Codeless Deep Learning with KNIME

Build, train, and execute various deep neural network architectures using KNIME Analytics Platform

Rosaria Silipo | Kathrin Melcher

KNIME Analytics Platform: a tool to support deep learning

If you've been working in the field of data science for a few years, you will probably have noticed the recent explosion of scholarly and practitioners' articles about successful solutions based on deep learning techniques.

The big breakthrough happened in 2012 when the deep learning-based AlexNet network won the ImageNet challenge by an unprecedented margin. That victory led to a surge in the use of deep learning networks, which have since been successfully applied in an increasing number of domains.

So, what are we referring to exactly when we talk about 'deep learning'? The term covers a subset of machine learning algorithms, mostly stemming from neural networks. Thus deep learning represents the evolution of traditional neural networks. To classic feedforward, fully connected, backpropagation-trained, multilayer perceptrons, 'deeper' architectures have been added. Deeper here means more hidden layers and a number of additional neural paradigms, including Recurrent Neural Networks (RNNs), Long-Short Term Memories (LSTMs), Convolutional Neural Networks (CNNs), Generative Adversarial Networks (GANs), and others besides.

The recent success of these new types of neural network can be attributed to a number of factors. First, the increased computational power in modern machines has favored the introduction and development of new paradigms and more complex neural architectures. Being able to train a complex neural network in minutes leaves time for more experimentation than when training the same network takes hours or days. A second reason is their flexibility. Neural networks are universal function approximators, which means they can approximate almost anything provided that their architecture is sufficiently complex.

Understanding of the algorithms, experience of the most effective paradigms and architectures, and domain knowledge, are all necessary ingredients for the success of a data science project. However, there are other more contingent factors – such as ease of learning, speed of prototyping, options for debugging and testing to ensure the correctness of the solution, flexibility to experiment, availability of help from external experts, and finally automation and security capabilities – that also influence the success of a project. It is in relation to these additional factors that choice of software platform can be especially important, and one analytics platform that is finding favor with a growing number of data scientists is **KNIME**.

KNIME

If you were intrigued by neural networks and deep learning architectures but were put off by the coding effort that seemed likely to be involved, then it is will come as a relief to discover a tool like KNIME, and in particular the KNIME Analytics Platform in conjunction with KNIME Server.

KNIME Analytics Platform provides support for the machine learning and deep learning algorithms and data operations needed for a modern data science project, while KNIME Server provides the IT infrastructure for an easy and secure deployment and model monitoring over time. Let's first look at KNIME Analytics Platform and its capabilities.

KNIME Analytics Platform

KNIME Analytics Platform is open-source software. It is free to download from the KNIME website (https://www.knime.com/downloads) and is free to use, and it covers all the main data wrangling and machine learning techniques.

A key feature of KNIME Analytics Platform is **visual programming**, which makes the tool very easy to use and ideal for quick prototyping. In visual programming, a Graphical User Interface (GUI) guides you through all necessary steps to build a pipeline (workflow) of dedicated blocks (nodes). Each node implements a given task; each workflow of nodes takes your data from the beginning to the end of the designed journey. A workflow substitutes a script; a node substitutes one or more script lines.

However, ease of use would be of limited benefit if there were not also extensive coverage of the commonly and less commonly used data wrangling techniques, machine learning algorithms, data types and formats, or if there was not integration with the most commonly used database software, data sources, reporting tools, external scripts and programming languages. For this reason, KNIME Analytics Platform has been designed to be open to different data formats, data types, data sources, and data platforms, as well as to external tools such as Python and R.

To start with, consider *machine learning algorithms*. Here KNIME Analytics Platform has all the major bases covered: from decision trees to random forest and gradient boosted trees, from recommendation engines to a number of clustering techniques, from naïve Bayes to linear and logistic regression, from neural networks to deep learning. Most of these algorithms are native to KNIME Analytics Platform, while others are integrated from other open source tools such as Python and R.

To train different deep learning architectures, such as recurrent neural networks, autoencoders, or convolutional neural networks, KNIME Analytics Platform has integrated the **Keras** deep learning library through the extension **KNIME Deep Learning - Keras**

integration (https://www.knime.com/deeplearning/keras). Through this extension, it is possible to drag and drop nodes to define complex neural architectures and to train the final network without necessarily writing code.

Defining the network is one step. Bringing the data into the right shape so the network can be trained is another crucial step. For this, a very large number of nodes is available to implement a myriad of *data wrangling techniques*. By combining nodes dedicated to small tasks, you can implement very complex data transformation operations.

KNIME Analytics Platform also connects to most of the required *data sources*, from databases to cloud repositories, from big data platforms to individual files.
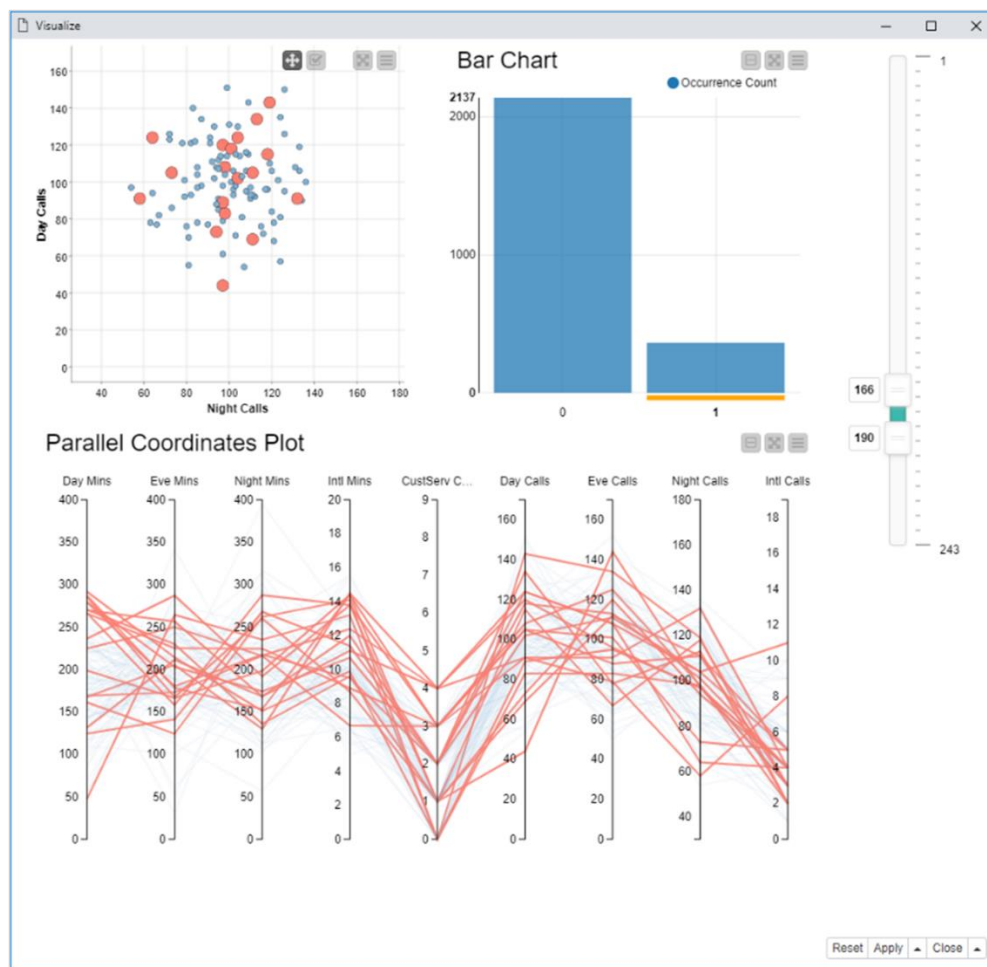

Figure 1 The composite view from a component containing a scatter plot, a bar chart, and a parallel coordinate plot. The three plots visualize the same data and are connected in a way that, selecting data in the bar chart, selects and optionally visualizes only the selected data in the other two charts.

This is a sample chapter of *Codeless Deep Learning with KNIME* by Kathrin Melcher and Rosaria Silipo and published by Packt Publishing. The complete book is available for purchase at https://packt.live/3ewrVDJ

What if all that is not enough? What if I need a specific procedure for a specific domain? What if I need a specific network manipulation function from Python? Where KNIME Analytics Platform and its extensions cannot reach, there are the *integrations with other scripting and programming languages,* such as Python, R, Java, and Javascript, to name but a few. In addition, KNIME Analytics Platform offers seamless integration with the BIRT Reporting tool. Integrations with other *reporting platforms* like Tableau, QlikView, PowerBI, and Spotfire are also available.

Several Javascript-based nodes are dedicated to creating data visualization plots and charts: from a simple scatter plot to a more complex sunburst chart, from a simple histogram to a parallel coordinate plot, and more. These nodes seem simple but are potentially quite powerful. If you combine them within a *component*, interactive selection of data points across multiple charts is automatically enabled. The component thus inherits and combines all views from the contained nodes and connects them together in a way that, if points are selected and visualized in one chart, they can also be selected and visualized in the other charts of the component's composite view. Figure 1 shows an example of a composite view.

In short, KNIME Analytics Platform has everything you need to create a powerful data science solution. However, KNIME Server offers a few additional features to ease your job when moving the solution into production.

KNIME Server for the enterprise

The last step in any data science cycle is deployment into production and, in the case of an enterprise, straightforward, secure deployment.

This process of moving the application into the real-world is called *moving into production*. The process of including the trained model into this final application is called *deployment*. Both phases are deeply connected and can be quite problematic since all errors in the application design show up at this stage.

It is possible to move an application into production using just KNIME Analytics Platform. If you, as a lone data scientist or a data science student, do not regularly deploy applications and models, KNIME Analytics Platform is probably enough for your needs. However, in the context of an enterprise environment, where scheduling, versioning, access rights, disaster recovery, web applications and REST services, and all other typical functions of a production server are needed, then just using KNIME Analytics Platform for production can be cumbersome.

In this case**, KNIME Server**, which comes with an annual license fee, can make your life easier. First of all, it is going to fit better the governance model of the enterprise's IT environment. Then, it offers a collaboration protected environment for your group and the whole data science lab. But its main advantage consists in making model deployment and the move into production easier and safer, for example by using the *integrated deployment*

feature and the *one-click deployment* into production. End users would then run the application from a KNIME Analytics Platform client or even better from a web browser.

Remember those composite views offering interactive interconnected views of selected points we mentioned? Those become fully formed web pages when the application is executed on a web browser via the *KNIME Server's WebPortal.*

Using the components as touchpoints within the workflow, we get a *guided analytics* (https://www.knime.com/blog/principles-of-guided-analytics) application within the web browser. Guided analytics inserts touchpoints to be consumed by the end user from a web browser within the flow of the application. The end user can take advantage of these touchpoints to insert knowledge or preferences and to steer the analysis in the desired direction.

KNIME Deep Learning Keras Integration

KNIME Analytics Platform's extensions and integrations greatly enrich the software's core functionalities, enabling it to tap into the most advanced algorithms for artificial intelligence.

TensorFlow is an open source library provided by Google which includes a variety of deep learning paradigms. TensorFlow functions can run on single devices as well as on multiple CPUs and multiple GPUs. This parallel calculation feature is the key to speeding up the computationally intensive training required for deep learning networks.

However, using the TensorFlow library within Python can prove quite complicated, even for an expert Python programmer or deep learning professional. Thus, simplified interfaces have been developed on top of TensorFlow, exposing a subset of its functions and parameters. The most successful of such TensorFlow-based libraries is Keras. Even Keras, however, still requires some programming skills. But as we briefly mentioned above, there is a KNIME extension to help us here: the **KNIME Deep Learning – Keras Integration**. This offers a codeless GUI-based integration of the Keras library, while using TensorFlow as backend. This means that a number of Keras functions have been wrapped into KNIME nodes, most of them providing a visual dialog window to set the required parameters. In this way, you can read, write, create, train, and execute deep learning networks without writing code.

Another important deep learning integration is the **KNIME Deep Learning – TensorFlow Integration.** This extension allows you on the one hand to convert **Keras** models into **TensorFlow** models and on the other to read, execute, and write TensorFlow models. Powerful stuff!

Useful links and material

A rich ecosystem of resources exists to help you make the most of KNIME in your data science and deep learning activities.

The KNIME Hub (https://hub.knime.com/) is a very useful public repository of applications, extensions, examples, and tutorials provided by the KNIME community. Here, you can share your workflows and download workflows by other KNIME users. Just type in the keywords and you will get a list of related workflows, components, extensions, and more. For example, just type in 'read file' and you will get a list of example workflows illustrating how to read CSV files, .table files, Excel files, etc … (**Error! Reference source not found.**).
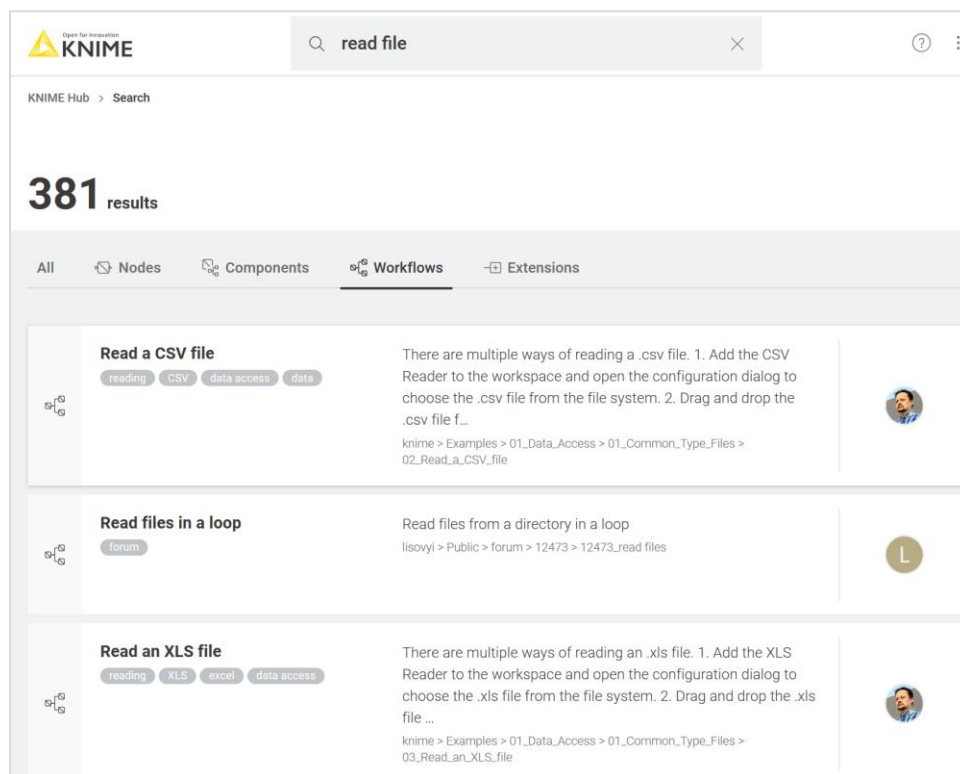
Figure 2 Resulting list of workflows from search "read file" on KNIME Hub

Once you isolate the workflow you are interested in, click on it to open its page, and then download it or open it on your own KNIME Analytics Platform to customize it to your own needs.

On the other side, to share your work on the KNIME Hub, just copy your workflows from your local workspace into the *My-KNIME-Hub/Public* folder in the KNIME Explorer panel within the KNIME workbench. It will be automatically available to all members of the KNIME community.

This is a sample chapter of *Codeless Deep Learning with KNIME* by Kathrin Melcher and Rosaria Silipo and published by Packt Publishing. The complete book is available for purchase at https://packt.live/3ewrVDJ

The KNIME community is also very active, with tips and tricks shared on the **KNIME Forum** (https://forum.knime.com/). Here you can also ask questions or search for answers.

Finally, contributions by the community are available as posts on the **KNIME Blog** (https://www.knime.com/blog), as books in the **KNIME Press** (https://www.knime.com/knimepress), and as videos on the **KNIME TV** (https://www.youtube.com/user/KNIMETV) channel on YouTube.