Neural Networks for Information Retrieval

Tom Kenter[1], Alexey Borisov[1,2], Christophe Van Gysel[1], Mostafa Dehghani[1], Maarten de Rijke[1], and Bhaskar Mitra[3,4]

[1] University of Amsterdam, Amsterdam, The Netherlands
tom.kenter@uva.nl, cvangysel@uva.nl, dehghani@uva.nl, derijke@uva.nl
[2] Yandex, Moscow, Russia
alborisov@yandex-team.ru
[3] Microsoft, Cambridge, UK
bmitra@microsoft.com
[4] University College London

Title and length

- **Title**: Neural Networks for Information Retrieval
- **Length**: Full day

Detailed contact information

Tom Kenter Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands.
Alexey Borisov Yandex, Lva Tolstogo 16, 119021 Moscow, Russia.
Christophe Van Gysel Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands.
Mostafa Dehghani Institute for Logic, Language and Computation, University of Amsterdam, Science Park 107, 1098 XG Amsterdam, The Netherlands.
Maarten de Rijke Informatics Institute, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands.
Bhaskar Mitra Microsoft Cambridge, 21 Station Road, Cambridge CB1 2FB, United Kingdom.

Brief biographies

Tom Kenter has a computational linguistics background and worked at several IR and data mining companies prior to starting his PhD at the University of Amsterdam, supervised by Maarten de Rijke. He did two internships at Google Research in Mountain View, one on character-level sequence-to-sequence models and one on semi-supervised learning in a dialog setting.

Alexey Borisov is an applied researcher at Yandex. He is pursuing a (part-time) doctorate at the University of Amsterdam under the supervision of Maarten de Rijke. His research interests lie at the intersection of deep learning and IR/NLP: modeling user behavior, semantic matching, conversational systems. He received the SIGIR 2016 best student paper award for work on (neural) modeling of times between user actions.

Christophe Van Gysel is a post-doctoral fellow at the University of Amsterdam. He obtained his doctorate degree at the University of Amsterdam, where he was supervised by Evangelos Kanoulas and Maarten de Rijke. In his doctorate research, he studied neural unsupervised representation learning for IR. His research interests include IR, machine learning, speech processing, web security and distributed systems. He did multiple internships, at Google, Facebook, Apple, Microsoft and Snap Inc.

Mostafa Dehghani is a PhD student at the University of Amsterdam. His doctorate research lies at the intersection of IR and machine learning, in particular employing neural models for core IR problems like ranking and representation learning. He has recently done an internship at Google Research on search conversationalization using sequence-to-sequence models.

Maarten de Rijke is a Professor of Computer Science at the Informatics Institute of the University of Amsterdam. Together with a team of PhD students and postdocs he works on problems in deep learning and on- and offline learning for IR. Recent tutorials include SIGIR 2015, 2016, 2017, WSDM 2016, RUSSIR 2016, ESSIR 2015, 2017.

Bhaskar Mitra is a Senior Applied Scientist at Bing in Microsoft Cambridge. He started at Bing in 2007 (then called Live Search) at the Search Technology Center in India. His research interests include representation learning and neural networks, and their applications to IR. He co-organized multiple workshops (at SIGIR 2016 and 2017) and tutorials (at WSDM2017) on neural IR, and served as guest editor for an IRJ special issue on the same topic.

Short abstract

Recent advances in deep learning have seen neural networks being applied to all key parts of the modern IR pipeline, such as core ranking algorithms, click models, query autocompletion, query suggestion, knowledge graphs, text similarity, entity retrieval, question answering, and dialogue systems. The fast pace of modern-day research has given rise to many different architectures and paradigms, such as auto-encoders, recursive networks, recurrent networks, convolutional networks, various embedding methods, deep reinforcement learning, and, more recently, generative adversarial networks, of which most have been applied to IR settings. The amount of information available can be overwhelming both for junior students and for experienced researchers looking for new research topics and directions. The aim of the tutorial is to provide an overview of the main network architectures currently applied in IR and to show explicitly how they relate to previous work and how they benefit IR research. Additionally, key insights into IR problems that the new technologies give us are provided. The tutorial covers methods employed in industry and academia, with in-depth insights into the underlying theory, core IR tasks, applicability, key assets and handicaps, scalability concerns and practical tips & tricks. We expect the tutorial to be useful both for academic and industrial researchers and practitioners who want to develop new neural models, use them in their own research in other areas or apply the models described here to improve actual IR systems.

Previous offerings

NN4IR was first presented at SIGIR 2017, Tokyo Japan [24] by the authors of this proposal. We build on this first iteration of the tutorial and incorporate lessons learnt. Based on user feedback, we reduce the material on semantic text matching to one session instead of two, and add a session on recommender systems. Also, given the importance of the topic in industry, we include an extra session devoted to entities and an extra session on insights from industry. Finally, we add a brief section to every session, devoted to efficiency issues regarding the methods presented.

Extended abstract

Machine learning plays a role in many aspects of modern IR systems, and deep learning is applied in all of them. The fast pace of modern-day research has given rise to many approaches to many IR problems. The amount of information available can be overwhelming both for junior students and for experienced researchers looking for new research topics and directions. The aim of this full-day tutorial is to give a clear overview of current tried-and-trusted neural methods in IR and how they benefit IR.

Motivation

Prompted by the advances of deep learning in computer vision, neural networks (NNs) have resurfaced as a popular machine learning paradigm in many other directions of research, including IR. Recent years have seen NNs being applied to all key parts of the typical modern IR pipeline, such as click models, core ranking algorithms, dialogue systems, entity retrieval, knowledge graphs, language modeling, question answering, and text similarity.

A key advantage that sets NNs apart from many learning strategies employed earlier, is their ability to work from raw input data. Where designing features used to be a crucial aspect and contribution of newly proposed IR approaches, the focus has shifted to designing network architectures instead. As a consequence, many different architectures and paradigms have been proposed, such as auto-encoders, recursive networks, recurrent networks, convolutional networks, various embedding methods, and deep reinforcement learning. The aim of this tutorial is to provide an overview of the main network architectures currently applied in IR and to show how they relate to previous work. The tutorial covers methods applied in industry and academia, with in-depth insights into the underlying theory, core IR tasks, applicability, key assets and handicaps, efficiency and scalability concerns, and tips & tricks.

We expect the tutorial to be useful both for academic and industrial researchers and practitioners who either want to develop new neural models, use them in their own research in other areas or apply the models described here to improve actual IR systems.

Brief outline of the topics to be covered

Table 1 gives an overview of the time schedule of the tutorial. The total time is 6 hours, plus breaks. We bring a team team of six lecturers, all with their specific

Table 1: Time schedule for NN4IR tutorial

Morning		Afternoon	
Preliminaries	45 min.	Recommender systems	45 min.
Semantic matching	45 min.	Modeling user behavior	45 min.
Learning to rank	45 min.	Generating responses	45 min.
Entities	45 min.	Industry insights	45 min.

areas of specialization. Each session will have two expert lecturers (indicated by their initials below) who will together present the session.

Preliminaries *[TK, MdR]* The recent surge of interest in deep learning has given rise to a myriad of model architectures. Different though the inner structures of NNs can be, many building blocks are shared. In this preliminary session, we focus on key concepts, all of which will be referred to multiple times in subsequent sessions. In particular we will cover distributed representations/embeddings [35], fully-connect layers, convolutional layers [25], recurrent networks [34] and sequence-to-sequence models [43].

Semantic matching *[CVG, BM]* The problem of matching items based on textual descriptions arises in many retrieval systems. The traditional IR approach involves computing lexical term overlap between query and document [40]. However, a vocabulary gap occurs when query and documents use different terms to describe the same concepts [29]. Semantic matching methods bridge the vocabulary gap by matching concepts rather than exact word occurrences. Neural network-based methods that provide a semantic matching signal come in supervised, semi-supervised, and unsupervised flavours. In the supervised setting, explicit (e.g., human-labelled relevance judgements [31, 36]) or implicit labels (e.g., clicks [19, 37]) are available. In semi-supervised learning, domain-specific or external information is used to generate pseudo-relevance labels that are subsequently used to train a supervised approach [11]. Unsupervised methods learn semantic representations without relevance labels by either combining pre-trained word representations [13, 21, 51, 55, 56, 58] or learning representations from scratch [2, 23, 27, 45–47].

Learning to rank *[AB, MD]* Capturing the notion of relevance for ranking needs to account for different aspects of the query, the document, and their relationship. Neural methods for ranking can use manually crafted query and document features, and combine them with regards to a ranking objective. Moreover latent representations of the query and document can be learnt in situ. We

cover scenarios with different levels of supervision—unsupervised [41, 45, 46], semi/weakly-supervised [11, 44], or fully-supervised using labeled data [36] or interaction data [19].

Entities *[CVG, TK]* Entities play a central role in modern IR systems [12]. We cover neural approaches to solving the basic task of named entity recognition [8, 10, 26], as well learning representations in an end-to-end neural model for learning a specific task like entity ranking for expert finding [46], product search [45] or email attachment retrieval [48]. Furthermore, work related to knowledge graphs will be covered, such as graph embeddings [5, 53, 57].

Recommender systems *[MdR, BM]* Deep learning has also found its way into recommender systems. We cover learning of item (products, users) embeddings [4, 15, 49], as well as deep collaborative filtering using different deep learning techniques and architectures [7, 52]. Furthermore, NN-based feature extraction from content (such as images, music, text) [3, 32, 38], and session-based recommendations with RNNs [18, 39] will be covered.

Modeling user behavior *[AB, MdR]* Modeling user browsing behavior plays an important role in the development of modern IR systems. Accurately interpreting user clicks is difficult due to various types of bias. Over the last decade, many click models based on Probabilistic Graphical Models (PGMs) have been proposed [9]. Such click models can only model patterns that are explicitly encoded in the PGM. Recently, it was shown that recurrent neural networks can learn to account for biases in user clicks directly from the click-through data, i.e., without the need for a predefined set of rules as is customary for PGM-based click models [6]. Additionally, there are similar biases in click dwell times, which the neural approach can account for too.

Generating responses *[TK, MD]* Recent inventions such as smart home devices, voice search, and virtual assistants provide new ways of accessing information. They require a different response format than the classic ten blue links.

Examples are conversational and dialog systems [30, 50] or machine reading and question answering tasks where the model either infers the answer from unstructured data, like textual documents that do not necessarily feature the answer literally [16, 17, 22, 42, 54], or generates natural language given structured data, like data from knowledge graphs or from external memories [1, 14, 28, 33].

Industry insights *[AB, BM]* Where the focus of academic papers can be on a specific subtask, industry approaches have to ensure that a system works from start to end. As a result, extra challenges are involved concerning the user experience. For example in Google's SmartReply system [20] the neural model at the core of the system is embedded in a much larger framework of non-neural methods to make sure quality and efficiency requirements are met.

In this session, lessons learned from industry are shared and discussed.

Support materials supplied to attendees

Slides Slides will be made publicly available on http://nn4ir.com.
Bibliography An annotated compilation of references will list all work discussed in the tutorial and should provide a good basis for further study.
Code Apart from the various open source neural toolkits (Tensorflow, Theano, Torch) many of the methods presented come with implementations released under an open source license. These will be discussed as part of the presentation of the models and algorithms. We provide a list pointers to available code bases.

Intended audience

– Intermediate level
– Familiarity with information retrieval terminology, basics of machine learning and neural networks. No special skills are required.

Bibliography

[1] S. Ahn et al. A neural knowledge language model. *arXiv preprint arXiv:1608.00318*, 2016.
[2] Q. Ai et al. Improving language estimation with the paragraph vector model for ad-hoc retrieval. In *SIGIR*, pages 869–872. ACM, 2016.
[3] T. Bansal, D. Belanger, and M. A. Ask the GRU: Multi-task learning for deep text recommendations. In *RecSys*, 2016.
[4] O. Barkan and N. Koenigstein. ITEM2VEC: Neural item embedding for collaborative filtering. In *MLSP*, 2016.
[5] A. Bordes, J. Weston, R. Collobert, and Y. Bengio. Learning structured embeddings of knowledge bases. In *Conference on artificial intelligence*, 2011.
[6] A. Borisov et al. A neural click model for web search. In *WWW*, pages 531–541, 2016.
[7] H. Cheng et al. Wide & deep learning for recommender systems. In *DLRS*, 2016.
[8] J. P. Chiu and E. Nichols. Named entity recognition with bidirectional LSTM-CNNs. *TACL*, 2015.
[9] A. Chuklin, I. Markov, and M. de Rijke. *Click Models for Web Search*. Morgan & Claypool, 2015.
[10] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*, pages 160–167. ACM, 2008.
[11] M. Dehghani et al. Neural ranking models with weak supervision. In *SIGIR*, 2017.
[12] L. Dietz, A. Kotov, and E. Meij. Utilizing knowledge bases in text-centric information retrieval. In *ICTIR*, pages 5–5. ACM, 2016.

[13] D. Ganguly et al. Word embedding based generalized language model for information retrieval. In *SIGIR*, pages 795–798. ACM, 2015.
[14] A. Graves, G. Wayne, and I. Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.
[15] M. Grbovic et al. E-commerce in your inbox: Product recommendations at scale. In *KDD*, 2015.
[16] K. M. Hermann et al. Teaching machines to read and comprehend. In *NIPS*, pages 1693–1701, 2015.
[17] D. Hewlett et al. WIKIREADING: A novel large-scale language understanding task over Wikipedia. In *ACL*, 2016.
[18] B. Hidasi et al. Session-based recommendations with recurrent neural networks. In *ICLR*, 2016.
[19] P.-S. Huang et al. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*, pages 2333–2338. ACM, 2013.
[20] A. Kannan et al. Smart Reply: Automated response suggestion for email. In *KDD*, 2016.
[21] T. Kenter and M. de Rijke. Short text similarity with word embeddings. In *CIKM*, pages 1411–1420. ACM, 2015.
[22] T. Kenter and M. de Rijke. Attentive memory networks: Efficient machine reading for conversational search. In *Workshop on Conversational Approaches to Information Retrieval (CAIR'17) at SIGIR 2017*, 2017.
[23] T. Kenter, A. Borisov, and M. de Rijke. Siamese CBOW: Optimizing word embeddings for sentence representations. In *ACL*. ACL, 2016.
[24] T. Kenter et al. Neural networks for information retrieval. In *SIGIR*, pages 1403–1406. ACM, 2017.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012.
[26] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. Neural architectures for named entity recognition. In *NAACL-HLT*, 2016.
[27] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*, 2014.
[28] R. Lebret, D. Grangier, and M. Auli. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*, 2016.
[29] H. Li, J. Xu, et al. Semantic matching in search. *FnTIR*, 7(5):343–469, 2014.
[30] J. Li et al. Deep reinforcement learning for dialogue generation. In *EMNLP*, 2016.
[31] Z. Lu and H. Li. A deep architecture for matching short texts. In *NIPS*, pages 1367–1375, 2013.
[32] J. McAuley et al. Image-based recommendations on styles and substitutes. In *SIGIR*, 2015.
[33] H. Mei, M. Bansal, and M. R. Walter. What to talk about and how? Selective generation using LSTMs with coarse-to-fine alignment. *arXiv preprint arXiv:1509.00838*, 2015.

[34] T. Mikolov et al. Recurrent neural network based language model. In *Interspeech*, 2010.
[35] T. Mikolov et al. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
[36] B. Mitra, F. Diaz, and N. Craswell. Learning to match using local and distributed representations of text for web search. In *WWW '17*, 2017.
[37] B. Mitra et al. A dual embedding space model for document ranking. *arXiv preprint arXiv:1602.01137*, 2016.
[38] A. Oord, S. Dieleman, and B. Schrauwen. Deep content-based music recommendation. In *NIPS*, 2013.
[39] M. Quadrana et al. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *RecSys*, 2017.
[40] S. E. Robertson et al. Okapi at TREC-3. *Nist Special Publication Sp*, 109: 109, 1995.
[41] R. Salakhutdinov and G. Hinton. Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978, 2009.
[42] I. V. Serban et al. Generating factoid questions with recurrent neural networks: The 30m factoid question-answer corpus. *arXiv preprint arXiv:1603.06807*, 2016.
[43] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.
[44] M. Szummer and E. Yilmaz. Semi-supervised learning to rank with preference regularization. In *CIKM*, pages 269–278. ACM, 2011.
[45] C. Van Gysel, M. de Rijke, and E. Kanoulas. Learning latent vector spaces for product search. In *CIKM*, pages 165–174, 2016.
[46] C. Van Gysel, M. de Rijke, and M. Worring. Unsupervised, efficient and semantic expertise retrieval. In *WWW*, pages 1069–1079, 2016.
[47] C. Van Gysel, M. de Rijke, and E. Kanoulas. Neural vector spaces for unsupervised information retrieval. *arXiv preprint arXiv:1708.02702*, 2017.
[48] C. Van Gysel et al. Reply with: Proactive recommendation of email attachments. In *CIKM*, 2017.
[49] F. Vasile et al. Meta-Prod2Vec – product embeddings using side-information for recommendations. In *RecSys*, 2016.
[50] O. Vinyals and Q. Le. A neural conversational model. In *ICML*, 2015.
[51] I. Vulić and M.-F. Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proc. SIGIR*, pages 363–372. ACM, 2015.
[52] H. Wang et al. Collaborative deep learning for recommender systems. In *KDD*, 2015.
[53] Z. Wang, J. Zhang, J. Feng, and Z. Chen. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119, 2014.
[54] J. Weston et al. Towards AI-complete question answering: A set of prerequisite toy tasks. In *ICLR*, 2016.
[55] H. Zamani and W. B. Croft. Embedding-based query language models. In *ICTIR*, pages 147–156. ACM, 2016.

[56] H. Zamani and W. B. Croft. Estimating embedding vectors for queries. In *ICTIR*, pages 123–132. ACM, 2016.
[57] Y. Zhao, L. Zhiyuan, and M. Sun. Representation learning for measuring entity relatedness with rich information. In *IJCAI*, pages 1412–1418, 2015.
[58] G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *ADCS*. ACM, 2015.