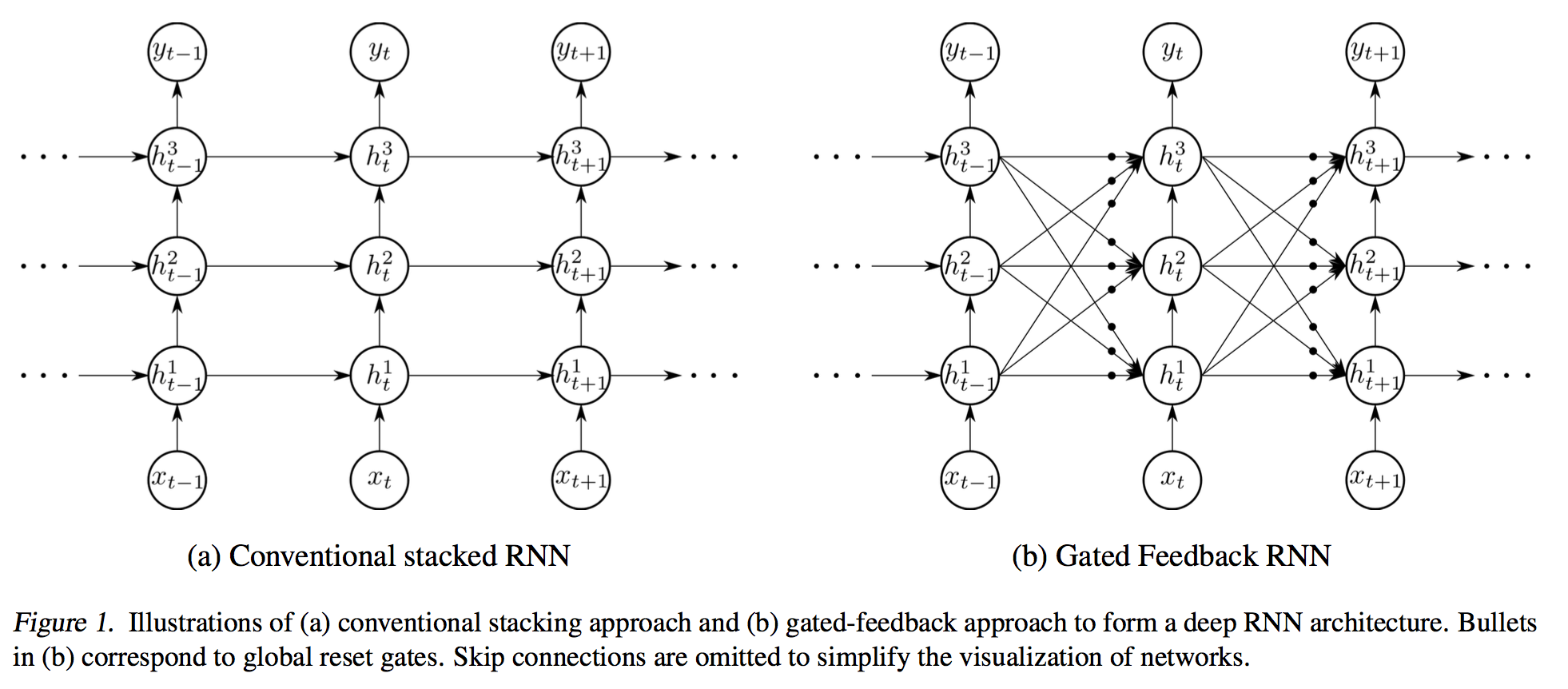
Paper Introduction:

Gated Feedback Recurrent Neural Networks

作者：Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, Yoshua Bengio  
介紹：李青峰、陳金博、羅羿牧

1. 摘要介紹

這篇文章主打介紹的RNN 架構是在RNN上增加了Gate的neuron還有feedback的功能，目的在於讓RNN。本篇文章所謂的Gate指的是記憶單元在時間軸上的連接有著可以調控資訊流通的開關。在這篇文章的Memory cell選用LSTM、GRU兩種能夠處理長時記憶和短時記憶的的記憶單元，另外再用以下右圖的架構連接：

（圖一）

LSTM/GRU被用在每個單元裡用來決定當前的input 和前一個state 是如何影響到當下的output 及 state 。如圖一所示，每個layer的state透過fully-connected的state-to-state transition有機會能影響下一個state的任何一個layer。這樣做的目的在於處理不同的timescale（有些時候資訊在兩字之間，有時候資訊在相隔很多個字後出現）的語意，因為有可以feedback到前一層單元的機構，所以不同長度的sequence dependencies可以互相傳達，達到有效分析語意的目的。

其中每條state-to-state的connection透過global reset gate來調控，計算如下：

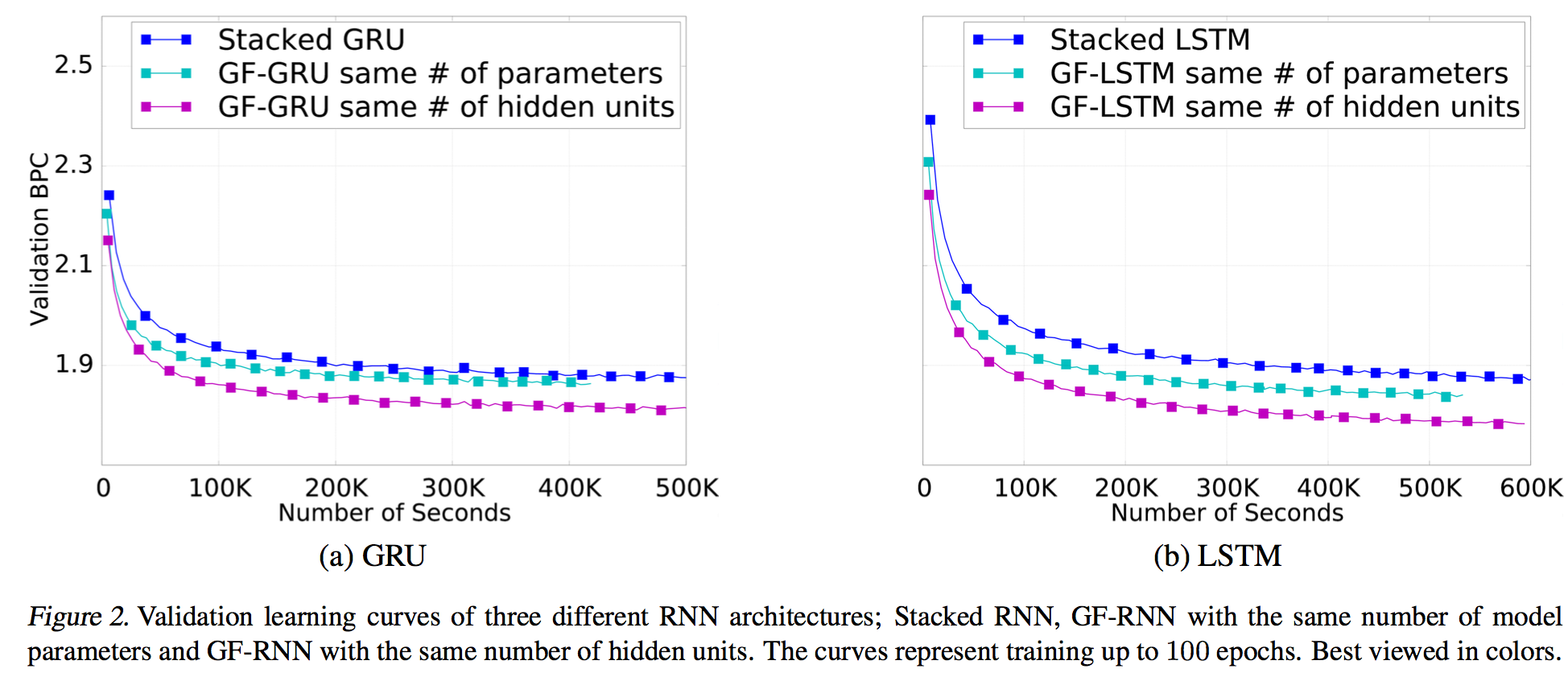
其中 是第 層到第 層間的weight vector（在圖一中以垂直向上的箭頭代表）， 是previous state到current state的weight vector。

Stacked LSTM第j個state記憶內容的計算如下：

之後透過 即可更新當前的state，其中 為LSTM output gate的值。

Stacked GRU的更新也遵循相似的原則，不再贅述。

因此上面的公式可以理解成從狀態 到 的資訊透過 這個「控制閥」過濾，而這個控制閥與當前於時間 的input 和所有之前的state 相關。因此這個架構在直覺上能夠透過學習權衡當前 input 對比之前的state 的重要性，並選擇性關閉state-to-state的資訊流通。

以上就是這篇提到的RNN架構，它提出的這個架構用在Python script 的語意理解上。Input是python script共有41個不同的符號，output是執行Python script的答案有13 種符號。最後，作者證實用Gated Feedback Recurrent Neural Networks （GF-RNN）比一般在時間軸上平行連接的方法還要好，如圖二：

圖(二)

衡量方法用的是BPC，一種值越低準確率越高的評量方法。

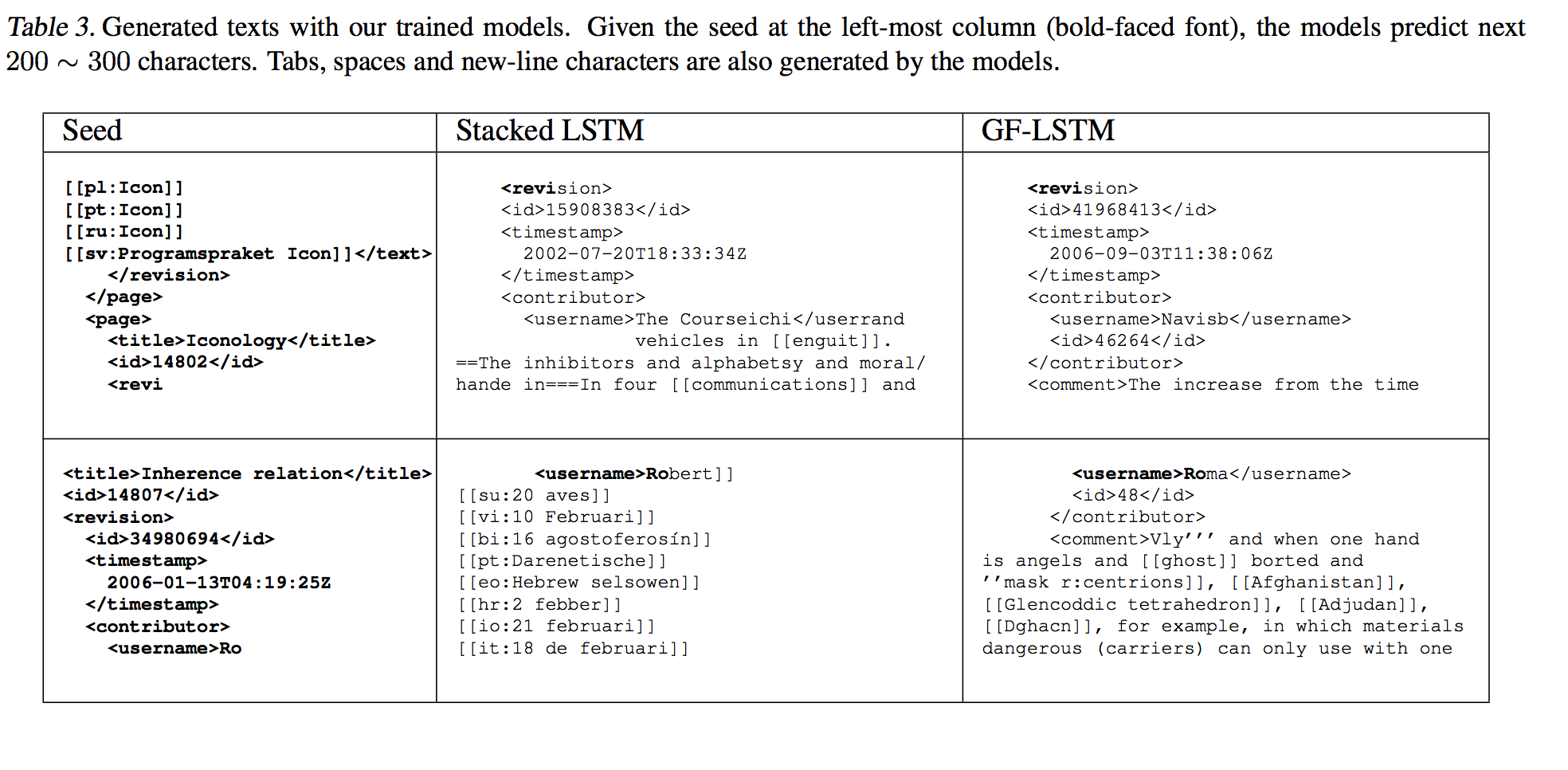
1. GF-RNN的特色

這種架構對顯著的特色在於能有效率地捕捉long-term和short-term的關聯性，應用和LSTM的開發目的是一樣的，只不過這是一種時間軸上的堆疊方式改良。

以往的RNN在時間軸上的連接方式只允許資訊從靠近 input layers 那端（對應短期關聯性）流向靠近 output layers 那端（對應長期關聯性），GF-RNN 則透過fully-connected的state-to-state transition同時允許另一個方向（long-term to short-term）的資訊傳遞。

這種架構在不同的layer的neurons會捕捉不同timescale的概念，愈靠近output layers的neurons會傾向捕捉long-term dependencies，並且強迫同一個layer形成一個module，捕捉同一個timescale的關聯性。這點與CW-RNN相同，但是達成的方式截然不同。CW-RNN硬性設定neurons update的週期與2的層數次方相關；然而在實務上的input sequence，某些時間上的關聯不見得會永遠在規律的時間間隔發生，因此GF-LSTM並沒有特別設定每個 layer 運作的週期，而是透過global reset gates的控制來 capture 不同 timescale 的 dependencies。最重要的是，global reset gates的開關是可以學習的（learnable），免除了捕捉timescale特徵的先天限制，我們認為這點是本文的架構能夠超越GF-LSTM架構表現的最主要原因。

在圖二上面的x軸代表訓練所需時間，實驗證實在此種架構下的RNN不論是在「相同參數量」或是「相同hidden units數量」之下，訓練時間都較傳統時間軸上的連接的方式短。

作者在實驗中強制設定global reset gate的值為1，得到明顯較差的效果，證實了gated feedback這種架構在學習這種sequence input的應用上是有提升訓練效率的。作者還執行了第二個實驗：

實驗內容是輸入初期的html code，讓程式產生出之後的程式（條件是必須符合語法）。Input可以從左邊的seed繼續延伸到之後的程式語法，並且證實使用GF-LSTM比較能夠辨認出 HTML tag 。由上面的實驗可以看出，用GF-LSTM的方法可以學習到tag的概念，展示了強大的sequence跨越長短timescale時間性特徵的捕捉。

這類的應用可以在Machine translation裡面看到，我們認為在未來或許可以應用在程式語言輸出的Generative NN，甚至是股票市場分析等需要重視「長周期與短周期特徵關聯」的應用。

1. 貢獻

李青峰：摘要介紹部分的架構數學解釋、GF-RNN特色部分與CW-RNN的比較

陳金博：GF-DNN特色部分於Python 指令碼上的比較、html語言的生成實驗

羅羿牧：摘要介紹部分的架構概覽、GF-DNN特色部分與LSTM/RNN的比較