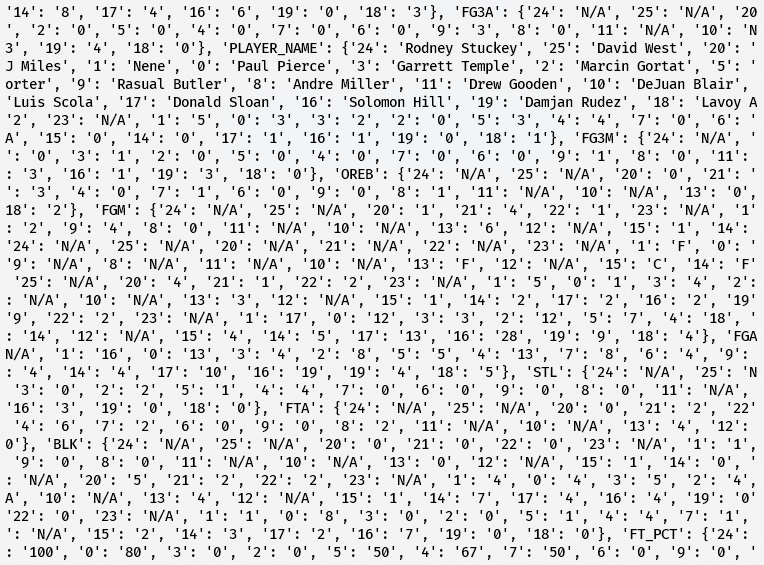
**CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP**

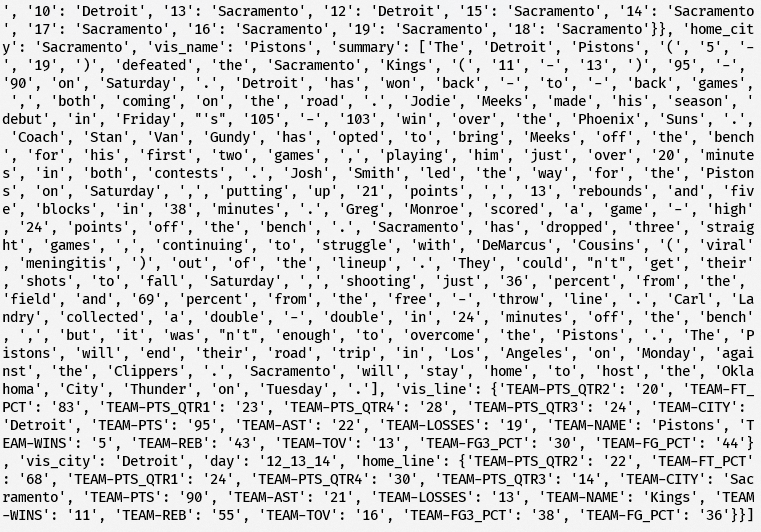
1. **Giới thiệu tập dữ liệu**

Mô hình sử dụng tập dữ liệu Rootowire:

<https://github.com/harvardnlp/boxscore-data.git>

Tập dữ liệu Rotowire là các bản tóm tắt các trận đấu bóng rổ NBA (do con người lập nên) được trình bài một cách hợp lý dựa vào sự canh chỉnh điểm số thành từng dòng tương ứng. Rotowire có 4853 bản tóm tắt riêng biệt, bao gồm các trận đầu NBA được chơi từ 1/1/2014 đến 29/3/2017, trong đó có một số trận đấu có nhiều bản tóm tắt. Các bản tóm tắt đã được chia ngẫu nhiên thành những tệp JSON bao gồm: 3398 tập huấn luyện (training), 727 tập thẩm định (validation) và 728 tập kiểm thử (test).

Hình. Tập dữ liệu huấn luyện (training)

Hình. Tập dữ liệu thẩm định (validation)



Hình. Tập dữ liệu kiểm thử (test)

Trong đó có các trường dữ liệu bao gồm:

* home\_name: tên đội chủ nhà (unicode).
* home\_city: Tên thành phố đội chủ nhà (unicode).
* vis\_name: tên đội khách (unicode).
* vis\_name: tên thành phố của đội khác (unicode).
* day: ngày thi đầu được định dạng %MM\_%DD\_%YY (unicode).
* summary: Mã của tóm tắt trận đầu.
* home-line: Dòng ghi nhận điểm của đội chủ nhà.
* vis-line: Dòng ghi nhận điểm của đội khách.
* box-core: Điểm hộp số.

Đối với trường dữ liệu Line-score (điểm theo dòng) bao gồm các thuộc tính như sau:

* TEAM-NAME: Tên đội bóng (unicode).
* TEAM-CITY: Thành phố cả đội bóng (unicode).
* TEAM-AST: Số lần hỗ trợ của đội bóng (integer as unicode).
* TEAM-FG3\_PCT: Tỷ lệ 3 con trỏ theo nhóm (integer as unicode).
* TEAM-FG\_PCT: Tỷ lệ bàn thắng được thực hiện bởi đội bóng (integer as unicode).
* TEAM-FT\_PCT: Tỷ lệ 3 điểm được thực hiện bởi đội bóng (integer as unicode).
* TEAM\_LOSSES: Đội thua (integer as unicode).
* TEAM-PTS: Tổng điểm của đội bóng (integer as unicode).
* TEAM-PTS\_QTR1: Điểm trong hiệp đầu tiên của đội bóng (integer as unicode).
* TEAM-PTS\_QTR2: Điểm trong hiệp thứ hai của đội bóng (integer as unicode).
* TEAM-PTS\_QTR3: Điểm trong hiệp thứ ba của đội bóng (integer as unicode).
* TEAM-PTS\_QTR4: Điểm trong hiệp thứ tư của đội bóng (integer as unicode).
* TEAM-REB: Tổng điểm Rebounds của đội (integer as unicode).
* TEAM-TOV: Tổng điểm Turnovers (integer as unicode).
* TEAM-WINS: Đội bóng thắng cuộc (integer unicode).

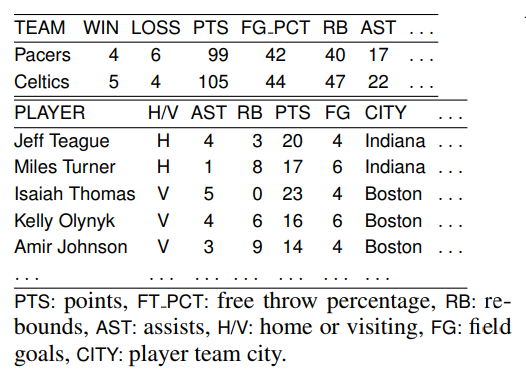
Đối với trường dữ liệu Box-core gồm có các thuộc tính sau:

* AST – cầu thủ hỗ trợ (row\_number -> integer as unicode).
* BLK - cầu thủ vị trí blocks (row\_number -> integer as unicode).
* DREB – cầu thủ ngăn chặn rebounds (row\_number -> integer as unicode).
* FG3A – cầu thủ tấn công 3 điểm (row\_number -> integer as unicode).
* FG3M – cầu thủ tạo 3 điểm (row\_number -> integer as unicode)
* FG3\_PCT – Tỷ lệ cầu thủ tạo 3 điểm (row\_number -> integer as unicode).
* FGA – cầu thủ với mục tiêu tấn công (row\_number -> integer as unicode).
* FGM – cầu thủ với mục tiêu ghi điểm (row\_number -> integer as unicode).
* FG\_PCT – Tỷ lệ cầu thủ ghi điểm (row\_number -> integer as unicode).
* FIRST\_NAME – Tên cầu thủ (row\_number -> unicode).
* FTA – Cầu thủ ném phạt (row\_number -> integer as unicode).
* FTM – Cầu thủ thực hiện ném phạt (row\_number -> integer as unicode).
* FT\_PCT – Tỷ lệ phần trăm cầu thủ ném phạt (row\_number -> integer as unicode).
* MIN – Số phút cầu thủ đã ra sân (row\_number -> integer as unicode).
* OREB – Cầu thủ phản hồi rebounds (row\_number -> integer as unicode).
* PF – Cầu thủ phạm lỗi cá nhân (row\_number -> integer as unicode).
* PLAYER\_NAME – Tên đầy đủ của cầu thủ (row\_number -> integer as unicode).
* PTS – Điểm của cầu thủ (row\_number -> integer as unicode).
* REB – Tổng điểm rebounds của cầu thủ (row\_number -> integer as unicode).
* SECOND\_NAME – Tên lót của cầu thủ (row\_number -> integer as unicode).
* START\_POSITION – Vị trí cầu thủ (row\_number -> unicode).
* STL – Cầu thủ lấy bóng (row\_number -> integer as unicode).
* TEAM\_CITY – Thành phố của đội bóng (row\_number -> unicode).
* TO – Cầu thủ turnovers (row\_number -> integer as unicode).

1. **Thiết kế và cài đặt giải pháp**
   1. **Mô hình xử lý dữ liệu**

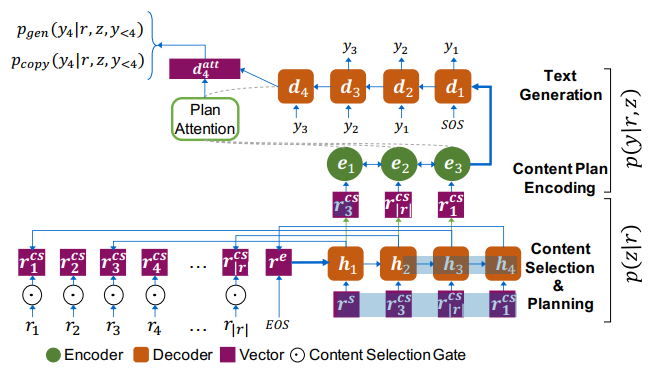
Đầu vào của mô hình là một bảng các bảng ghi (hình ). Mỗi bản ghi sẽ được gọi là rj, được biểu diễn dưới dạng , trong mỗi bản ghi gồm có 4 thuộc tính:

* Loại thuộc tính (rj,1 ; …; LOSS, CITY): là thuộc tính tên từng cột trong bản ghi bao gồm các thuộc tính như WIN, LOSS, PTS…
* Đối tượng (rj,2 ;…; Pacers, Miles Turner): là những thuộc tính liên quan đến đội bóng thi đầu và cầu thủ của đội bóng.
* Dữ liệu (rj,3 ;…; 11; Indiana): là thuộc tính liên quan đến thành phố và tổng ba chỉ số AST, RB, FG của cầu thủ.
* Dữ liệu (rj,4 ;…, H): là thuộc tính chỉ ra là đội khách hay đội nhà (cột H/V).



Hình. Một bản ghi đầu vào

Đầu ra của mô hình là một tài liệu được gọi là y bao gồm các thành phần y = y1 · · · với là độ dài của tài liệu.

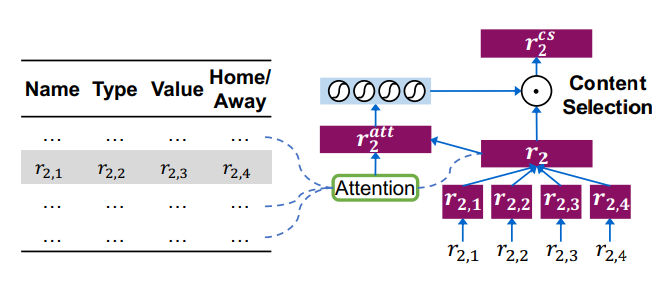
Hình. Mô hình lựa chọn và lập kế hoạch nội dung

Từ dữ liệu đầu vào là các bản ghi đã được lựa chọn nội dung để mã hóa dưới dạng các véc-tơ. Các véc-tơ sẽ được đi qua cổng tuyển chọn nội dung để tìm lọc ra các nội dung hữu ích, tránh sự trùng lập nội dung không cần thiết tạo thành một bản ghi mới. Tiếp theo, bản ghi mới này sẽ được sắp xếp theo trình tự kế hoạch tương ứng với thứ tự các véc-tơ trong bản ghi ban đầu. Đồng thời tại bước này, sử dụng bộ giải mã LSTM để tạo ra mã thông báo tương ứng với các vị trí trong đầu vào. Sau khi đã đồng bộ được thứ tự các véc-tơ, quá trình sẽ giải mã các thông tin văn bản dựa trên mạng nơ-ron tuần hoàn với các đơn vị LSTM, đó là sự đối chiếu của bản ghi với sự dự đoán từ trước. Đồng thời, mô hình cũng tăng cường thêm với bộ giải mã với cơ chế sao chép trực tiếp từ các phần giá trị của bản ghi trong kế hoạch nội dung.

* 1. **Quá trình xử lý dữ liệu**
     1. **Bộ mã hóa bản ghi (Record Encoder)**
* Đầu vào của mô hình là một bảng các bản ghi không có thứ tự, mỗi bản ghi được biểu diễn dưới dạng các tính năng .
* Công việc tiếp theo được thực hiện đó là nhúng các tính năng thành véc-tơ và sử dụng multilayer perceptron để lấy một biểu diễn véc-tơ rj cho mỗi bản ghi theo công thức:

Trong đó “ ; ” là chỉ phép nối véc-tơ, với n là tham số, ReLU là chức năng kích hoạt chỉnh lưu.

* + 1. **Cổng lựa chọn nội dung (Content Selection Gate)**
* Ngữ cảnh của một bản ghi rất quan trọng trong việc xác định giá trị của bản ghi đó có hữu ích hay không so với các bản ghi khác trong bảng. Vì thế mô hình sử dụng cơ chế cổng lựa chọn nội dung nhằm xác định tính hữu dụng của bản ghi trong bảng theo từng ngữ cảnh.
* Bước đầu tiên, mô hình tính điểm chú ý () qua đầu vào bảng, và sử dụng chúng để thu được một véc-tơ lưu ý () cho mỗi bản ghi , theo công thức:

Trong đó , là ma trận tham số, = 1

Hình. Cơ chế lựa chọn nội dung

* Tiếp theo cơ chế lựa cổng lựa chọn nội dung được áp dụng cho các véc-tơ . Đồng thời tìm được bản ghi đại diện mới được xác định dựa theo công thức:

Trong đó ⊙ biểu thị phép nhân khôn ngoan nguyên tố và cổng ∈ kiểm soát lượng thông tin xuất ra từ

Nói cách khác, mỗi phần tử trong  được cân nhắc bởi phần tử tương ứng của cổng lựa chọn nội dung .

* + 1. **Nội dung kế hoạch (**Content Planning**)**