ÇANAKKALE 18 MART ÜNİVERSİTESİ

Ders Adı

Algoritma Analizi

Dersin Sorumlusu

Öğr. Gör. İsmail KAHRAMAN

Hazırlayanlar

150401028 – Bartu Utku SARP

150401045 – Hasan KAPAN

Konu

Long – Short Term Memory (LSTM)

Teslim Tarihi

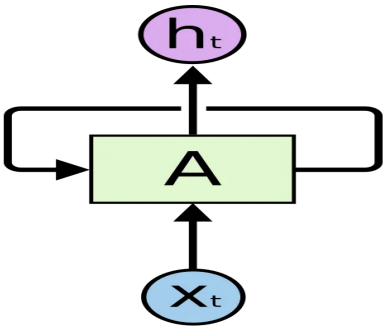
12.05.2020

Uzun - Kısa Süreli Bellek (Long – Short Term Memory) (LSTM)

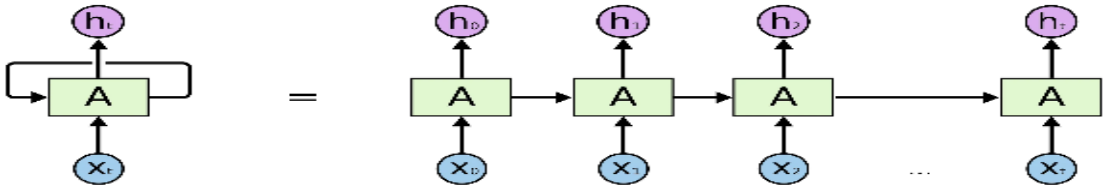
➢ Tekrarlayan Yapay Sinir Ağları (Recurrent Neural Network) (RNN)

Hepimizin bildiği gibi insanlar düşüncelerini hep sıfırdan başlatmaz. Bir konu üzerinde düşünürken her şeyi sıfırdan düşünmeye başlayamıyoruz. Yazıları okurken bile her kelimeyi önceki kelimelerle ilgili anlayışımıza göre değerlendiriyoruz. Çünkü tüm düşüncelerimizin bir sürekliliği var.

Fakat geleneksel sinir ağları bunu yapamaz ve bu büyük bir eksiklik gibi görülmektedir. Örneğin, bir filmin belirli noktalarında ne tür bir olay olduğunu sınıflandırmak istediğinizi düşünün. Geleneksel bir sinir ağının filmdeki olaylar arasındaki bağlantıyı kurabilmesi için bir sürekliliğe ihtiyaç olduğu açıktır. Tekrarlayan sinir ağları bu sorunu ele almaktadır. Bunlar, bilginin devam etmesine izin veren süreklilik ve döngüler içeren ağlardır.



Yandaki diyagramda, sinir ağının A parçası, Xt girdi değerlerine bakar ve bir çıktı (ht) üretir. Bir döngü, bilginin, ağın bir adımından diğerine iletilmesine izin vererek süreklilik oluşturulmasını sağlar. Bu döngüler, tekrarlayan sinir ağlarını gizemli gibi gösterir. Ancak, biraz daha derinlere inecek olursak aslında bu sinir ağlarının normal sinir ağlarının yapısından pek de farklı olmadığı ortaya çıkıyor. Tekrarlayan bir nöral ağ, her biri bir diğerine mesaj gönderen, aynı ağın çoklu kopyaları olarak düşünülebilir. Döngüyü açarsak ne olacağını düşünelim:



Bir Kayıtsız Tekrarlayan Sinir Ağı Diyagramı

➢ Kayıtsız Tekrarlayan Sinir Ağı

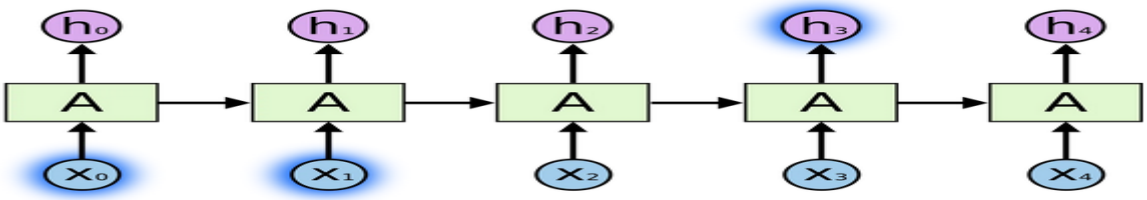
Bu zincir benzeri sinir ağının doğası, tekrarlayan sinir ağlarının diziler ve listelerle yakından ilişkili olduğunu ortaya çıkarır. Son birkaç yılda, RNN'leri çeşitli problemlere uygularken inanılmaz bir başarı elde edildi: konuşma tanıma, dil modelleme, çeviri, resim yazısı… Liste uzayıp gidiyor. Bu işlemleri yaparken kullanılan matematiksel formül aşağıdaki gibidir:

$$\mathbf{h}_t = \phi(W\mathbf{x}_t + U\mathbf{h}_{t-1})$$

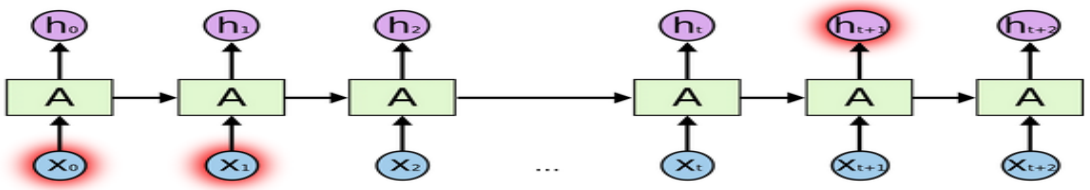
Bu başarılarda asıl önemli olan, "LSTM" lerdir. LSTM, çok sayıda görev için standart versiyondan çok daha iyi çalışan çok özel bir tür tekrarlayan sinir ağıdır. Nüks (tekrarlayan) nöral ağlara dayanan hemen hemen tüm heyecan verici sonuçlar, LSTM sayesinde elde edilir.

Uzun Vadeli Bağımlılık Problemi

Tekrarlayan sinir ağlarının cazibelerinden biri, önceki video karelerini kullanarak, mevcut karenin anlaşılmasını sağlamak gibi, önceki bilgileri mevcut göreve bağlayabilecekleri fikridir. RNN'ler bunu yapabilirse, son derece faydalı olurlar. Bazen, mevcut görevi yerine getirmek için sadece son bilgilere bakmamız gerekir. Örneğin, bir önceki kelimeye dayanarak bir sonraki kelimeyi tahmin etmeye çalışan bir dil modelini düşünün. "Bulutlar gökyüzündeyken" son kelimeyi tahmin etmeye çalışıyorsak, daha fazla bağlama ihtiyacımız yoktur. Bir sonraki kelime gökyüzü olacak gibi görünüyor. İlgili bilgiler ile ihtiyaç duyulan – tahmin edilecek bilgi arasındaki bilgi farkının ya da tahmin sayısının küçük olduğu bu gibi durumlarda, RNN'ler geçmiş bilgileri kullanmayı öğrenebilir.



Ancak, daha fazla içeriğe ihtiyaç duyduğumuz durumlar da var. "Fransa'da büyüdüm… akıcı Fransızca konuşuyorum." metnindeki son kelimeyi tahmin etmeye çalışın. Son bilgiler, bir sonraki kelimenin muhtemelen bir dilin adı olduğunu, ancak tahmini daraltmak istiyorsak Fransa bağlamında daha fazla geçmiş bilgiye ihtiyacımız olduğunu gösteriyor. İlgili bilgi ile tahmin arasındaki bilgi farkının olmasını gerekenden çok daha fazla. Ne yazık ki, bu fark büyüdükçe, RNN'ler bilgiyi bağlamayı öğrenemez hale gelir.



Teorik olarak, RNN'ler kesinlikle bu tür "uzun vadeli bağımlılıkları" çözümleyebilirler. Fakat ne yazık ki, pratikte, RNN'ler bu tür bağımlılıkları öğrenemiyor gibi görünüyor. Neyse ki LSTM'ler bu soruna sahip değil.

➢ Vanishing/Exploding Gradient (Gradient Yok Olması / Uçması)

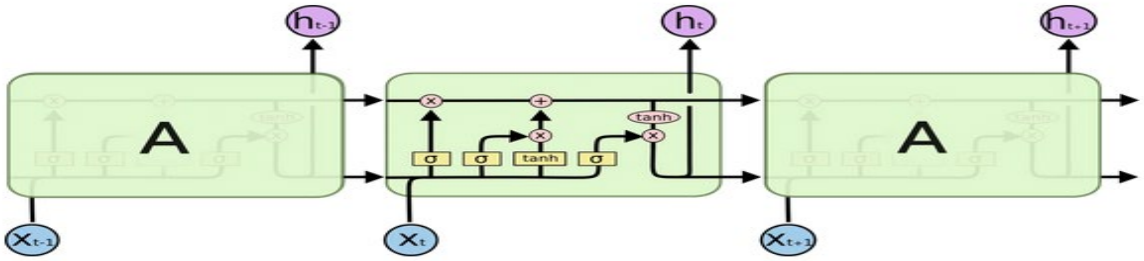
Gradient tüm ağırlıkları ayarlamamızı sağlayan bir değerdir. Ancak birbirine bağlı uzun ağlarda hatanın etkisi oldukça düşerek gradient kaybolmaya başlayabilir. Bu da doğru sonucu bulmayı imkansızlaştırmaktadır. Bütün katmanlar ve zamana bağlı adımlar birbirine çarpımla bağlı olduğundan, türevleri yok olma veya uçma yani aşırı yükselme tehlikesindedir.

Bu sorunu çözmek için de birkaç çözüm bulunmaktadır. W için uygun başlangıç değerleri seçmek yok olma etkisini azaltacaktır. Bir diğer çözüm ise sigmoid ve tanh aktivasyon fonksiyonları yerine ReLU kullanmaktır. ReLU fonksiyonunun türevi 0 veya 1 olduğundan böyle bir problem yaşanmayacaktır. Bir diğer yöntem ise bu problemi çözmek için tasarlanmış olan LSTM metodudur. LSTM'in her adımdaki güncellenen karmaşıklığı;

$$O(KH + KCS + HI + CSI) = O(W)$$
$$W = KH + KCS + CSI + 2CI + HI = O(W)$$

➢ LSTM Ağları

Genellikle “LSTM” olarak adlandırılan Uzun - Kısa Süreli Bellek ağları, uzun süreli bağımlılıkları öğrenebilen özel bir RNN türüdür. Hochreiter ve Schmidhuber tarafından 1997 vanishing gradient (kaybolan bilgi) problemini çözmek için geliştirildi. Daha sonra birçok kişinin katkısıyla düzenlenen ve popülerleşen LSTM şu anda geniş bir kullanım alanına sahiptir.

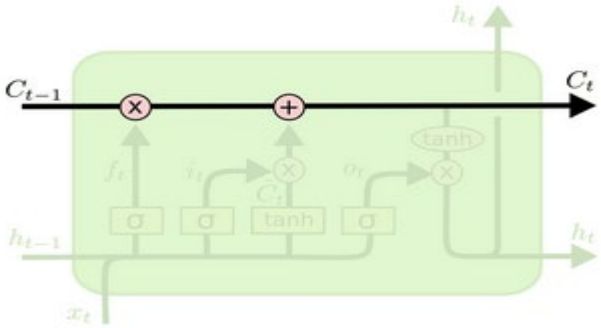


LSTM Network Yapısı

LSTM'ler uzun vadeli bağımlılık probleminden kaçınmak için açıkça tasarlanmıştır. Bilgiyi uzun süre hatırlamak, pratikte varsayılan davranışlarıdır. Tüm tekrarlayan sinir ağları, sinir ağının tekrar eden modüllerinin bir zinciridir. Standart ileri beslemeli sinir ağlarının aksine, LSTM'nin geri bildirim bağlantıları vardır. LSTM yapısında, tekrar eden modülün bir diğer farkı ise, tek bir neural network katmanı yerine, özel bir şekilde bağlı 4 katman bulunmasıdır. Bu katmanlara kapı da denmektedir. Normal akışın dışında dışarıdan bilgi alan bir yapıdır. Bu bilgiler depolanabilir, hücreye yazılabilir, okunabilir. Hücre neyi depolayacağını, ne zaman okumasına, yazmasına veya silmesine izin vereceğini kapılar sayesinde karar verir. Bu kapılarda bir ağ yapısı ve aktivasyon fonksiyonu bulunmaktadır. Aynı nöronlarda olduğu gibi gelen bilgiyi ağırlığına göre geçirir veya durdurur. Bu ağırlıklar recurrent ağın öğrenmesi sırasında hesaplanır. Bu yapı ile hücre, veriyi alacak mı bırakacak mı silecek mi öğrenir.

LSTM Ağlarının Arkasındaki Fikir

LSTM'lerin anahtarı, diyagramın üstünden geçen yatay çizgi olan hücre durumudur. Hücre durumu bir tür taşıma bandı gibidir. Sadece bazı küçük doğrusal etkileşimlerle tüm zincir boyunca aşağı doğru ilerler. Bilginin değişmeden akması çok kolaydır.



LSTM, kapı adı verilen yapılar tarafından düzenlenmiş hücre durumu ile, bilgi çıkarma veya ekleme yeteneğine sahiptir. Kapılar, isteğe bağlı olarak bilgi aktarmayı sağlarlar. Bir sigmoid sinir ağı tabakası ve noktasal bir çarpma işleminden oluşurlar. Sigmoid katmanı, sıfır ile bir arasında değerler üretir. Hücre durumu, bu sayılara bakarak her bir bileşenin ne kadarının geçmesi gerektiğini açıklar. Sıfır değeri “hiçbir şeyi bırakma” anlamına gelirken, bir değeri “her şeyi yapma” anlamına gelmektedir. Bir LSTM, hücre durumunu korumak ve kontrol etmek için bu kapılardan üçüne sahiptir.

KAYNAKÇA

- http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://devhunteryz.wordpress.com/2018/07/14/uzun-kisa-sureli-bellek-long-short-term-memory/
- https://medium.com/@hamzaerguder/recurrent-neural-network-nedir-bdd3d0839120
- https://tr.wikipedia.org/wiki/Long_short-term_memory
- https://ayearofai.com/rohan-lenny-3-recurrent-neural-networks-10300100899b
- http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/

KODLAR

- http://derindelimavi.blogspot.com/2017/03/kerasa-giris-2-lstm.html
- https://gist.githubusercontent.com/karpathy/d4dee566867f8291f086/raw/119a6930b670bced5800b6b03ec4b8cb6b8ff4ec/min-char-rnn.py
- https://github.com/jaungiers/LSTM-Neural-Network-for-Time-Series-Prediction/blob/master/run.py
- https://github.com/ivanarielcaceres/timeseries-lstm-keras/blob/master/timeseries-prediction.ipynb