Derin Öğrenme
Deep Learning

Hazırlayan: M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

İçerik

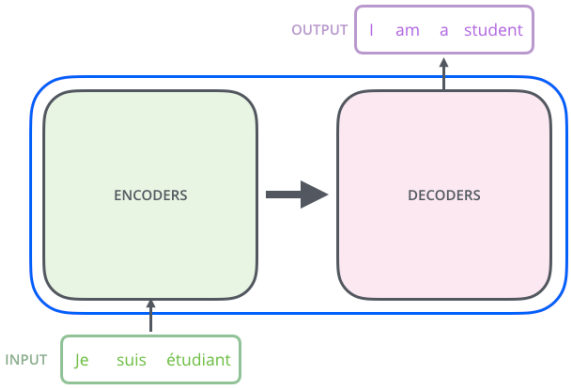
- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

Transformer'ların yapısı

- Dikkat mekanizması, sinirsel makine çevirisi uygulamalarının performansını artırır.
- **Transformer,** çeviri modellerinin eğitim hızını artırmak için **dikkat (attention)** mekanizması kullanan modeldir.
- Transformer, ilk defa "Attention is All You Need" adlı makalede önerilmiştir.
- Transformer makine çevirisi uygulamasında bir dildeki cümleyi alıp başka bir dildeki çevirisini üretir.

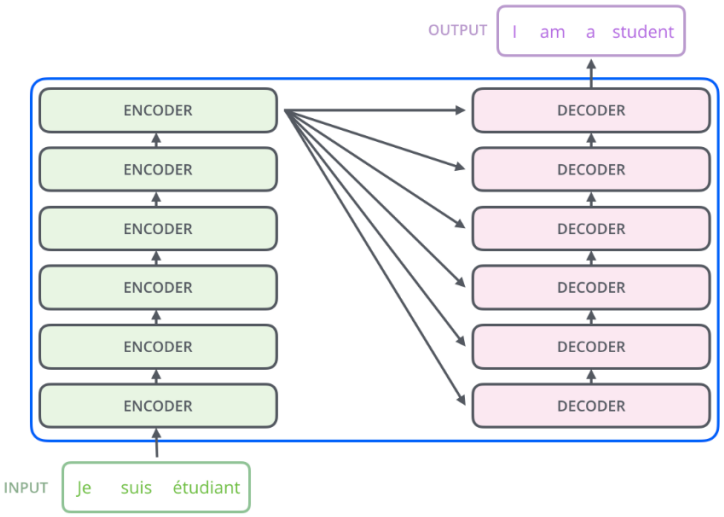
Transformer'ların yapısı

- Bir encoder, bir decoder ve aralarındaki bağlantıdan oluşur.



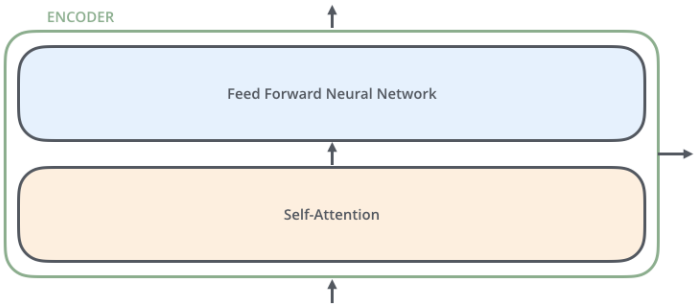
Transformer'ların yapısı

- **Encoder**, bir grup kodlayıcıdan oluşur. **Decoder** bileşeni de aynı sayıda kod çözücüden oluşan bir yığındır.



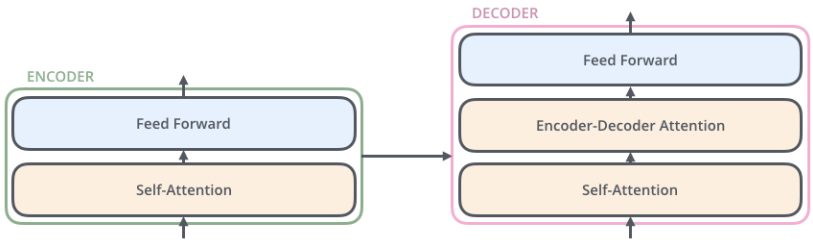
Transformer'ların yapısı

- Kodlayıcıların hepsi yapı olarak özdeştir, ancak ağırlıkları paylaşmazlar.
- Her kodlayıcı iki alt katmandan oluşur.



Transformer'ların yapısı

- Kodlayıcının girdileri önce bir self-attention katmanından geçer.
- Self-attention katmanı, kodlayıcının belirli bir kelimeyi kodlarken giriş cümlesindeki diğer kelimelere bakmasına yardımcı olur.
- Self-attention katmanının çıktıları ileri beslemeli sinir ağına girilir.
- Aynı ileri beslemeli ağ, her konumda bağımsız olarak uygulanır.



- Decoder'da bu iki katman da bulunur, ancak aralarında kod çözücünün giriş cümlesinin ilgili kısımlarına odaklanmasına yardımcı olan bir attention katmanı vardır.

İçerik

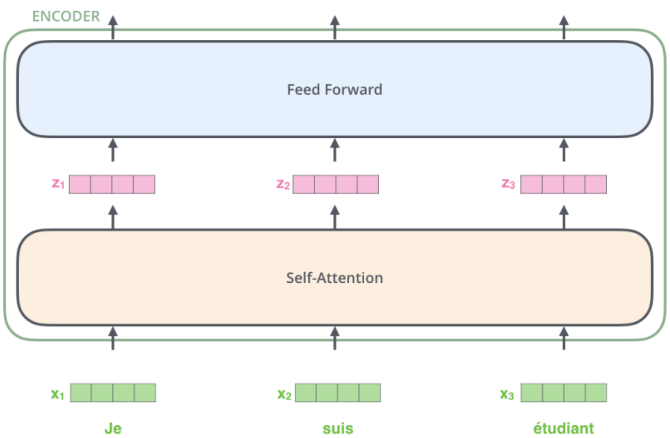
- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

Transformer'ların çalışması

- NLP uygulamalarında olduğu gibi, her bir girdi kelimesi bir embedding (gömme) algoritması kullanılarak bir vektöre dönüştürülür.
- Her kelime belirli uzunlukta vektöre dönüştürülür.
- Gömme, sadece en alttaki kodlayıcıda yapılır. Tüm kodlayıcılar, her birinin boyutu 512 olan bir vektör listesi alır.
- En alttaki kodlayıcı kelime gömme vektörleri alırken, diğer kodlayıcılar doğrudan altındaki kodlayıcının çıktısını alır.
- Vektörün boyutu bir hiperparametredir.

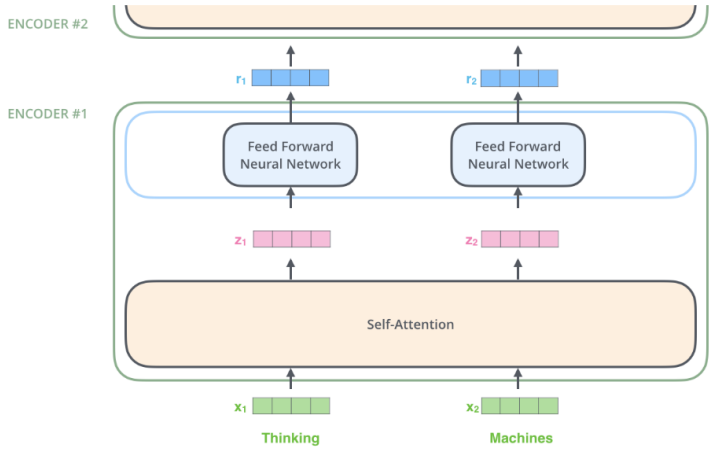
Transformer'ların çalışması

- Kelimeler vektöre dönüştürüldükten sonra, her biri kodlayıcının iki katmanından da geçer.
- Her konumdaki kelime, kodlayıcıda kendi yolundan geçer.
- Öz dikkat katmanında yollar arasında bağımlılıklar vardır.



Transformer'ların çalışması

- Bir kodlayıcı, girdi olarak bir vektör listesi alır.
- Bu listeyi, vektörleri bir 'öz dikkat' katmanına, ardından bir ileri beslemeli sinir ağına geçirerek işler.
- Daha sonra çıktıyı bir sonraki kodlayıcıya yukarı doğru gönderir.



İçerik

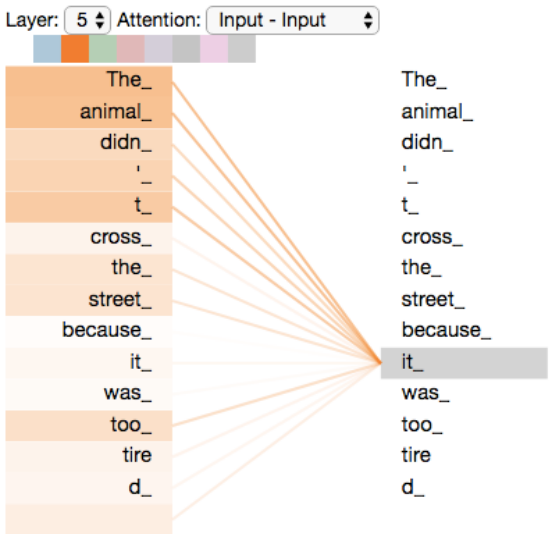
- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

Self-attention

- Aşağıdaki cümlede "it" kelimesi caddeyi mi yoksa hayvanı mı ifade ediyor?
  *"The animal didn't cross the street because **it** was too tired"*
- Bu soru bir insan için çok kolaydır, ancak makine için çok zordur.
- Daha sonra çıktıyı bir sonraki kodlayıcıya yukarı doğru gönderir.
- Model "it" kelimesini işlerken, self-attention mekanizması "it" kelimesinin "hayvan" ile ilişkilendirmesini sağlar.
- Self-attention mekanizması "it" kelimesi için daha iyi bir kodlamayı sağlayacak ipuçları için giriş dizisindeki diğer kelimelere bakmasını sağlar.

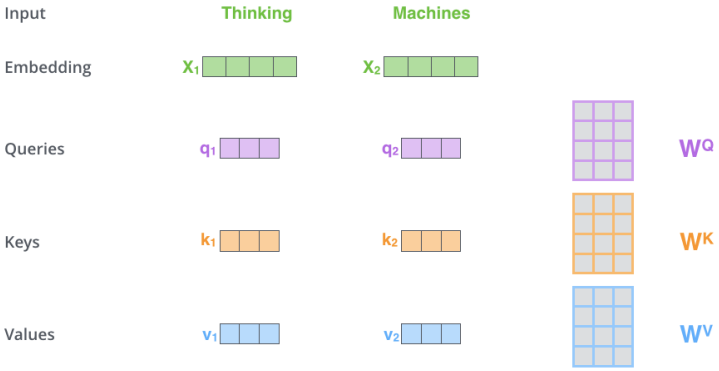
Self-attention

- Transformer, self-attention mekanizmasını diğer ilgili kelimelerin "anlayışını" şu anda işlenen kelimeye yerleştirmek için kullanır.



Self-attention

- İlk adımda, kodlayıcının giriş vektörlerinin her birinden üç vektör oluşturulur.
- Her kelime için bir Sorgu (Query) vektörü, bir Anahtar (Key) vektörü ve bir Değer (Value) vektörü oluşturulur.
- Bu vektörler, eğitim sürecinde elde edilen üç matris ile gömme vektörünün çarpılmasıyla elde edilir.

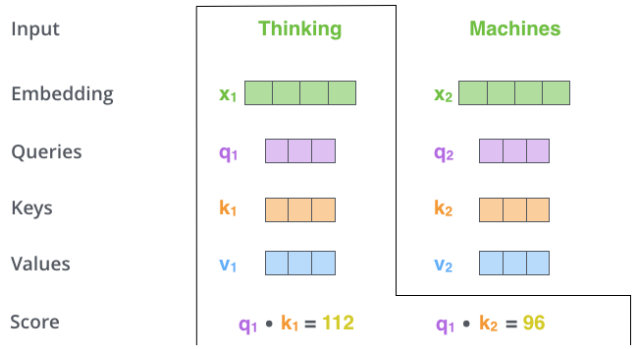


Self-attention

- "Query", "Key" ve "Value", attention hesaplamasında kullanılır.
- Self-attention hesaplamasının ikinci adımda her kelime için bir puan hesaplanır.
- Örneğin "Thinking" için self-attention hesaplayalım.
- Giriş cümlesinin her kelimesi bu kelimeye göre puanlanır.
- Puan, belirli bir konumdaki bir kelimeyi kodlarken giriş cümlesinin diğer bölümlerine ne kadar odaklanacağımızı belirlemek için kullanılır.

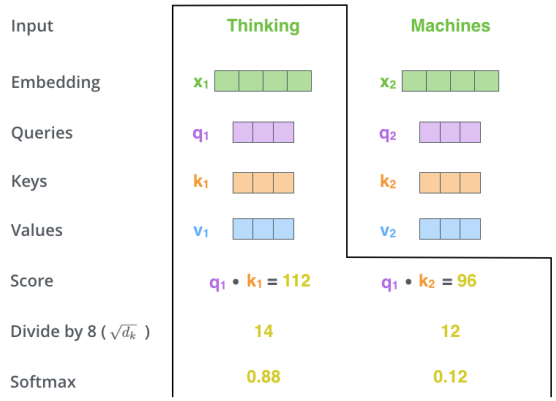
Self-attention

- Puan, query vektörünün puanladığımız ilgili kelimenin key vektörüyle dot product alınarak hesaplanır.



Self-attention

- Üçüncü ve dördüncü adımlarda, puanlar anahtar vektör boyutunun (64) karekköüne (8) bölünür.
- Ardından sonucu bir softmax işleminden geçirilir.
- Softmax, puanları tümü pozitif olacak ve toplamları 1 olacak şekilde normalleştirir.

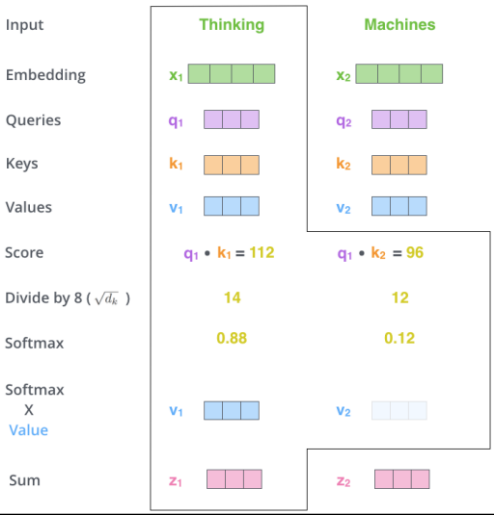


Self-attention

- Softmax puanı, her kelimenin bu konumda ne kadar ifade edileceğini belirler.
- Bu konumdaki kelime en yüksek softmax puanına sahip olacaktır.
- Beşinci adımda, her değer (value) vektörü softmax puanıyla çarpılır.
- Buradaki mantık, odaklanmak istediğimiz kelime(ler)in değerlerini korumak ve alakasız kelimeleri (örneğin 0,001 gibi küçük sayılarla çarparak) elimine etmektir.

Self-attention

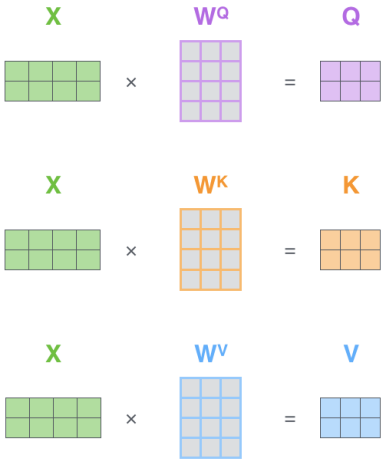
- Altıncı adımda, ağırlıklı değer vektörleri toplanır.
- Elde edilen vektör, bu konumdaki (ilk kelime için) self-attention katmanının çıktısıdır ve feed-forward network'e gönderilir.



Self-attention

Matris hesabı

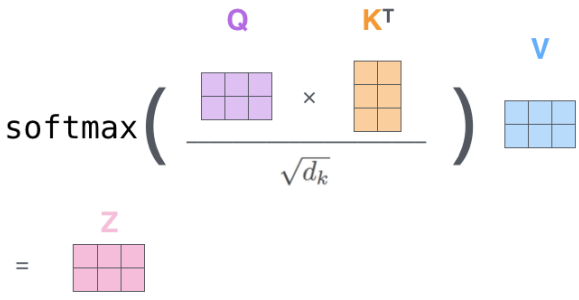
- İlk adımda, Query, Key ve Value matrisleri hesaplanır.
- Ardından, gömme vektörleri bir X matrisinde birleştirilerek eğitilen ağırlık matrisleriyle (W^Q, W^K, W^V) çarpılır.



Self-attention

Matris hesabı

- Son olarak, softmax katmanı çıkışları hesaplanır.

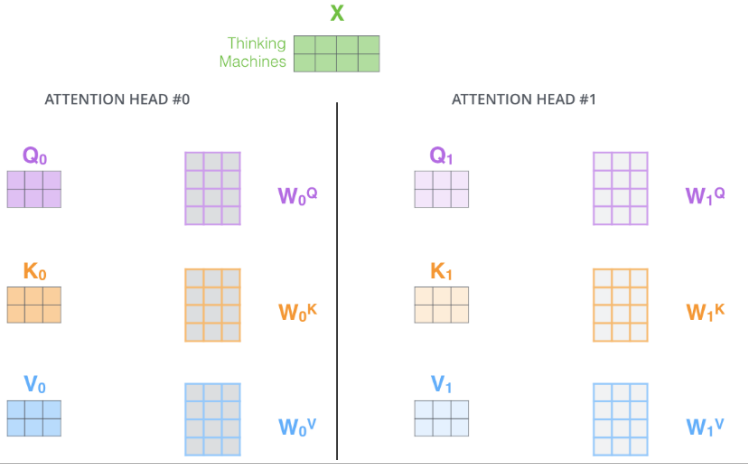


İçerik

- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

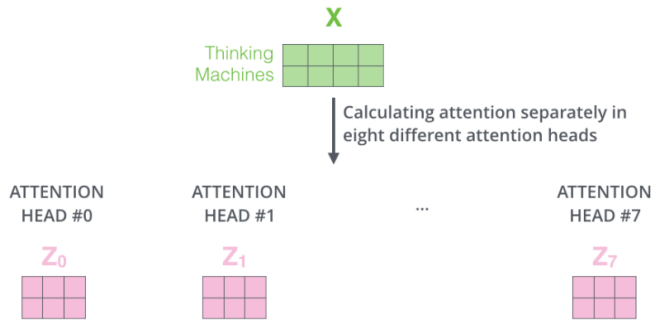
Multi-head self-attention

- Multi-head attention mekanizması eklenerek self-attention katmanı geliştirmiştir.
- Attention katmanına birden fazla temsil alt uzayı sağlar.
- Transformer sekiz multi-head attention kullanır.



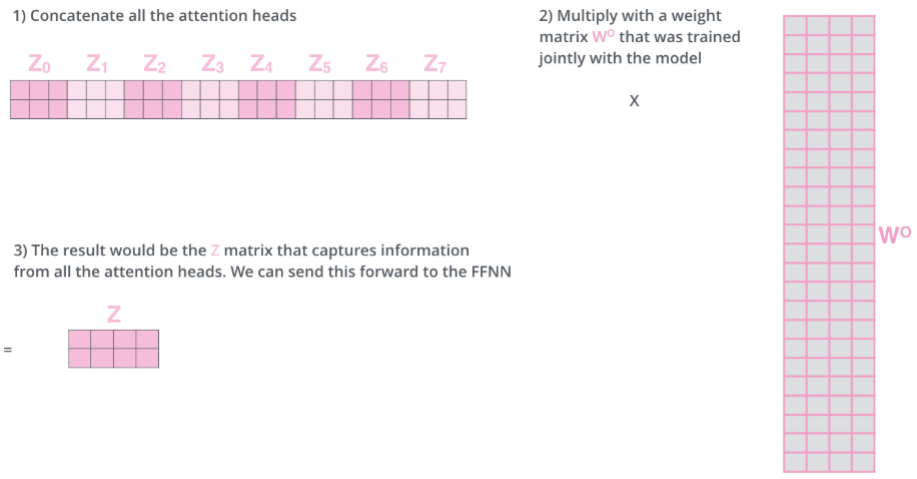
Multi-head self-attention

- Self-attention hesaplamasını, farklı ağırlık matrisleriyle sekiz kez yaparsak, sekiz farklı Z matrisi elde ederiz.



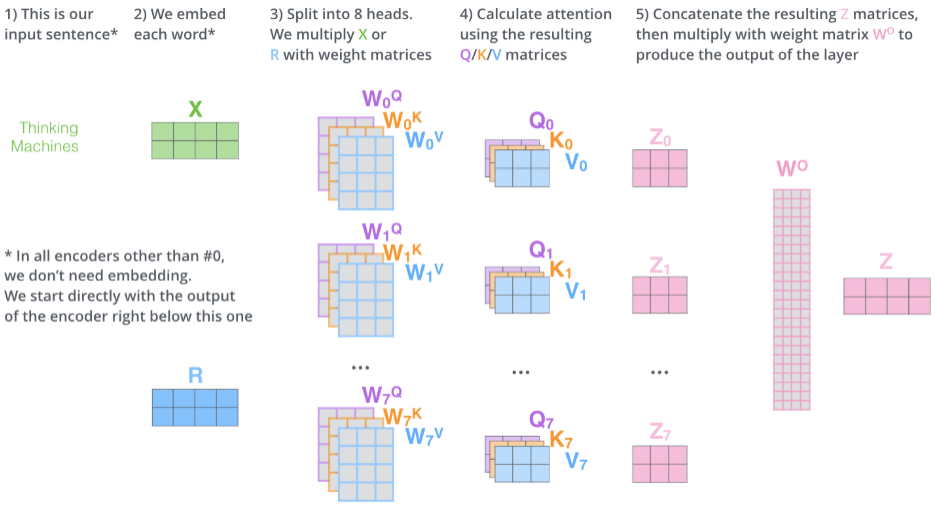
Multi-head self-attention

- Sekiz farklı Z matrisi birleştirilerek W^0 ağırlık matrisiyle çarpılır.



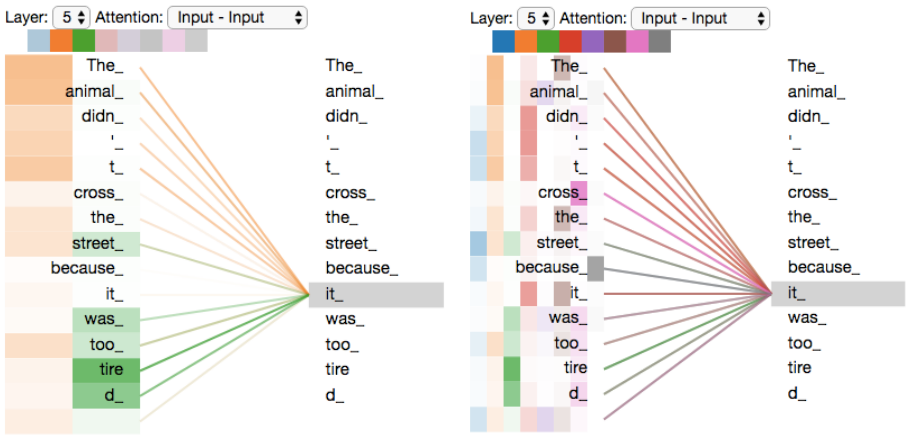
Multi-head self-attention

- İlk encoder gömme değerini, diğerleri öncekinin çıktısını alır.



Multi-head self-attention

- Örnek cümlede "it" kelimesi kodlanırken farklı attention head'leri farklı değerlere sahiptir.



İçerik

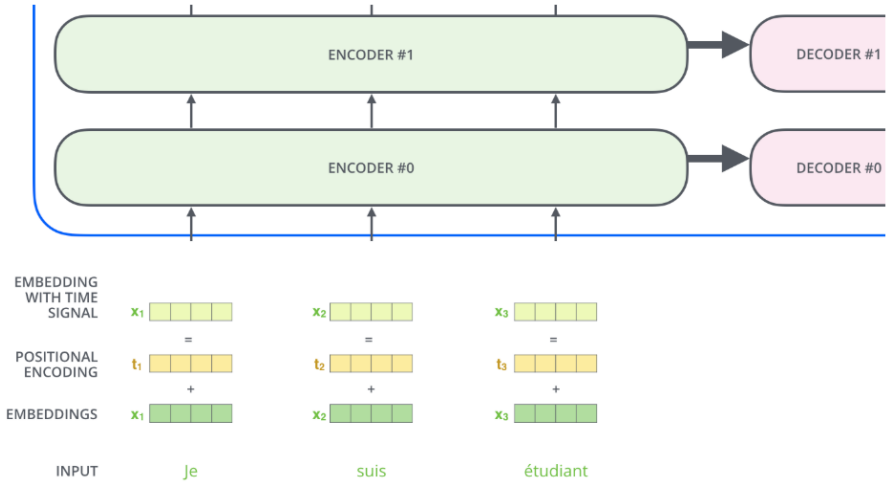
- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

Positional encoding kullanarak sıralama

- Giriş dizisindeki kelimelerin sırası hesaba katılmalıdır.
- Bunun için, transformer her giriş gömme vektörüne başka bir vektör ekler.
- Bu yeni vektör, modelin öğrendiği belirli bir deseni tutar.
- Böylece model, her kelimenin konumunu veya dizideki farklı kelimeler arasındaki mesafeyi belirler.
- Bu yeni vektörler gömme değerlerine eklenerek Q/K/V vektörlerine yansıtılır.
- Dot-product attention mekanizması sırasında gömme vektörleri arasında anlamlı mesafeler elde edilir.

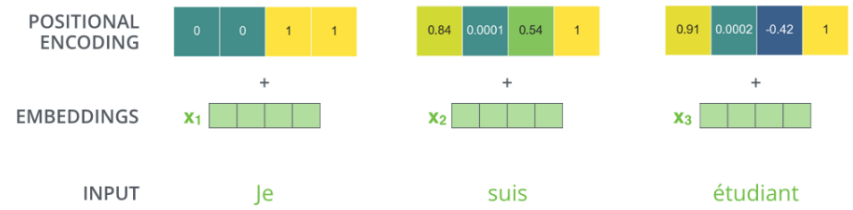
Positional encoding kullanarak sıralama

- Elde edilen vektörler ilk encoder'ın girdileri olur.



Positional encoding kullanarak sıralama

- Gömme işleminin 4 boyutlu olduğunu varsayarsak, positional encoding'ler aşağıdaki gibi olur.

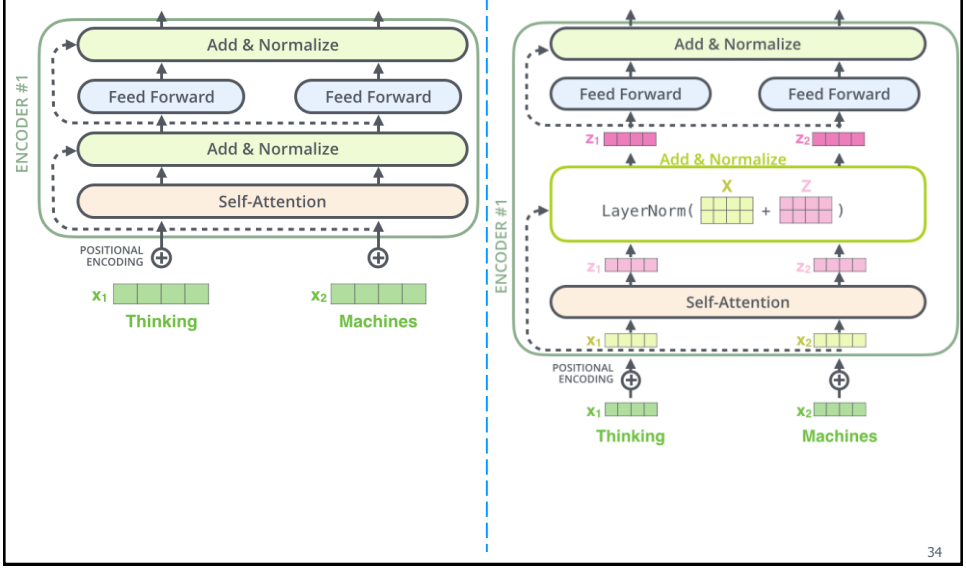


İçerik

- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

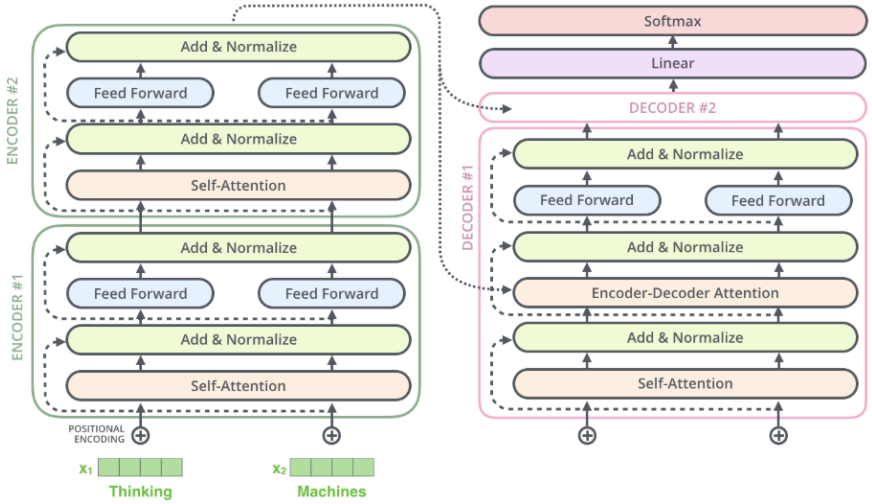
Residuals

- Her attention mekanizmasından sonra bir normalizasyon katmanı gelir.



Residuals

- Bu durum kod çözücünün alt katmanları için de geçerlidir.
- İki katmanlı kodlayıcı ve kod çözücüden oluşan bir Transformer aşağıdaki gibidir.

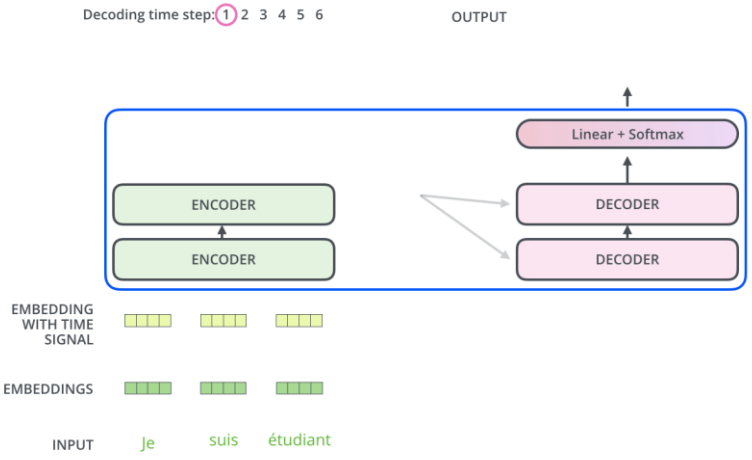


İçerik

- Transformer'ların yapısı
- Transformer'ların çalışması
- Self-attention
- Multi-head self-attention
- Positional encoding kullanarak sıralama
- Residuals
- Decoder

Decoder

- İlk encoder giriş dizisini alarak başlar.
- En üstteki encoder K ve V attention vektörlerini elde eder.
- K ve V vektörleri decoder'ların girişlerini oluşturur.



Decoder

- İşlemler, kod çözücünün çıktısını tamamladığını gösteren özel bir sembole (end of sentence) ulaşana kadar tekrarlanır.
- Her adımın çıktısı, bir sonraki adımdaki alt kod çözücüye giriş olarak verilir.
- Kod çözücüler, kodlayıcıların yaptığı gibi kod çözme sonuçlarını yukarı doğru iletir.
- Her kelimenin konumunu belirtmek için bu kod çözücü girişlerine konumsal kodlama eklenir.

Decoder

- Her kelime sıralı olarak çıkışı oluşturur.