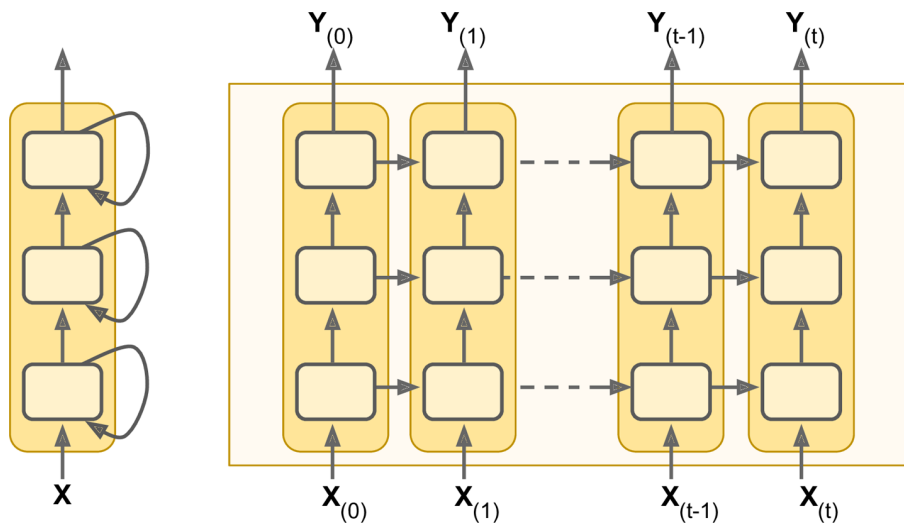


# Խորը և երկկողմանի անդրադարձ ցանցեր

Հայկ Կարապետյան

## 1 Խորը անդրադարձ ցանցեր

Սովորական RNN օգտագործելիս ունենք մուտքային հաջորդական տվյալներ և ցանկանում ենք output-ում նույնպես ստանալ հաջորդականություն (թարգմանված տեքստ) կամ մեկ արժեք (լավ, վատ դասակարգում): Այս դեպքում մենք օգտագործում ենք մի շերտ RNN: Dense շերտերի դեպքում կարող էինք ունենալ մեկ թաքնված շերտ (բացի input և output շերտերից) և դա կոչվում էր պարզ նեյրոնային ցանցը (simple neural network), իսկ մեկից ավել թաքնված շերտերից բաղկացած ցանցը կոչվում էր խորը նեյրոնային ցանց (deep neural network): RNN-ի դեպքում նույնպես կարող ենք ունենալ խորը ցանցեր: Դա նշանակում է, որ մի քանի RNN շերտ ենք ունենալու (Նկար 1):

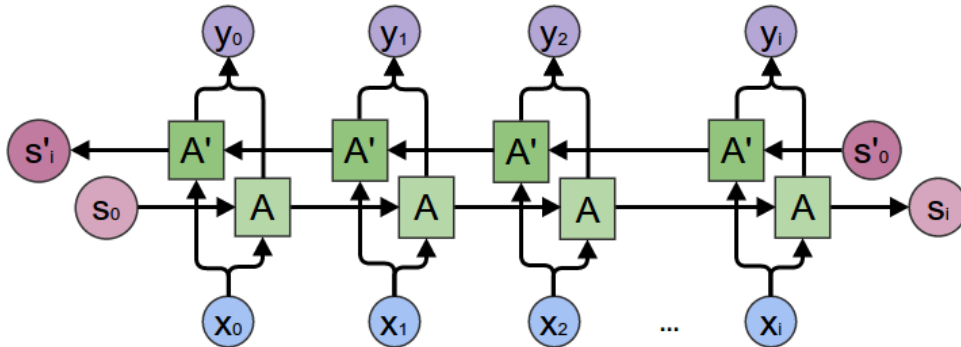


Նկար 1: Խորը անդրադարձ ցանց

RNN-ի դեպքում նույնպես կարող ենք ունենալ խորը ցանցեր: Ամեն RNN շերտ մուտքում ստանալու է հաջորդականություն, որը լինելու է նախորդ RNN շերտից ստացված output-ների հաջորդականությունը: Այստեղ կարող ենք որոշ չափով զուգահեռացում կատարել: Օրինակ՝ երկրորդ շերտի առաջին բլոկի output-ը հաշվելուն զուգահեռ կարող ենք հաշվել առաջին շերտի երկրորդ բլոկի output-ը:

## 2 Երկկողմանի անդրադարձ ցանցեր

Պարզ RNN-ում մուտքային տվյալները և history-ները փոխանցում էինք բլոկներին ձախից աջ: Երկկողմանի անդրադարձ ցանցերի հիմնական գաղափարն է ոչ միայն փոխանցել ձախից աջ, այլ նաև աջից ձախ (Նկար 2):



Նկար 2: Երկկողմանի անդրադարձ ցանց

Պատկերված երկու շերտը միջանկյալ իրար հետ կապված չեն: Այսինքն խորը ցանցերի դեպքում երկրորդ շերտը մուտքում ստանում էր նախորդ շերտի output-ները, իսկ այս դեպքում երկու շերտն էլ մուտքում ստանում են նույն մուտքային հաջորդական տվյալները: Երկու շերտն էլ ունեն գրոյական history ( $s_0, s'_0$ ): Վերջում ստանալու ենք երկու հաջորդական տվյալները և դրանք իրար ենք միացնելու (concat): Այս ցանցում ոչ միայն վերջին տվյալներն (բառեր) են ինֆորմացիա ունենալու սկզբնական տվյալների մասին, այլ նաև սկզբնական տվյալները (բառերը) ինֆորմացիա են ունենալու վերջին բառերի մասին: Օրինակ թարգմանության խընդրում վերջին բառից կախված կարող է փոփոխվել սկզբի բառի թարգմանությունը: