

Sequence Models

1 Định nghĩa thuật ngữ:

1.1 ANN:

- **Khái niệm:** là một cách để máy tính học làm việc giống như bộ não con người, nhờ vào việc kết nối nhiều “nơ-ron nhân tạo” lại với nhau để giải quyết các bài toán phức tạp.
- **Ví dụ:** ANN giống như là một đội bạn cùng nhau giải bài tập về nhà, mỗi bạn (nơ-ron) nhận các thông tin khác nhau, tự xử lý rồi truyền kết quả đó cho bạn khác, cuối cùng tổng hợp lại thành một đáp án đúng.

1.2 RNN:

- **Khái niệm:** là một mô hình máy tính có khả năng nhớ lại một chút thông tin đã xử lý trước đó để hiểu, dự đoán hoặc tạo ra dữ liệu mới dựa trên chuỗi thông tin.
- **Ví dụ:** Bạn và bạn A chơi trò “truyền tin”. Bạn nói một câu cho bạn A, bạn A nhớ câu đó rồi nói tiếp cho bạn B, bạn B nhớ và nói cho bạn C, cứ thế đến cuối hàng.
Nếu câu nói quá dài, các bạn ở cuối hàng sẽ chỉ nhớ được một phần gần cuối, còn những chi tiết ở đầu câu đã bị quên mất.
→ RNN chỉ nhớ được gần, dễ quên thông tin ở phần đầu khi chuỗi dài.

1.3 LSTM:

- **Khái niệm:** là một mô hình máy tính có thể nhớ rất lâu các thông tin quan trọng từ đầu đến cuối chuỗi, nhờ có các “cổng thông minh” và “bộ nhớ đặc biệt”.
- **Ví dụ:** Bạn và bạn A chơi trò “truyền tin”. Bạn nói một câu dài cho bạn A. Bạn A ghi vào sổ rồi nói các keyword đó cho B. Cứ tiếp tục như vậy tới cuối hàng.
Người ở cuối hàng chỉ nhận được các thông tin keywords chứ sẽ không nhận được chính xác câu nói dài ban đầu

→ LSTM có trí nhớ dài hạn vì nó có cơ chế cổng và bộ nhớ dài hạn, nên có thể nhớ và truyền đạt lại đầy đủ các thông tin quan trọng.

1.4 GRU:

- **Khái niệm:** là một mô hình máy tính có khả năng nhớ được thông tin xa và quan trọng trong chuỗi dữ liệu, nhờ cơ chế 2 cổng tắc thông minh (cổng cập nhật và cổng xoá) để quyết định thông tin nào nên giữ lại, thông tin nào cần bỏ đi hay cần cập nhật.
- **Ví dụ:** Giả sử bạn được giao nhiệm vụ lắng nghe và tóm tắt lại một đoạn hội thoại dài giữa nhiều người.
Bạn chỉ cần 2 thao tác đơn giản:
 - Bật/ tắt mic (cổng xoá): Khi nào cần bỏ qua các thông tin không quan trọng, chỉ cần ghi lại những câu chứa thông tin chính.
 - Cập nhật vở ghi (cổng cập nhật): Bạn chọn lọc, chỉ ghi vào vở ghi những thông tin cần cập nhật, những thông tin cũ và không quan trọng thì bỏ qua.

2 Giải thích các tính chất của khái niệm:

2.1 Đặc điểm kiến trúc:

2.1.1 ANN:

- Xử lý dữ liệu độc lập, mỗi đầu vào không có mối liên hệ tự nhiên với nhau.
- Không có bộ nhớ, không có khái niệm "ngữ cảnh" hay "chuỗi".
- Rất mạnh cho phân loại, hồi quy dữ liệu tĩnh như hình ảnh, điểm số, thông tin bảng biểu.
- Đơn giản, nhanh, dễ huấn luyện.

2.1.2 RNN:

- Có thể xử lý dữ liệu chuỗi (tuần tự), nhớ được một phần thông tin từ các bước trước đó qua hidden state.
- Có liên kết hồi quy, cho phép truyền thông tin qua các bước thời gian.
- Dễ gặp vấn đề "vanishing gradient" khi chuỗi dài.

2.1.3 LSTM:

- Duy trì bộ nhớ dài hạn (cell state) kết hợp với 3 loại cổng (quên, nhập, xuất) giúp chọn lọc thông tin tốt.
- Kiến trúc phức tạp hơn, tính toán tốn thời gian.
- Xử lý ngữ cảnh, ngôn ngữ, dự báo dài hạn rất tốt.

2.1.4 GRU:

- Đơn giản hóa LSTM, chỉ có 2 cổng (cập nhật, xóa) giúp chọn lọc thông tin.
- Không có cell state riêng, nhờ đó chạy nhanh hơn LSTM.
- Vẫn duy trì được khả năng nhớ dài hạn, giải quyết vanishing gradient.
- Hiệu quả trong các bài toán cần tốc độ, độ phức tạp trung bình.

RNN ra đời để giải quyết bài toán dữ liệu chuỗi (văn bản, giọng nói, chuỗi thời gian...) mà ANN không thể làm được vì ANN chỉ xử lý dữ liệu độc lập, không hiểu thứ tự, không biết "ngữ cảnh" của các phần tử trong chuỗi. Tuy nhiên, RNN vẫn không xử lý tốt chuỗi dài do vanishing gradient, dễ quên thông tin ở phần đầu.

LSTM ra đời để giải quyết vanishing gradient của RNN, giúp nhớ được thông tin rất xa trong chuỗi, hiểu được "ngữ cảnh dài hạn", phù hợp cho dịch máy, hiểu đoạn hội thoại dài, dự báo chuỗi thời gian phức tạp mà RNN không làm được tốt. Nhược điểm của LSTM là kiến trúc phức tạp, tính toán nặng, chạy chậm.

GRU ra đời để đơn giản hóa LSTM, giảm độ phức tạp mà vẫn giữ được khả năng nhớ dài hạn, phù hợp cho các bài toán cần tốc độ xử lý cao, tài nguyên tính toán hạn chế mà hiệu quả vẫn gần bằng LSTM. GRU tối ưu khi cần cân bằng giữa hiệu suất và tốc độ, trong khi LSTM quá mạnh nhưng tốn kém.

2.2 So sánh:

Kiến trúc	Ưu điểm	Nhược điểm
ANN	Đơn giản, dễ huấn luyện; Hiệu quả với dữ liệu độc lập (hình ảnh, điểm số...); Tốc độ xử lý nhanh.	Không xử lý được dữ liệu tuần tự; Không nhớ được ngữ cảnh.
RNN	Xử lý được dữ liệu tuần tự (văn bản, giọng nói, chuỗi thời gian...); Có khả năng nhớ ngắn hạn (hidden state); Phù hợp với các chuỗi ngắn.	Dễ quên thông tin ở đầu chuỗi khi chuỗi dài; Mắc phải vanishing gradient; Kém hiệu quả với chuỗi dài.
LSTM	Nhớ được dài hạn nhờ cell state; Giải quyết hiệu quả vanishing gradient; Hiểu được ngữ cảnh phức tạp, phù hợp cho các bài toán chuỗi rất dài.	Kiến trúc phức tạp, nhiều tham số; Tốn thời gian huấn luyện; Tốn tài nguyên.
GRU	Nhớ dài hạn; Cấu trúc đơn giản hơn LSTM (chỉ 2 cổng, update và reset); Tốc độ huấn luyện nhanh hơn LSTM; Tiết kiệm tài nguyên hơn.	Ít tham số hơn LSTM nên đôi khi độ chính xác thấp hơn; Ít phổ biến hơn LSTM.

2.3 Mối liên hệ của các kiến trúc:

- ANN là mạng đơn giản nhất, chỉ xử lý từng thông tin riêng lẻ, không hiểu ngữ cảnh hay thứ tự.
- RNN phát triển từ ANN, có thể xử lý dữ liệu tuần tự (như câu văn, chuỗi số theo thời gian), nhưng chỉ nhớ được thông tin gần, dễ quên nếu chuỗi dài.
- LSTM là phiên bản nâng cao của RNN, thêm bộ nhớ riêng và các cổng điều khiển, giúp nhớ được lâu hơn, hiểu được ngữ cảnh phức tạp dù chuỗi rất dài.
- GRU là phiên bản đơn giản của LSTM, dùng ít cổng hơn, chạy nhanh hơn nhưng vẫn giữ được khả năng nhớ dài hạn.

→ Mối liên hệ giữa các kiến trúc này: Mỗi kiến trúc sau được sinh ra để khắc

phục điểm yếu của kiến trúc trước, giúp máy tính giải quyết từ bài toán đơn giản đến phức tạp, từ nhớ gần đến nhớ lâu, từ xử lý chậm đến xử lý nhanh.

2.4 Kiến trúc và bài toán NLP phù hợp:

2.4.1 ANN:

- Chỉ phù hợp với bài toán phân loại văn bản đơn giản, không cần nhớ ngữ cảnh hoặc thứ tự câu.
- Bài toán tiêu biểu: Text Classification, Sentiment Analysis.

2.4.2 RNN:

- Đã có thể xử lý dữ liệu tuần tự (văn bản, giọng nói, chuỗi thời gian...), nhưng chỉ nhớ được thông tin ngắn hạn, phù hợp với các chuỗi ngắn.
- Bài toán tiêu biểu: Các bài toán giống ANN và thêm vào Text Generation, Text Summarization, Text Simplification, Question Answering.

2.4.3 LSTM:

- Kiến trúc mạnh, phức tạp, nhớ rất lâu, phù hợp cho các bài toán NLP cần độ chính xác cao, ngữ cảnh dài.
- Bài toán tiêu biểu: Machine Translation, Question Answering, Text Summarization, Text Simplification, NER, Relation Extraction, Sentiment Analysis.

2.4.4 GRU:

- Đơn giản hóa LSTM, chạy nhanh hơn, tiết kiệm tài nguyên, hiệu quả với các bài toán yêu cầu tốc độ nhưng vẫn cần nhớ ngữ cảnh dài.
- Bài toán tiêu biểu: Machine Translation, Question Answering, Text Generation, Text Summarization, Sentiment Analysis, NER (chuỗi dài, cần tốc độ)

3 Bài tập:

Sử dụng dataset "<https://www.kaggle.com/datasets/iambestfeeder/10000-vietnamese-books>", tự cài đặt LSTM from scratch cho bài toán text generation, và report các đoạn text sinh ra. Phân tích đánh giá các kết quả này.

- Kiến trúc:
 - Vocabulary: 800 từ
 - Hidden dimension: 256
 - Embedding dimension: 128
 - Sequence length: 20
- Tính năng chính:
 - Memory-efficient chunked processing
 - Gradient clipping để tránh exploding gradients
 - Adaptive learning rate scheduling
 - Numerical stability với stable softmax
- Kết quả mong đợi:
 - Perplexity: <200
 - Model có thể tạo text có nghĩa cơ bản
- Điểm mạnh:
 - Implementation từ đầu giúp hiểu rõ thuật toán
 - Xử lý hiệu quả file dữ liệu lớn