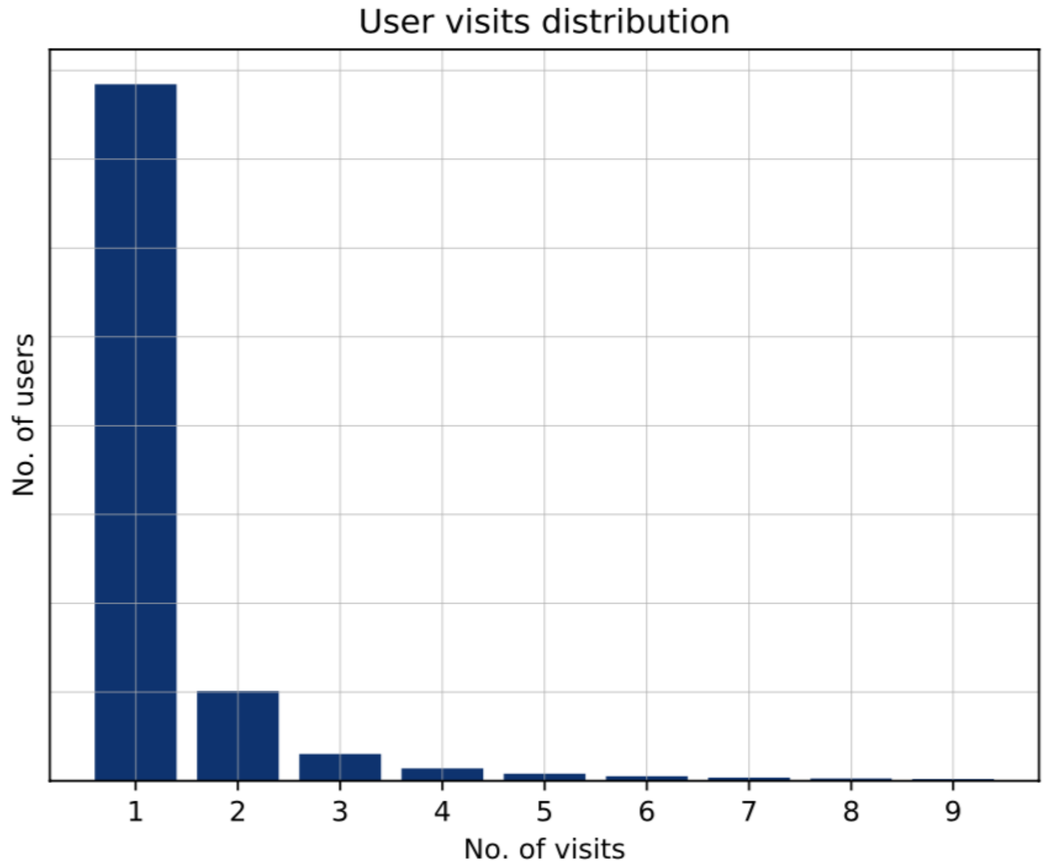
Our team’s challenge:

Bring the BBCs trusted editorial values into the algorithmic age.

Case Study

- BBC News for Latin America
- Non signed-in users
- Many cold start users
- Many cold start items

Why do we need algorithmic recommendations?



El reclamo de Apple a Google por la forma en la que informó sobre las fallas de seguridad del iPhone

Dave Lee
Corresponsal de Tecnología, BBC

6 horas

Compartir

Según Apple, el hackeo de iPhones denunciado por Google se concentró exclusivamente en la comunidad uigur, una minoría étnica en la mira de las autoridades chinas.

Quizás también te interese

Cómo es HarmonyOS, el sistema operativo lanzado por Huawei para sustituir a Android en sus celulares

Por qué es un mito que los teléfonos nos escuchan en secreto

Cuánto le cuestan a Apple los componentes de un iPhone (en comparación con lo que pagas)

Chongseo 1.0, el software desarrollado por Corea del Norte para enseñar ideología política

Our overall approach

Quantitative benchmarking

Baselines
- Random
- Most Recent
- Most Popular
- LDA Nearest Neighbour

Models
- Weighted Average LDA
- Rank optimised neural network
- Cosine collaborative filtering

Qualitative benchmarking

Baselines
- LDA Nearest neighbour

Models
- Wik2Vec Nearest Neighbour
- RDF2Vec Nearest Neighbour

Online Multi Variate Testing

Baselines
- Most Popular

Models
- LDA Nearest Neighbour
- Weighted Average LDA

Today’s focus

Quantitative benchmarking

Baselines
- Random
- Most Recent
- Most Popular
- LDA Nearest Neighbour

Models
- Weighted Average LDA
- Rank optimised neural network
- Cosine collaborative filtering

Architectural constraints

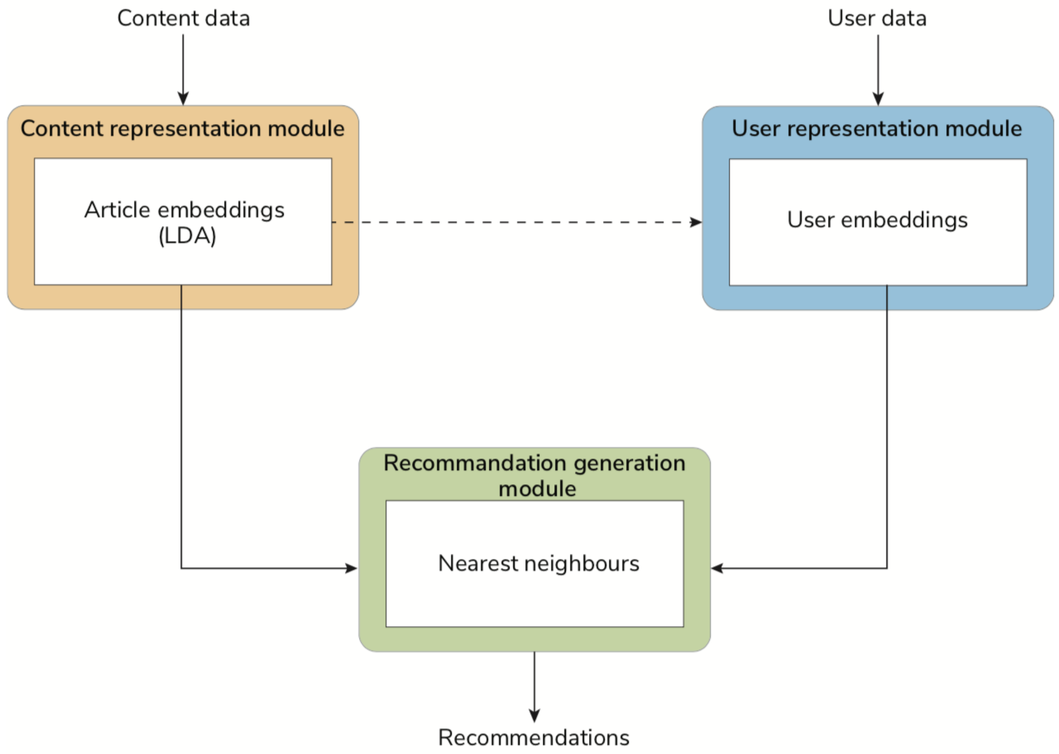
Content Representation Module

•LDA generation infratructure in place.

User Representation module

•Maximum of two articles are available at serve time

•No signed in user data – no demographic data



Baselines

- Random: List of randomly selected articles in test period.
- Most popular: List of most popular articles during train period.
- Recency: List of most recent articles from the end of the train period.
- LDA Nearest Neighbour: Derive nearest neighbours to the current article in LDA vector space.

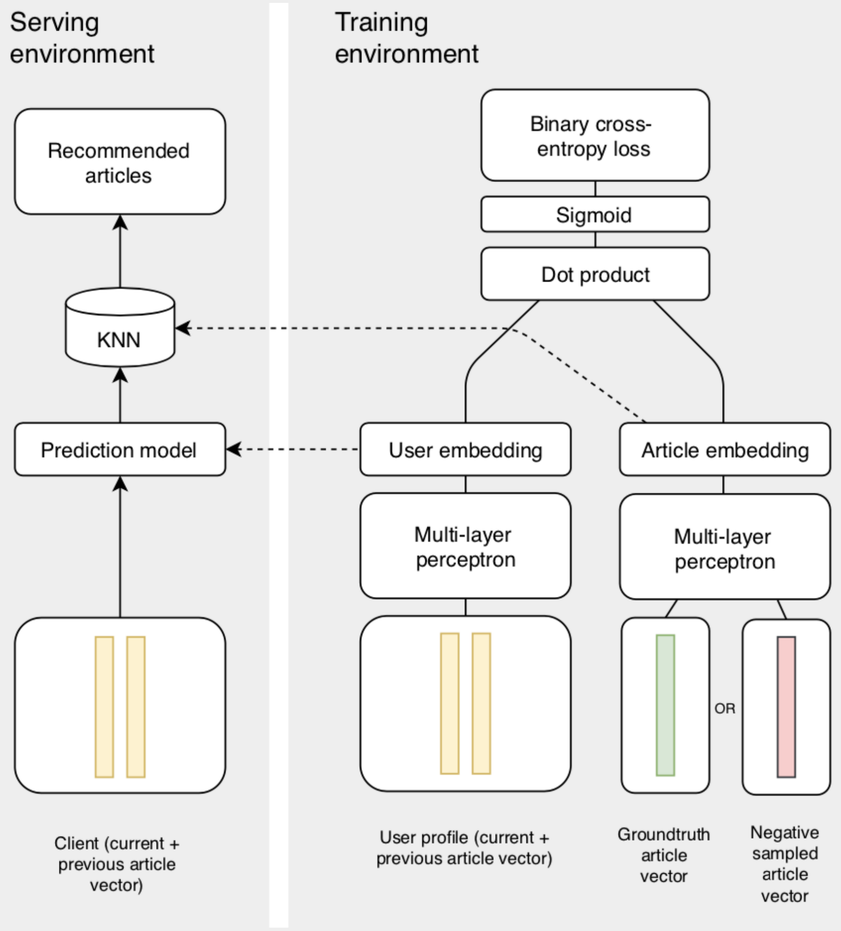
Models

- Weighted average LDA: Take a weighted average of the current and previous article then derive nearest neighbours.
- Rank optimised neural network: List nearest neighbours to the embedding layer output of a deep neural network.
- Cosine collaborative filtering: Item to item similarity derived from users that have interacted with both.

Rank optimised neural network

Design a network that finds an embedding space that **minimises the inner product between a user profile and the most appropriate article**.

- **Deeper** = better
- MLP has **5 hidden layers** (1024, 512 256, 128, 75 nodes).
- **Dropout** not helpful.
- **Batch normalisation** halved convergence time and significantly improved performance.



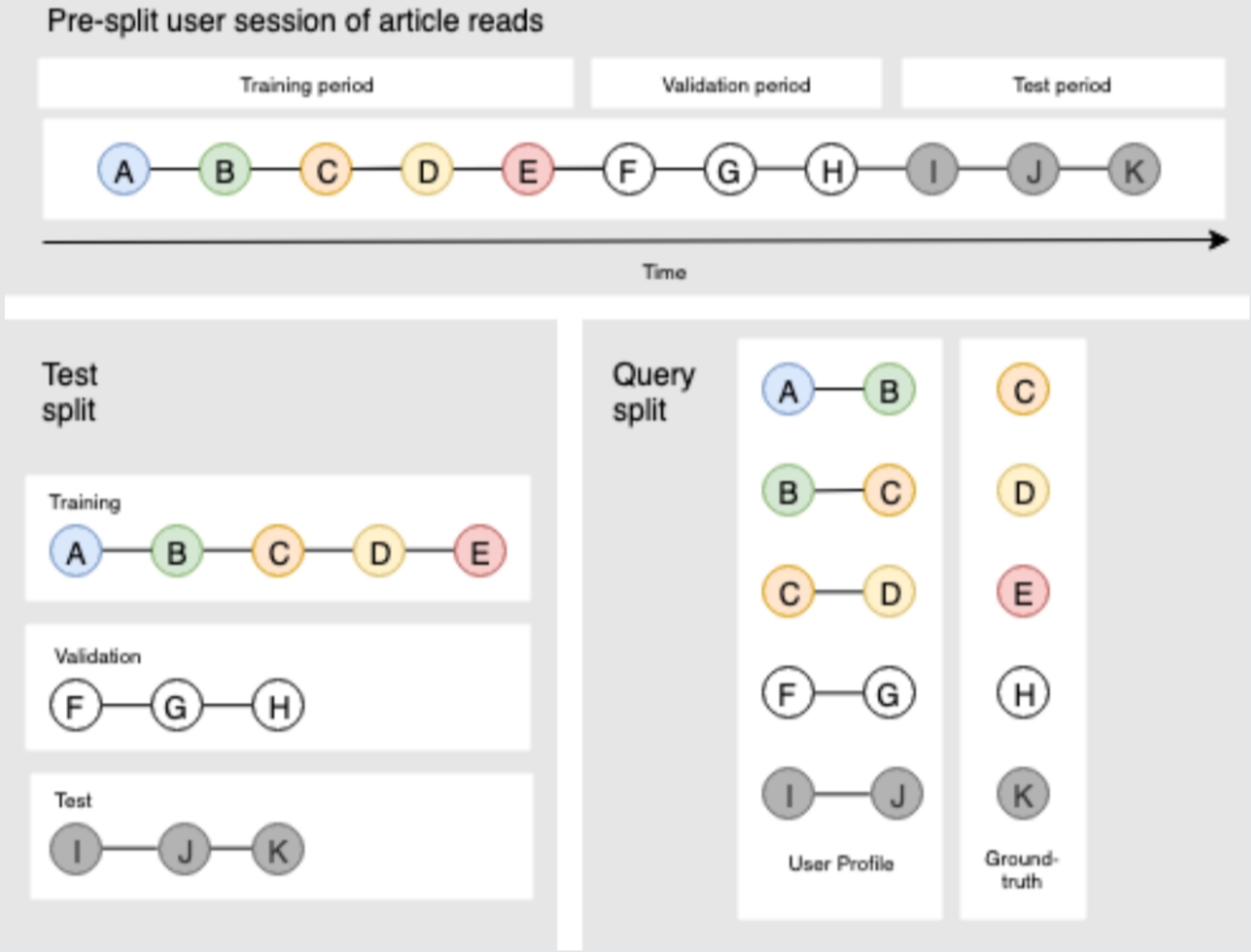
Data preparation

Test split

- 13 days **train**;
- 1 day **validation**;
- 1 day **test**.

Model input

- Journeys split into **trigrams**
- First two articles used to predict the third



Evaluation

Accuracy

- Hit rate *
- nDCG *

Diversity

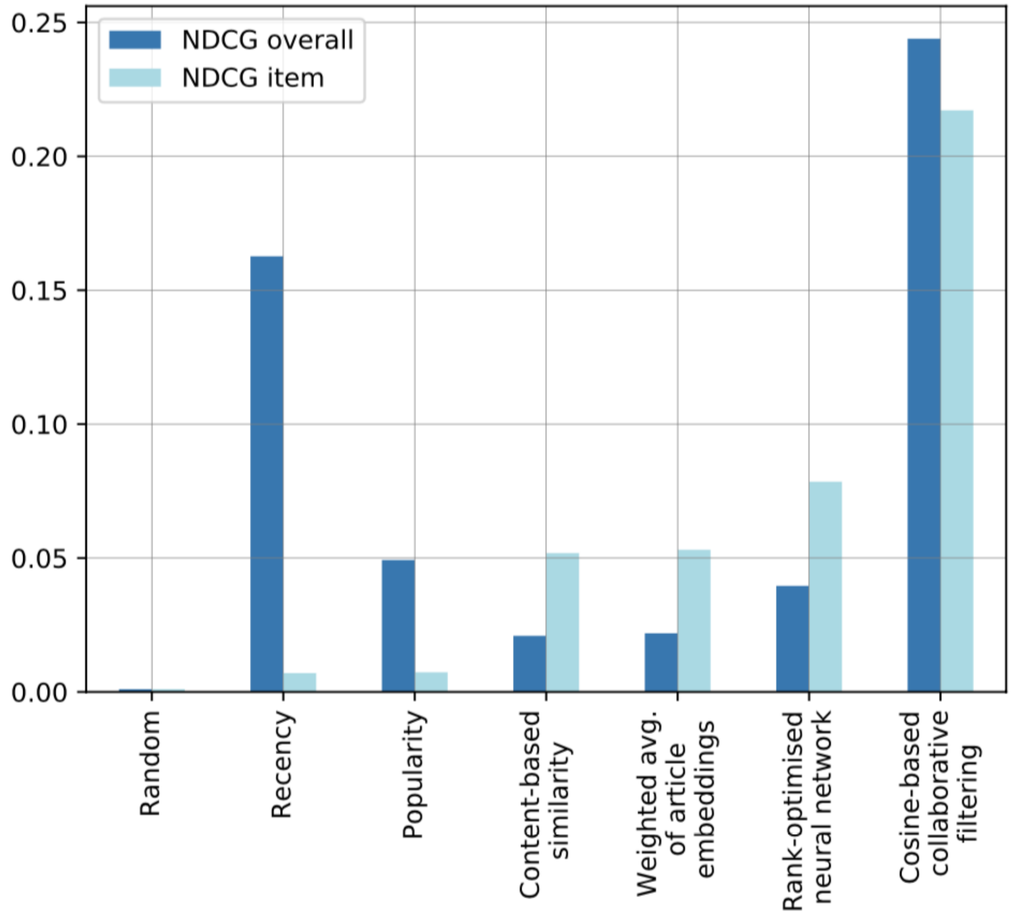
- Inter-list diversity
- Intra-list diversity *
- Surprisal *

Temporal

- Recency *

* For these metrics we calculate an **overall metric** and an **item-normalised metric which factors out the bias of the most popular items**.

- **CF solution** performed significantly **better than all others.**
- **Recency outperformed** most **popular**.
- **LDA-based models outperformed** non-LDA baselines, in item–normalised.
- **Neural network** approach produced **highest LDA performance**.



Other metrics

Recommender System	Hitrate	NDCG	Intra-list diversity	Inter-list diversity	Surprisal	Recency
Random baseline	0.005	0.001	**1.192**	**0.995**	0.430	0.010
Recency baseline	0.695	0.163	1.175	0.000	0.000	**0.975**
Popularity baseline	0.315	0.049	1.170	0.000	0.000	0.495
Content similarity baseline	0.085	0.021	0.641	0.968	0.790	0.018
Weighted average of item embeddings	0.065	0.022	0.641	0.968	**0.790**	0.018
Cosine-based collaborative filtering	**0.741**	**0.244**	1.154	0.584	0.480	0.512
Rank-optimised neural network	0.128	0.040	0.909	0.731	0.781	0.036

What we have learnt and next steps

Lessons learnt

- Three implemented **models all out-perform simple baselines** of random, recency and most popular.
- Traditional **collaborative filtering techniques perform very well** on our dataset (but at cost of personalisation) – provide us with a target for hybrid models.
- Inclusion of **previous article provided steady but modest improvement** in performance.
- **Item-normalising** scores provides a **good way of removing the popularity bias** in evaluation.
- **Recency** has a significant **positive impact** upon performance.

Next steps

Models

- Pairwise and list-wise neural architectures.
- Combine additional metadata to the text embedding: context, image thumbnails
- Collaborative topic regression
- New embeddings: e.g. wiki2vec, rdf2vec.

Deployment

- Qualitative evaluation
- Multi-variate tests.
- Automated retraining

New products

Acknowledgements

Maria Panteli

Alessandro Piscopo

Adam Harland

Jon Tutcher

Thank you.

Tak.

https://findouthow.datalab.rocks/