LSTM

11:26 AM Tuesday, February 2, 2021

בכל יחידת LSTM יש 4 שערים לוגיים
$3 - sigmoids \ \& \ 1 - tanh$

כל יחידה מזינה את עצמה בסקלר C
שתפקידו הוא זכרון לטווח ארוך

בשרטוטים נוהגים להציג רשת
Unfolded LSTMs תחת זמן,
אך לא מציגים את מספר היחידות
בשכבת ה LSTM

Long - Short Term Memory

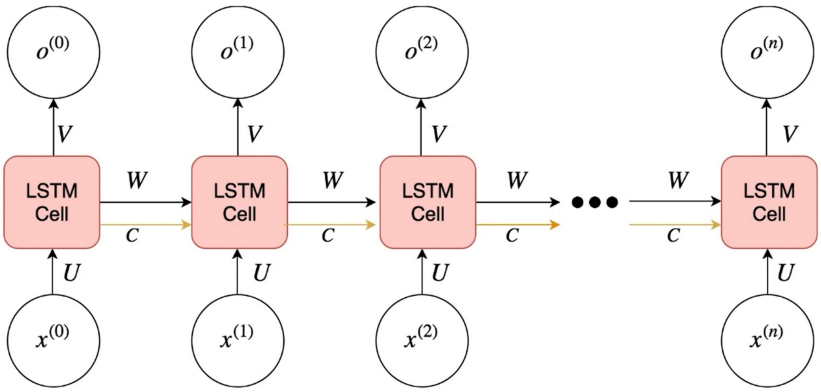
פותר בעיות
כמו

בעיית הגרדיאנטים נעלמים
Vanishing Gradient Problem

בעיית הגרדיאנטים המתפוצצים
Exploding Gradient Problem

התפוצצות או העלמות גרדיאנטים ברשתות עמוקות
מכיוון שאנחנו תמיד מעדכנים את אותן משקולות U, W , V בכל שכבה בRNN,
אם כל עדכון הגדול מ1 יגרור התפוצצות,
וכל עדכון קטן מ1 יגרור את איפוס המשקולות
LSTM משתמש בסוג של Gated Highway (נוירוני סיגמואיד)

מבט מבחוץ, כאשר אנחנו משתמשים ביחידת LSTM בודדה:



בנוסף לשערים הללו, יש לנו עוד וקטור $\bar{C}^t$ שתפקידו הוא לשנות את מצב התא C_t פונקציית האקטיבציה שלו היא Tanh
מכיוון שאנחנו יוצרים טווח ערכים ממורכז סביב 0, זה יאפשר לנו "לפזר" את הגרדיאנטים בצורה שווה,

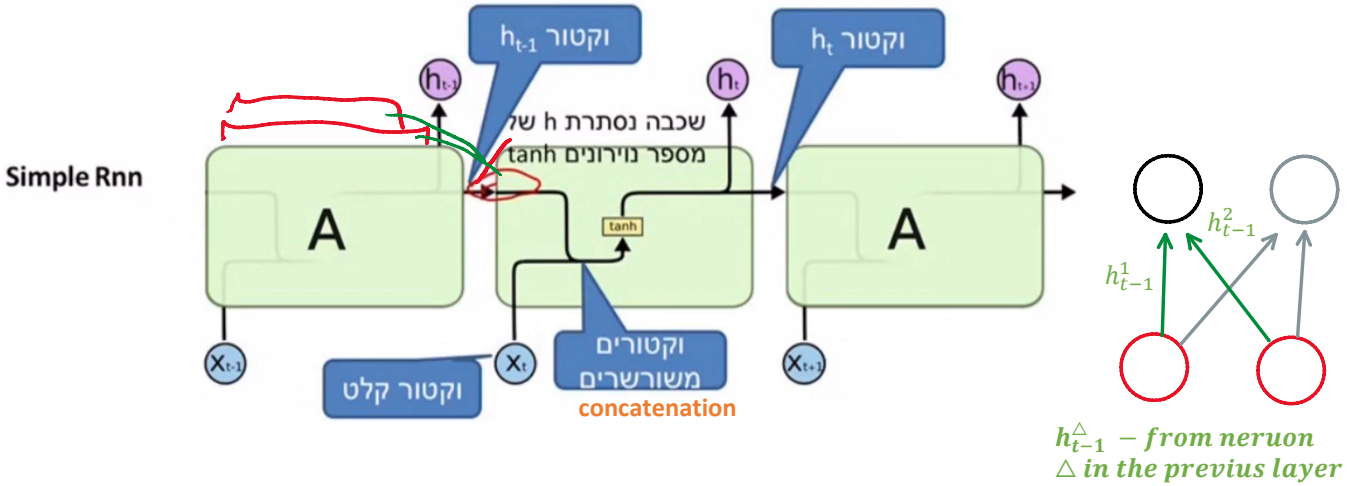
Why $tanh$ function on $\bar{C}^t$ → טווח ערים ממורכז סביב 0 → פיזור הגרדיאנטים בצורה שווה → מונע העלמות \ פיצוץ גרדיאנטים

ש. למה כל השערים הללו משתמשים בSigmoid לאקטיבציה?
ת. פונקציה גזירה שנותנת ערכים בין 0 ל 1

הרצאה 11a דקה 1:02:00

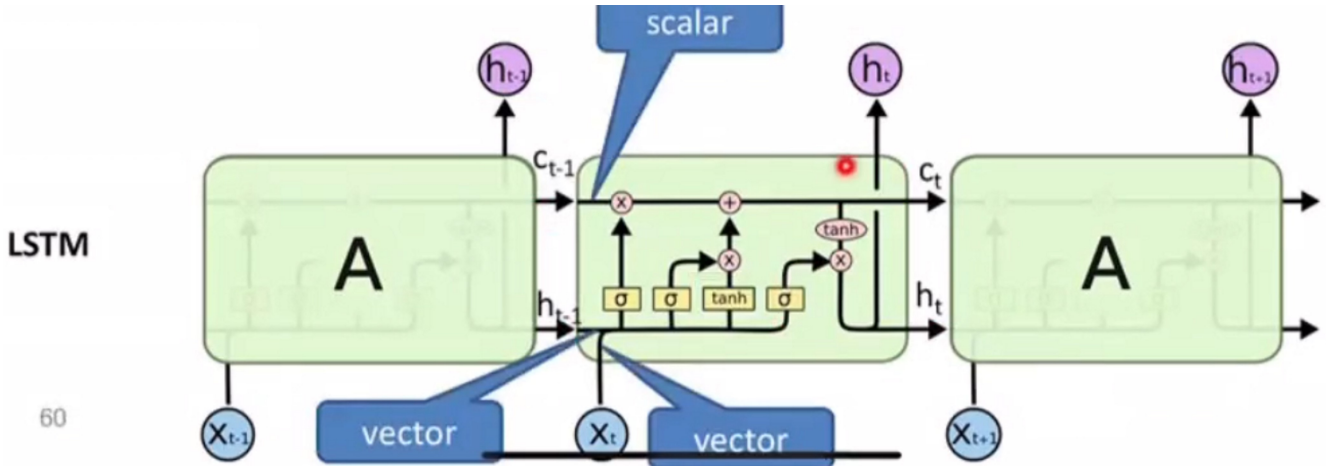
LSTM Input / Output

ברשת RNN רגילה, אנחנו מקבלים את h_{t-1} כווקטור שמגיע מכל הנוירונים בשכבה הקודמת ב-t1, כמו ברשת רגילה



ב LSTM אנחנו מקבלים את הוקטור h כמו בRNN רגילה (מכל אחד מהנוירונים בשכבה הקודמת)

אבל בLSTM בנוסף נקבל את מצב הנוירון C, שאותו הוא מקבל רק מעצמו בפרק הזמן הקודם, ולכן C סקלר



כניסות לLSTM

h_{t-1} **ווקטור** שמגיע מכל יחידות ה LSTM שפעלו בזמן $t-1$

C_{t-1} **סקלר** שמגיע מאותה יחידת LSTM שפעלה בזמן $t-1$

יציאות מLSTM

C_t - Cell State : **סקלר** נושא איתו זיכרון ארוך טווח וקצר טווח (נשלט על ידי מספר שערים)

h_t **סקלר** $[-1,1]$ המחושב על פי המצב האחרון של C_t (נשלט בעזרת שער בודד)

1:13:00

רכיבי LSTM

יחידת LSTM מכילה בתוכה 4 נוירונים נסתרים, לכל אחד מהם מטריצת משקולות וביאסים

כל אחד מהנוירונים מקבל את שרשור הוקטורים h_{t-1}, x_t

לדוגמה:

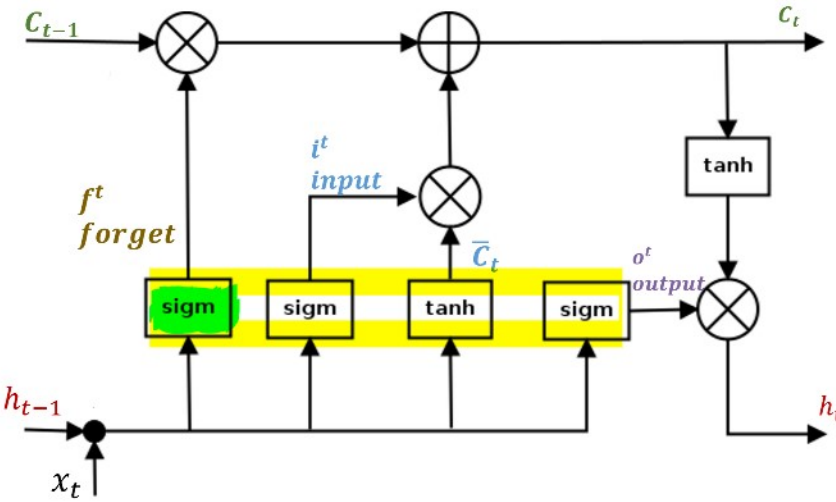
אם יש לנו רשת של 100 יחידות LSTM וקלט x מממד 50

אז מטריצת המשקולות לשער סיגמואיד יחיד, באחת מיחידות הLSTM תהיה $(100 + 50)$ כלומר וקטור באורך 150, ועוד ביאס בודד לעצמו

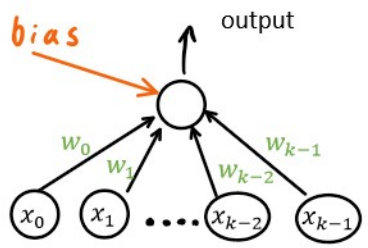
אם מסתכלים על כל 4 הנוירונים הנסתרים, אז לכל אחד מהם יש אותו גודל משקולות ולכן לכל יחידת LSTM יהיו $(100 + 50 + 1_{\text{bias}}) * 4 = 604$ משקולות

אם מסתכלים על כל 100 היחידות שיש ברשת שלנו אז יש $604 * 100 = 60{,}400$ משקולות לכל הרשת

מספר הנוירונים ברשת הם מספר השערים, סה"כ יהיה מספר יחידות ה LSTM כפול מספר השערים בכל אחת (4)
ולכן $100 * 4 = 400$



Sigmoid gate:



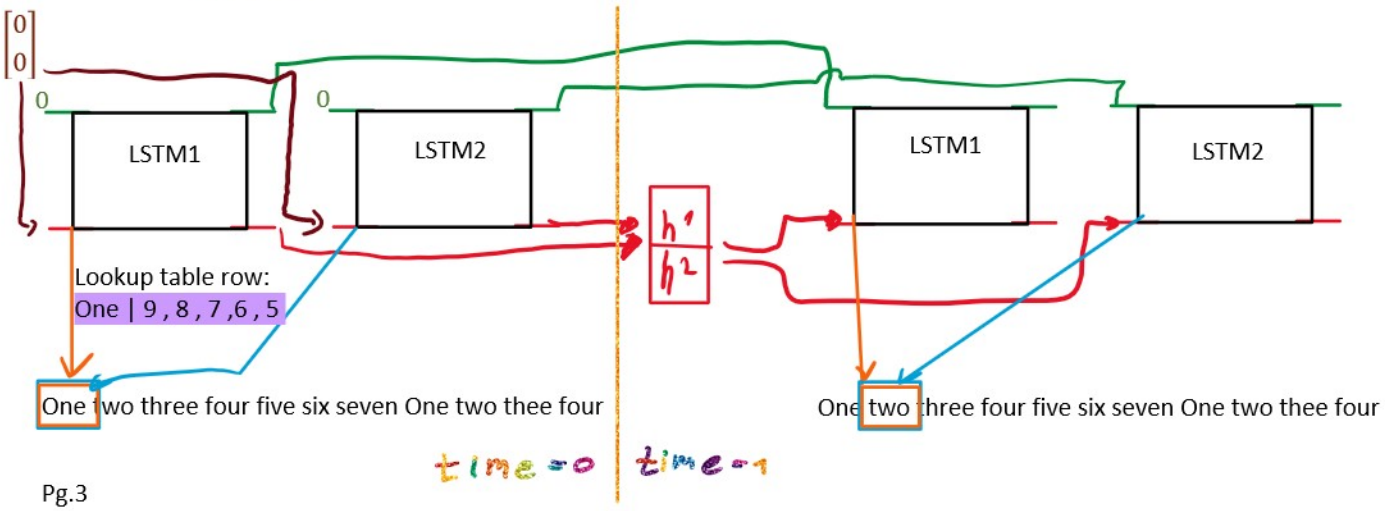
$$GateOutput = \sigma\left(\sum_{0}^{|[h_{t-1},x]|=k} (x_i * w_i) + b\right)$$

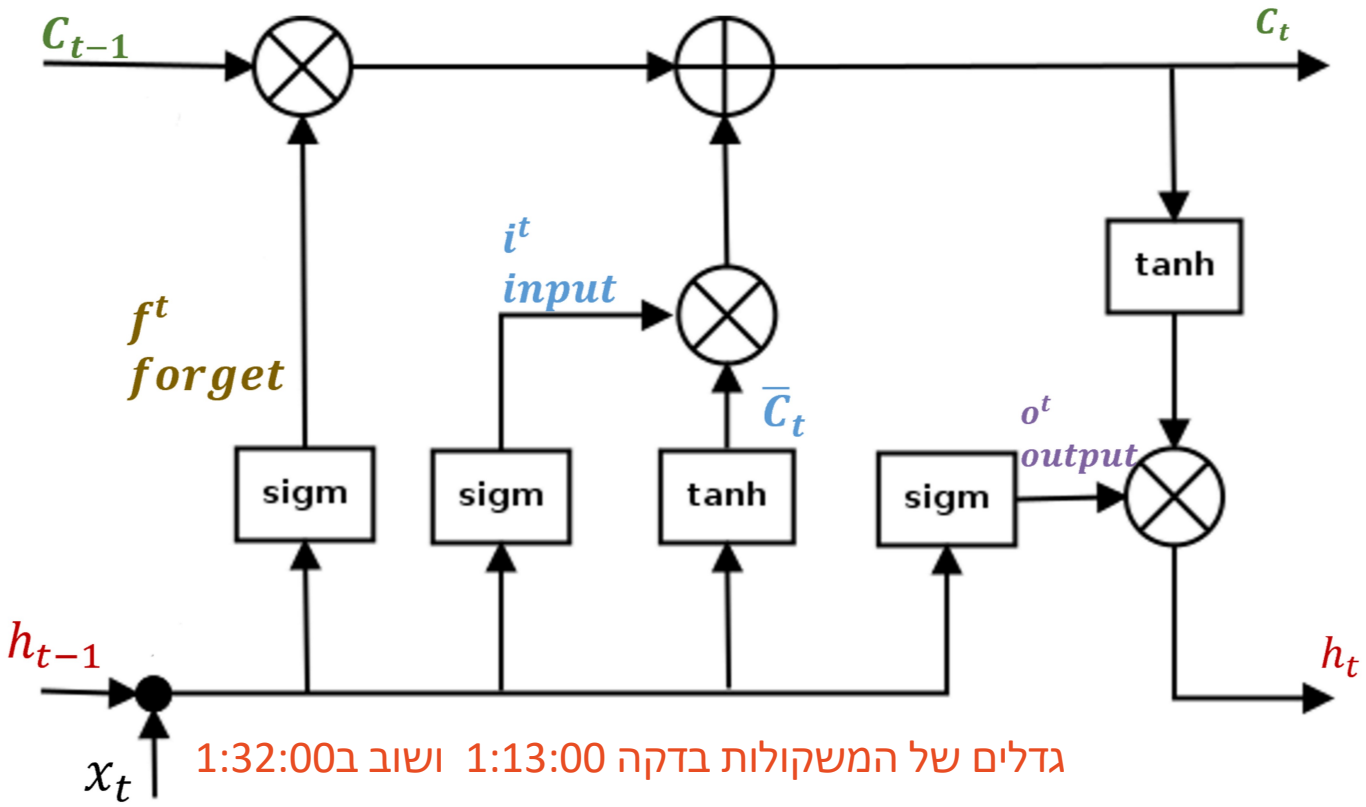
דוגמה מעשית:

אם יש לנו רשת של 2 יחידות LSTM, הקלט של כל אחת מהיחידות הוא ייצוג המילה (מהLookup table)

אז בהנחה שכל מילה מקודדת לוקטור באורך 5 אז אורך קלט
ואורך ה h_{t-1} יהיה 2 [כמספר יחידות ה LSTM]

ואז השרשור $[h_{t-1}, x_t]$ לדוגמה בודדת $[h^1, h^2, x^1, x^2, x^3, x^4, x^5]$ ---- ----------לאיטרציה הראשונה t=0 : [0,0,9,8,7,6,5]





גדלים של המשקולות בדקה 1:13:00 ושוב ב1:32:00

$$C_t = f^t \times C_{t-1} + i^t \times \overline{C}_t$$

$$\overline{C}_t = \tan h(W^c[h_{t-1}, x_t] + b^c)$$

$$i^t = \sigma(W^i[h_{t-1}, x_t] + b^i)$$

Input Gate

Controls weather the memory cell is updated based on the new input x_t

$$f^t = \sigma(W^f[h_{t-1}, x_t] + b^f)$$

forget Gate $f^t \in [0,1]$

Controls weather the memory cell is reset to 0.// controls how much of the old state should be forgotten.

$f_t = 1$ we preserve all the history

$f_t = 0$, we forget all history

$$o_t = \sigma(W^o[h_{t-1}, x_t] + b^o)$$

output gate

Weather the information of the current cell state is made visible

$W^{\aleph}$ *Weights of* $\aleph$

$b^{\aleph}$ Biases *of* $\aleph$

(Those Part Learn through back-prop)

$[h_{t-1}, x_t] \leftrightarrow h_{t-1}$ ***Concatinated with*** x_t

$h_{t-1} = [1,2,3], x_t = [4,5,6] \Rightarrow [h_{t-1}, x_t] = [1,2,3,4,5,6]$