### **1. Online Courses**

* **DeepLearning.AI’s Deep Learning Specialization (Coursera)**: This series of courses by Andrew Ng provides a comprehensive foundation in deep learning with practical examples, including different neural network architectures.
* **Udacity’s Deep Learning Nanodegree**: This program offers practical projects and covers various aspects of neural networks including convolutional and recurrent networks.

### **2. Books**

* **“Deep Learning” by Ian Goodfellow, Yoshua Bengio, and Aaron Courville**: This is considered a seminal book in the field, offering deep insights into the theory of various neural network architectures.
* **“Dive into Deep Learning” by Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola**: This book is practical and includes executable code, aligning theory with practice. It’s also freely available online.

### **3. Online Tutorials and Blogs**

* **Distill.pub**: Offers clear, visually intuitive explanations of concepts in deep learning, including transformers.
* **Jay Alammar’s Blog**: Provides visual and intuitive understanding of transformers and other NLP models.

### **4. Interactive Platforms**

* **Kaggle**: Engage in competitions or explore kernels/notebooks shared by others. It’s a great way to see practical applications of neural networks.
* **Google Colab**: Allows you to run deep learning models free on the cloud, including those with GPU resources.

### **5. Research Papers**

To get familiar with the latest in neural network architectures like transformers, reading papers can be invaluable:

* **“Attention Is All You Need” by Vaswani et al.**— the foundational paper for transformers.
* Platforms like **arXiv** and **Google Scholar** can be useful to find papers.

### **6. GitHub Repositories**

* Many researchers and developers share their code implementations of neural network architectures on GitHub. Searching for a particular model type will usually yield open-source codebases you can study and learn from.

### **7. Community and Forums**

* **r/MachineLearning** on Reddit or Stack Overflow can be great places to ask specific questions and engage with the community.

As you explore these resources, try to apply what you learn in small projects. This could be as simple as using a dataset from Kaggle and applying different neural network architectures to see how they perform. This kind of hands-on learning is invaluable.