Analytics Vidhya

Hands-On: Build a Contextual Retrieval based RAG System

Instructor

Dipanjan Sarkar

Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

Published Author

Data Source

Text Article JSON


Deep Residual Learning for Image Recognition

Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun
Microsoft Research

Attention Is All You Need

TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy[∗,†], Lucas Beyer[∗], Alexander Kolesnikov[∗], Dirk Weissenborn[∗], Xiaohua Zhai[∗], Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby[∗,†]
[∗]equal technical contribution, [†]equal advising
Google Research, Brain Team
{adosovitskiy, neilhoulsby}@google.com

ABSTRACT

While the Transformer architecture has become the de-facto standard for natural language processing tasks, its applications to computer vision remain limited. In vision, attention is either applied in conjunction with convolutional networks, or used to replace certain components of convolutional networks while keeping their overall structure in place. We show that this reliance on CNNs is not necessary and a pure transformer applied directly to sequences of image patches can perform very well on image classification tasks. When pre-trained on large amounts of data and transferred to multiple mid-sized or small image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer (ViT) attains excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources to train.[1]

1 INTRODUCTION

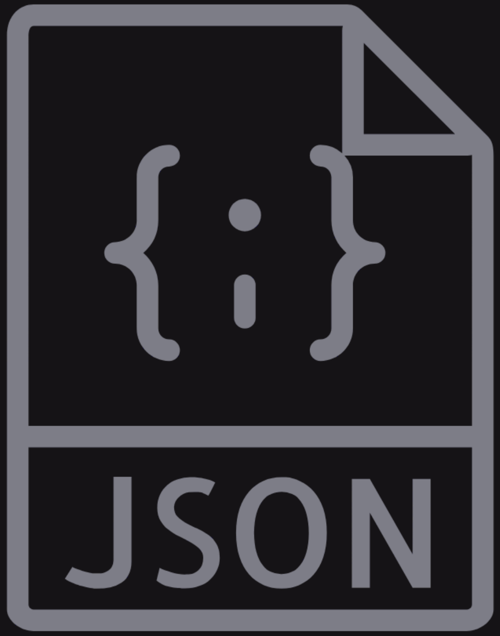
Self-attention-based architectures, in particular Transformers (Vaswani et al., 2017), have become the model of choice in natural language processing (NLP). The dominant approach is to pre-train on a large text corpus and then fine-tune on a smaller task-specific dataset (Devlin et al., 2019). Thanks to Transformers' computational efficiency and scalability, it has become possible to train models of unprecedented size, with over 100B parameters (Brown et al., 2020; Lepikhin et al., 2020). With the models and datasets growing, there is still no sign of saturating performance.

In computer vision, however, convolutional architectures remain dominant (LeCun et al., 1989; Krizhevsky et al., 2012; He et al., 2016). Inspired by NLP successes, multiple works try combining CNN-like architectures with self-attention (Wang et al., 2018; Carion et al., 2020), some replacing the convolutions entirely (Ramachandran et al., 2019; Wang et al., 2020a). The latter models, while theoretically efficient, have not yet been scaled effectively on modern hardware accelerators due to the use of specialized attention patterns. Therefore, in large-scale image recognition, classic ResNet-like architectures are still state of the art (Mahajan et al., 2018; Xie et al., 2020; Kolesnikov et al., 2020).

Inspired by the Transformer scaling successes in NLP, we experiment with applying a standard Transformer directly to images, with the fewest possible modifications. To do so, we split an image

Research Paper PDFs

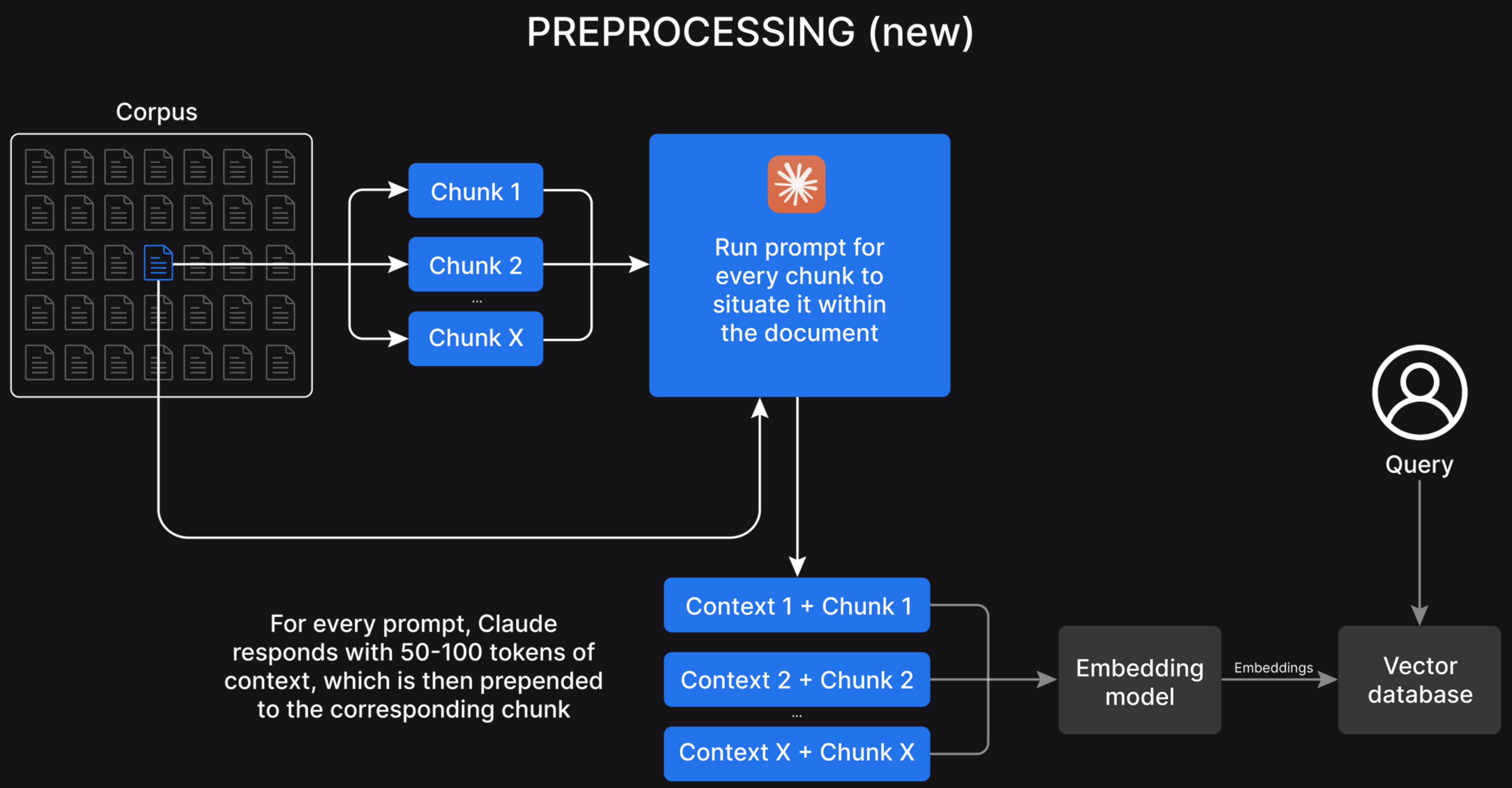
Data Loader



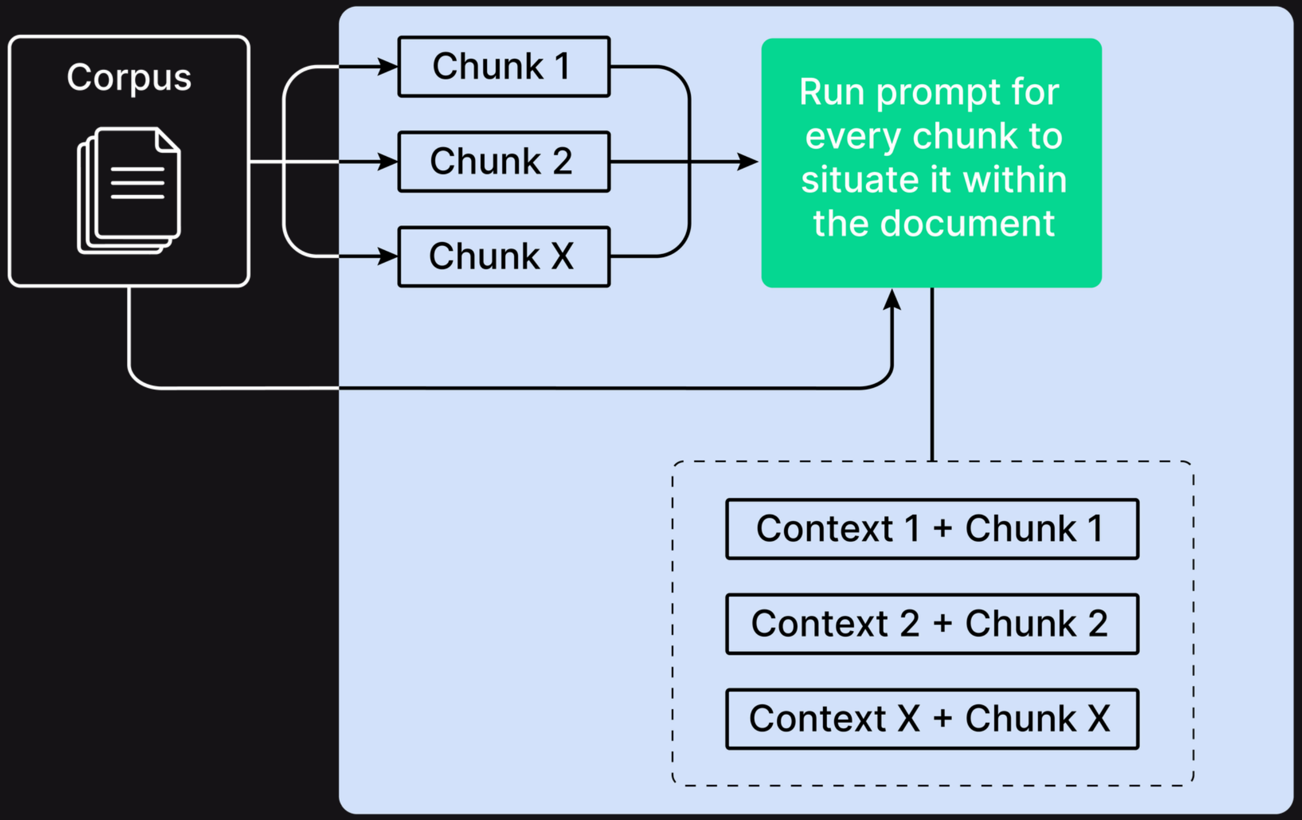
JSON Loader

PDF Loader

Contextual Retrieval Workflow



Chunking Strategies



Recursive Character + Contextual Chunking

Analytics Vidhya

Contextual Chunking

- Prepend chunk-specific explanatory context to each chunk before creating the vector DB embeddings.
- Helps with having keywords or phrases in each chunk based on its relevance to the overall document.
- Improves retrieval performance quite a bit which also helps with the overall RAG generation results because of better context.
- The contextual chunking prompt can be built in various ways depending on your use-case.

```
def generate_chunk_context(document, chunk):

    chunk_process_prompt = """You are an AI assistant specializing in research paper analysis.
                        Your task is to provide brief, relevant context for a chunk of text
                        based on the following research paper.

                        Here is the research paper:
                        <paper>
                        {paper}
                        </paper>

                        Here is the chunk we want to situate within the whole document:
                        <chunk>
                        {chunk}
                        </chunk>

                        Provide a concise context (3-4 sentences max) for this chunk,
                        considering the following guidelines:

                        - Give a short succinct context to situate this chunk within the overall
                        document for the purposes of improving search retrieval of the chunk.
                        - Answer only with the succinct context and nothing else.
                        - Context should be mentioned like 'Focuses on ....'
                        do not mention 'this chunk or section focuses on...'

                        Context:
                    """

    prompt_template = ChatPromptTemplate.from_template(chunk_process_prompt)

    agentic_chunk_chain = (prompt_template
                                |
                               llm
                                |
                            StrOutputParser())

    context = agentic_chunk_chain.invoke({'paper': document, 'chunk': chunk})

    return context
```

Thank You

Analytics Vidhya