

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

----------

**VŨ NGỌC HẢI**

**ỨNG DỤNG AI NÂNG CAO HIỆU QUẢ CHẤM ĐIỂM BÀI THI TRỰC TUYẾN**

**KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Công nghệ thông tin**

|  |
| --- |
| **HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  **KHOA CÔNG NGHỆ THÔNG TIN**  **VŨ NGỌC HẢI**  **ỨNG DỤNG AI NÂNG CAO HIỆU QUẢ CHẤM ĐIỂM BÀI THI TRỰC TUYẾN**  **KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**  **Ngành: Công nghệ thông tin**  **Cán bộ hướng dẫn: TS. Nguyễn Tất Thắng**  **HÀ NỘI – 2024** |

**NHẬN XÉT, ĐÁNH GIÁ, CHO ĐIỂM**

**(Của giảng viên hướng dẫn)**

………**………………………………………………………………………………………………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..**

Điểm: (Bằng chữ )

*Hà Nội,* ngày tháng năm 20…..

Giảng viên hướng dẫn

**NHẬN XÉT, ĐÁNH GIÁ, CHO ĐIỂM**

**(Của giảng viên phản biện)**

………**………………………………………………………………………………………………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..……………………………………………………..…………………**

Điểm: (Bằng chữ )

*Hà Nội,* ngày tháng năm 20…..

Giảng viên phản biện

**LỜI CẢM ƠN**

Trong thời gian tập trung nghiên cứu làm đề tài khoá luận mình đã chọn, em đã đạt được những thành quả riêng của mình. Để có được những thành quả đó em đã cố gắng, nỗ lực rất nhiều, bên cạnh đó em còn nhận được sự hỗ trợ và giúp đỡ của thầy giáo hướng dẫn, bạn bè và gia đình. Em xin được bày tỏ tình cảm và sự biết ơn của mình với những hỗ trợ và sự giúp đỡ này.

Đầu tiên em xin chân thành gửi lời cảm ơn đến Ban Giám Hiệu và toàn thể các thầy, các cô trong khoa Công nghệ thông tin, Học viện Công Nghệ Bưu Chính Viễn Thông đã tạo điều kiện giảng dạy, đào tạo và hướng dẫn cho em được học tập, nghiên cứu về lĩnh vực Công nghệ thông tin. Những kiến thức đó không chỉ là nền tảng, cơ sở để giúp em hoàn thành tốt khoá luận của mình mà còn là hành trang quý báu, vững chắc giúp em bước ra ngoài một cách tự tin hơn.

Và đặc biệt hơn để hoàn thành tốt khóa luận tốt nghiệp này, em xin gửi lời cảm ơn chân thành nhất đến thầy **TS. Nguyễn Tất Thắng** đã nhiệt tình hướng dẫn, hỗ trợ, giúp đỡ em trong suốt quá trình làm khóa luận tốt nghiệp.

Cuối cùng, em xin chúc toàn thể các thầy, cô luôn vui vẻ, mạnh khoẻ để tiếp tục đồng hành với bao thế hệ học trò trên con đường chinh phục tri thức.

Em xin chân thành cảm ơn !

Hà Nội, ngày… tháng… năm 2024

# **LỜI CAM ĐOAN**

Em xin cam đoan đề tài *“Ứng dụng AI nâng cao hiệu quả chấm điểm bài thi trực tuyến”* là là đề tài khoá luận của riêng em dưới sự hướng dẫn của thầy giáo: ***TS.Nguyễn Tất Thắng***. Ngoài ra, không có bất cứ sự sao chép của người khác. Nội dung trong khoá luận là các kết quả mà em đã nỗ lực nghiên cứu. Các số liệu, kết quả trong báo là hoàn toàn trung thực. Tất cả các nội dung tham khảo từ các tác giả khác đều đã được trích dẫn cụ thể và rõ ràng tại mục tài liệu tham khảo.

Hà Nội, ngày… tháng… năm 2024

Sinh viên

**Vũ Ngọc Hải**

# **TÓM TẮT**

Trí tuệ nhân tạo (AI) đã đóng góp rất lớn trong việc nâng cao trải nghiệm học tập theo nhiều cách khác nhau, đặc biệt là qua việc tự động hóa trong các phần mềm giáo dục. Một yếu tố quan trọng của phần mềm giáo dục là việc đánh giá các bài tập và kỳ thi của học sinh, và điều này có thể được cải thiện đáng kể thông qua việc chấm điểm tự động. Trong những năm qua, các nhà nghiên cứu đã áp dụng các phương pháp học máy (machine learning) được thiết kế đặc biệt để tự động hóa quy trình đánh giá điểm số. Tuy nhiên, những phương pháp này thường tập trung vào việc học mối quan hệ giữa các câu trả lời của học sinh và điểm số dự đoán, điều này yêu cầu một lượng lớn dữ liệu huấn luyện nhưng lại có khả năng tổng quát hạn chế đối với các lĩnh vực giáo dục mới. Gần đây, sự xuất hiện của các mô hình transformer được huấn luyện trước (pre-trained models) đã mang lại cơ hội mới để giải quyết vấn đề này. Trong đồ án này, ta sẽ tập trung vào loại câu hỏi trả lời ngắn (short answer question), loại câu hỏi yêu cầu học sinh trả lời bằng một hoặc vài câu ngắn gọn. Để xây dựng hệ thống chấm điểm tự động (Automated Short Answer Grading) cho loại câu hỏi này, ta sẽ sử dụng các mô hình huấn luyện trước (pre-trained models) như BERT và GPT. Việc áp dụng hai mô hình này sẽ giúp tối ưu hóa hiệu suất của hệ thống ASAG.

***Từ khoá:*** *Short answer · Automatic grading · Natural language processing*

**MỤC LỤC**

[LỜI CAM ĐOAN 4](#_Toc186484256)

[TÓM TẮT 5](#_Toc186484257)

[DANH MỤC TỪ VIẾT TẮT 8](#_Toc186484258)

[DANH MỤC BẢNG 12](#_Toc186484259)

[LỜI MỞ ĐẦU 13](#_Toc186484260)

[CHƯƠNG 1: TỔNG QUAN VỀ LĨNH VỰC CHẤM ĐIỂM TỰ ĐỘNG 15](#_Toc186484261)

[1.1 Câu hỏi trả lời ngắn (*Short answer question*) 16](#_Toc186484262)

[1.2 Quy trình chung phát triển hệ thống ASAG 16](#_Toc186484263)

[1.3 Các khía cạnh chung trong hệ thống ASAG 26](#_Toc186484264)

[*1.3.1 Tập dữ liệu (Data Sets) 27*](#_Toc186484265)

[*1.3.2 Quá trình xử lý ngôn ngữ tự nhiên (Natural Language Processing) 27*](#_Toc186484266)

[*1.3.3 Xây dựng mô hình (Build Model) 29*](#_Toc186484267)

[*1.3.4 Mô hình chấm điểm (Grading Model) 31*](#_Toc186484268)

[*1.3.5 Đánh giá mô hình (Model Evaluation) 33*](#_Toc186484269)

[*1.3.6 Hiệu quả (Effectiveness) 35*](#_Toc186484270)

[CHƯƠNG 2: ỨNG DỤNG BERT TRONG ASAG 38](#_Toc186484271)

[2.1 Giới thiệu về BERT 38](#_Toc186484272)

[2.2 Lý do sử dụng BERT 39](#_Toc186484273)

[2.3 Nền tảng của BERT 39](#_Toc186484274)

[2.4 Kiến trúc BERT 43](#_Toc186484275)

[2.5 Các giai đoạn đào tạo BERT 45](#_Toc186484276)

[2.6 Sử dụng BERT trong NLP 46](#_Toc186484277)

[*2.6.1 So sánh BERT với các mô hình khác 46*](#_Toc186484278)

[*2.6.2 Cách sử dụng BERT 49*](#_Toc186484279)

[2.7 BERT trong ASAG 51](#_Toc186484280)

[*2.7.1 Mô hình BERT Regressor cho hệ thống ASAG 53*](#_Toc186484281)

[*2.7.2 Xác định độ tương đồng giữa Key-Response 53*](#_Toc186484282)

[*2.7.3 Tích hợp Similarity Score với BERT Regressor 54*](#_Toc186484283)

[CHƯƠNG 3: SỬ DỤNG CHATGPT TRONG ASAG 55](#_Toc186484284)

[3.1 Giới thiệu ChatGPT 55](#_Toc186484285)

[3.2 Kiến trúc GPT 57](#_Toc186484286)

[3.3 ChatGPT trong ASAG 60](#_Toc186484287)

[CHƯƠNG 4: XÂY DỰNG HỆ THỐNG ASAG 62](#_Toc186484288)

[4.1 Kiến trúc tổng quan hệ thống 62](#_Toc186484289)

[4.2 Xây dựng mô hình 64](#_Toc186484290)

[*4.2.1 Kiến trúc BERT Regressor 64*](#_Toc186484291)

[*4.2.2 Kiến trúc ANN 65*](#_Toc186484292)

[*4.2.3 Tích hợp ChatGPT 66*](#_Toc186484293)

[*4.2.4 Xây dựng API chấm điểm dựa trên mô hình BERT 67*](#_Toc186484294)

[4.3 Triển khai ứng dụng web cho ASAG 67](#_Toc186484295)

[4.4 Kiểm thử và đánh giá hệ thống ASAG 70](#_Toc186484296)

[KẾT LUẬN ĐỀ 72](#_Toc186484297)

# **DANH MỤC TỪ VIẾT TẮT**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Ký tự viết tắt** | **Tiếng anh** | **Tiếng việt** |
| **A** | | | |
| 1 | ASAG | Automated Short Answer Grading | Tự động chấm điểm câu trả lời ngắn |
| 2 | AI | Artificial Intelligence | Trí tuệ nhân tạo |
| 3 | API | Application programming interface | Giao diện lập trình ứng dụng |
| 4 | ANN | Artificial neural networks | Mạng nơ-ron nhân tạo |
| **B** | | | |
| 5 | BERT | Bidirectional encoder representations from transformers | Mô hình học máy BERT |
| **D** | | | |
| 6 | DBMS | Database Management System | Hệ thống quản lý cơ sở dữ liệu |
| **M** | | | |
| 7 | MLM | Masked Language Modeling | Mô hình ngôn ngữ che giấu |
| **N** | | | |
| 8 | NSP | Next Sentence Prediction | Dự đoán câu tiếp theo |
| **L** | | | |
| 9 | LLM | Large language model | Mô hình ngôn ngữ lớn |

**DANH MỤC HÌNH VẼ**

[Hình 1: Các loại câu hỏi phổ biến có thể áp dụng ASAG. 15](#_Toc185894749)

[Hình 2: Hình ảnh biểu diễn quy trình phát triển hệ thống ASAG. 16](#_Toc185894750)

[Hình 3: Lịch sử hình thành ASAG qua 5 giai đoạn: concept mapping, information extraction, corpus-based methods, machine learning, evaluation 17](#_Toc185894751)

[Hình 4: Hình vẽ mô tả các kĩ thuật xử lý ngôn ngữ 28](#_Toc185894752)

[Hình 5: Hình vẽ mô tả các nhóm kĩ thuật xử lý ngôn ngữ 29](#_Toc185894753)

[Hình 6: Hình vẽ mô tả các cách thức khác nhau được sử dụng trong ASAG 32](#_Toc185894754)

[Hình 7: Hình vẽ so sánh giữa phương pháp thống kê và dựa trên quy tắc 33](#_Toc185894755)

[Hình 8: Hình vẽ mô tả ví dụ minh hoạ về hiệu quả của các hệ thống ASAG 35](#_Toc185894756)

[Hình 9: Hình vẽ mô tả các loại BERT 39](#_Toc185894757)

[Hình 10: Hình vẽ mô tả kiến trúc Transformer 41](#_Toc185894758)

[Hình 11: Hình vẽ mô tả kiến trúc của BERT 45](#_Toc185894759)

[Hình 12: Hình vẽ mô tả hai nhiệm vụ của quá trình đào tạo BERT 46](#_Toc185894760)

[Hình 13: Hình vẽ so sánh hiệu suất của BERT so với các mô hình khác đối với tập dữ liệu SQuAD 48](#_Toc185894761)

[Hình 14: Hình vẽ mô tả hiệu suất của BERT so với các mô hình khác đối với tập dữ liệu SWAG 49](#_Toc185894762)

[Hình 15: Hình vẽ mô tả các nhiệm vụ để kiểm tra sự hiểu biết của mô hình NLP 49](#_Toc185894763)

[Hình 16: Hình vẽ mô tả hiệu suất của BERT với các mô hình khác dối với tập dữ liệu GLUE 50](#_Toc185894764)

[Hình 17: Hình vẽ mô tả các bước sử dụng BERT trong hệ thống ASAG 53](#_Toc185894765)

[Hình 18: Hình vẽ mô tả chi tiết thành phần của ChatGPT 60](#_Toc185894766)

[Hình 19: Hình vẽ mô tả kiến trúc tổng quan hệ thống ASAG 63](#_Toc185894767)

[Hình 20: Hình vẽ mô tả kiến trúc ANN của BERT Regressor 65](#_Toc185894768)

[Hình 21: Hình vẽ mô tả kiến trúc của ANN được sử dụng để final grade 66](#_Toc185894769)

[Hình 22: Hình vẽ mô tả giao diện để lấy API\_KEY chatGPT 68](#_Toc185894770)

# **DANH MỤC BẢNG**

[Bảng 1: Bảng danh sách tài liệu tham khảo cho 35 hệ thống và 2 cuộc thi 18](#_Toc185894771)

[Bảng 2: Bảng mô tả kích thước dữ liệu được sử dụng để xây dựng mô hình 30](#_Toc185894772)

[Bảng 3: Bảng mô tả chi tiết kiến trúc của BERT 45](#_Toc185894773)

[Bảng 4: Bảng so sánh các phương pháp được sử dụng với các hệ thống ASAG đã tồn tại trước đó 52](#_Toc185894774)

[Bảng 5: Bảng so sánh kết quả kiểm thử đánh giá mô hình BERT và GPT 72](#_Toc185894775)

# **LỜI MỞ ĐẦU**

Trong thời đại chuyển đổi số hiện nay, công nghệ trí tuệ nhân tạo (AI) đang phát triển mạnh mẽ và trở thành công cụ quan trọng hỗ trợ nhiều lĩnh vực trong đời sống. Một trong những ứng dụng nổi bật của AI là trong giáo dục, đặc biệt là việc tự động hóa và nâng cao hiệu quả chấm điểm bài thi trực tuyến. Với sự gia tăng không ngừng của số lượng học viên và nhu cầu học tập trực tuyến, các phương pháp chấm điểm truyền thống ngày càng bộc lộ nhiều hạn chế, chẳng hạn như tốn thời gian, thiếu tính khách quan và không đáp ứng được khối lượng bài làm lớn trong thời gian ngắn.

Chấm điểm câu trả lời ngắn tự động (Automated Short Answer Grading - ASAG) là một kỹ thuật ứng dụng AI để đánh giá và chấm điểm các câu trả lời ngắn dựa trên ý nghĩa và mức độ liên quan đến đáp án mẫu. Phương pháp này không chỉ giúp giảm tải công việc cho giảng viên mà còn cung cấp phản hồi nhanh chóng, chính xác và công bằng cho học viên. ASAG đã được áp dụng trong nhiều hệ thống giáo dục trên thế giới, hỗ trợ đánh giá hiệu quả cả về mặt số lượng lẫn chất lượng. Khóa luận này sẽ xây dựng các dịch vụ trong việc quản lý tổ chức cuộc thi và đặc biệt sẽ có thêm sự trợ giúp của AI trong quá trình chấm điểm một cách tự động.

Trong khóa luận tốt nghiệp *“Ứng dụng AI nâng cao hiệu quả chấm điểm bài thi trực tuyến”*. Khóa luận gồm các phần và nội dung chính các phần sau:

**Chương 1: Tổng quan về lĩnh vực chấm điểm tự động**

Chương này sẽ giới thiệu tổng quan về lĩnh vực chấm điểm tự động, lịch sử hình thành và sự phát triển của phương pháp này từ trước đến nay.

**Chương 2: Ứng dụng BERT trong ASAG**

Chương này sẽ trình bày mô hình BERT một cách ngắn gọn, phân tích lý thuyết và ứng dụng của mô hình BERT trong hệ thống ASAG.

**Chương 3: Ứng dụng ChatGPT trong ASAG**

Chương này sẽ giới thiệu về ChatGPT, tính năng và cách ứng dụng ChatGPT trong hệ thống ASAG.

**Chương 4: Xây dựng hệ thống ASAG**

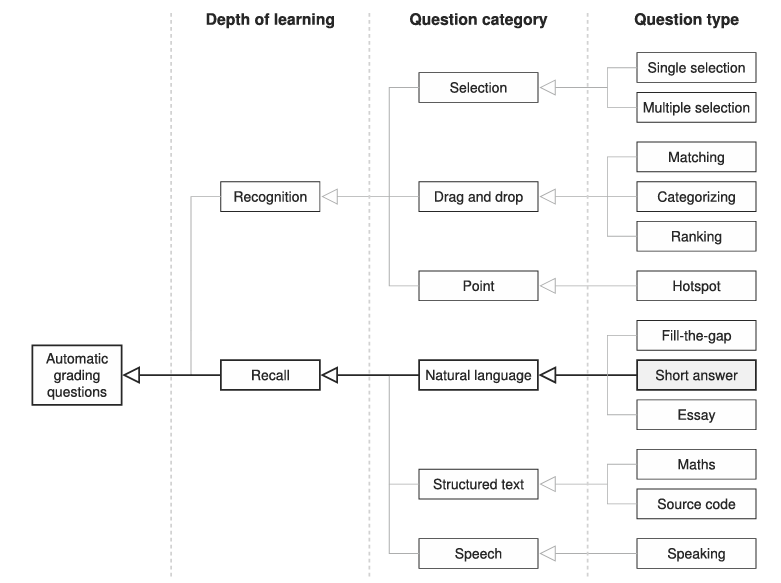
Chương này sẽ mô tả kiến trúc và quy trình xây dựng hệ thống ASAG.

**Kết luận**

Khóa luận được thực hiện dựa trên nghiên cứu cá nhân của em, vì vậy không thể tránh khỏi những thiếu sót. Em rất mong nhận được những góp ý, ý kiến đóng góp của các thầy cô để hoàn thiện khóa luận, từ đó tiếp thu thêm những kiến thức quý báu cho bản thân, góp phần phát triển sự nghiệp và tương lai sau này.

# **CHƯƠNG 1: TỔNG QUAN VỀ LĨNH VỰC CHẤM ĐIỂM TỰ ĐỘNG**

Việc đánh giá kết quả học tập thông qua các bài kiểm tra có thể bao gồm nhiều loại câu hỏi và phương pháp chấm điểm khác nhau. Các dạng câu hỏi có thể được thiết kế dưới nhiều hình thức, từ câu hỏi trắc nghiệm đơn giản cho đến các câu hỏi yêu cầu phản hồi bằng ngôn ngữ tự nhiên, như câu hỏi trả lời ngắn hoặc bài tiểu luận. Phương pháp chấm điểm có thể là thủ công (chấm điểm bằng tay) hoặc tự động thông qua các phương pháp tính toán. Trong khóa luận này, chúng ta sẽ tập trung vào câu hỏi trả lời ngắn (short answer question) và phương pháp chấm điểm tự động (automated short answer grading). Sự khác biệt giữa câu hỏi trắc nghiệm và câu hỏi trả lời ngắn là rõ ràng, nhưng sự phân biệt giữa các loại câu hỏi khác, như câu hỏi trả lời ngắn và bài luận, có thể không dễ nhận thấy.



Hình 1: Các loại câu hỏi phổ biến có thể áp dụng ASAG.

## **Câu hỏi trả lời ngắn (*Short answer question*)**

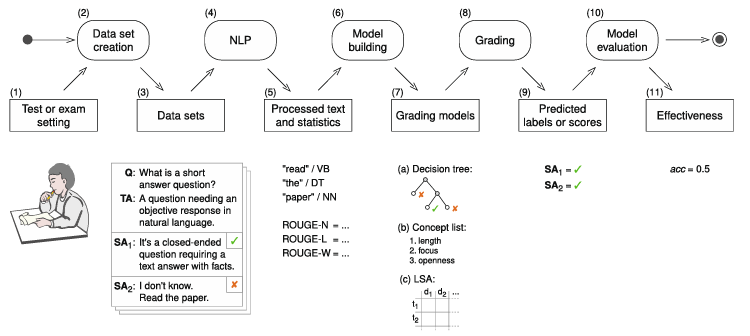
Một câu hỏi được coi là câu hỏi trả lời ngắn nếu đáp ứng ít nhất ba tiêu chí cụ thể sau:

* Đầu tiên, câu hỏi phải yêu cầu một câu trả lời liên quan đến kiến thức bên ngoài, thay vì chỉ yêu cầu trả lời dựa trên thông tin có sẵn trong chính câu hỏi.
* Thứ hai, câu hỏi phải yêu cầu câu trả lời được trình bày bằng ngôn ngữ tự nhiên.
* Thứ ba, độ dài câu trả lời nên nằm trong khoảng giữa một cụm từ và một đoạn văn ngắn.

Về phương pháp chấm điểm, một số loại câu hỏi khó chấm điểm hơn những loại khác. Ví dụ, câu hỏi trắc nghiệm có thể được coi là dễ chấm điểm vì mỗi câu chỉ có một đáp án đúng duy nhất. Ngược lại, việc chấm điểm câu trả lời bằng ngôn ngữ tự nhiên cho các câu hỏi trả lời ngắn lại khó khăn hơn nhiều, vì yêu cầu hiểu và phân tích ngôn ngữ tự nhiên.

Nghiên cứu về việc chấm điểm câu trả lời bằng ngôn ngữ tự nhiên đã bắt đầu từ năm 1966. Kể từ đó, việc chấm điểm tự động câu trả lời bằng ngôn ngữ tự nhiên đã phát triển thành một lĩnh vực nghiên cứu quan trọng. Việc áp dụng chấm điểm tự động mang lại nhiều lợi ích, bao gồm tiết kiệm chi phí và giảm thiểu sai sót do con người trong quá trình chấm điểm.

## **1.2 Quy trình chung phát triển hệ thống ASAG**

Quy trình phát triển một hệ thống Chấm điểm câu trả lời ngắn tự động (ASAG) bao gồm 11 thành phần, được chia thành 6 artifacts (tạo ra) và 5 processes (quy trình). Dưới đây là mô tả chi tiết:

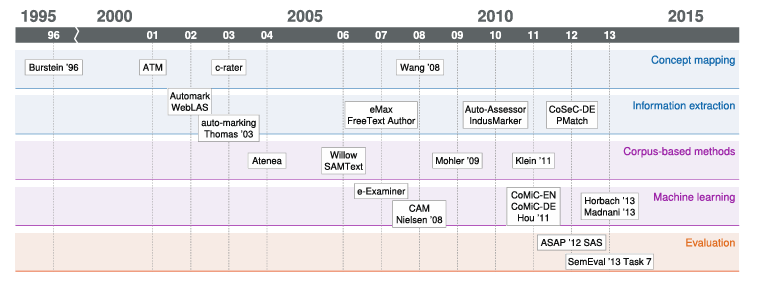
Hình 2: Hình ảnh biểu diễn quy trình phát triển hệ thống ASAG.

Đầu tiên, xác định các tài liệu phù hợp liên quan đến bài thi hoặc bài kiểm tra cần phát triển hệ thống. Tiếp theo, một hoặc nhiều bộ dữ liệu (datasets) được tạo ra bằng cách thu thập các câu hỏi (questions), đáp án của giáo viên (teacher answers), và câu trả lời của học sinh (student answers). Bộ dữ liệu này sẽ được lưu trữ trên ổ đĩa dưới các định dạng như CSV, XML, v.v.

Các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) được áp dụng để xử lý văn bản và tạo ra dữ liệu đã được xử lý (post-processed text). Dữ liệu này sau đó được sử dụng để xây dựng mô hình, dựa trên các phương pháp chấm điểm sử dụng học máy (machine learning) và trích xuất thông tin (information extraction).

Tiếp theo, quá trình chấm điểm tự động sẽ thực hiện dự đoán dựa trên các nhãn (labels) hoặc điểm số (scores). Những dự đoán này sẽ được đánh giá trong quá trình kiểm tra mô hình (model evaluation), và kết quả đánh giá sẽ được sử dụng để tính toán hiệu suất của mô hình (effectiveness).

**1.3 Lịch sử hình thành hệ thống ASAG?**

Hệ thống ASAG đã được hình thành qua nhiều giai đoạn khác nhau, mỗi giai đoạn đều nhấn mạnh một phương pháp riêng để thực hiện việc chấm điểm. Mỗi giai đoạn phát triển của hệ thống ASAG đều có những ý tưởng và phương pháp tiếp cận khác nhau:

Hình 3: Lịch sử hình thành ASAG qua 5 giai đoạn: concept mapping, information extraction, corpus-based methods, machine learning, evaluation

**Ánh xạ khái niệm (Concept Mapping)**: Ý tưởng của ánh xạ khái niệm là đánh giá câu trả lời của học sinh dựa trên một số khái niệm chủ chốt, xác định sự hiện diện hoặc vắng mặt của các khái niệm này. Các câu hỏi phù hợp với phương pháp này thường yêu cầu học sinh cung cấp giải pháp cho một vấn đề kèm theo lý giải, hoặc đưa ra nhiều lời giải thích khác nhau cho cùng một vấn đề. Ví dụ, (Burstein et al., 1996) đưa ra một câu hỏi yêu cầu học sinh cung cấp các lý do khác nhau để giải thích sự giảm số ca tử vong trong lực lượng cảnh sát theo thời gian. Ba khái niệm mẫu trong nghiên cứu này là: (1) "Tình hình kinh tế tốt hơn dẫn đến ít tội phạm hơn", (2) "Công nghệ y tế tiên tiến đã cứu sống nhiều người hơn", và (3) "Tội phạm hiện nay gặp khó khăn trong việc mua súng".

Bảng 1: Bảng danh sách tài liệu tham khảo cho 35 hệ thống và 2 cuộc thi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Hệ thống | Tài liệu tham khảo | Trang | Tài liệu tham khảo bổ sung |
| 1 | Atenea | Alfonseca and P´erez (2004) | 15 | Alfonseca et al. (2005); P´erez and Alfonseca (2005); P´erez et al. (2004a, b, 2005a, b, c); P´erez-Mar´ı et al. (2004) |
| 2 | ATM | Callear et al. (2001) | 12 |  |
| 3 | Auto-Assessor | Cutrone et al. (2011) | 14 | Formerly “Automarking”  (Cutrone and Chang 2010) |
| 4 | auto-marking | Sukkarieh et al. (2003) | 13 | Pulman and Sukkarieh (2005); Sukkarieh et al. (2004); Sukkarieh and Pulman (2005). |
| 5 | AutoMark | Mitchell et al. (2002) | 13 | Mitchell et al. (2003a, b). |
| 6 | Burstein ’96 | Burstein et al. (1996) | 12 |  |
| 7 | c-rater | Leacock and Chodorow (2003) | 12 | Attali et al. (2008); Sukkarieh (2010); Sukkarieh and Blackmore (2009); Sukkarieh and Bolge (2008, 2010); Sukkarieh and Kamal (2009); Sukkarieh and Stoyanchev (2009). |
| 8 | CAM | Bailey and Meurers (2008) | 17 | Bailey (2008). |
| 9 | CoMiC-DE | Meurers et al. (2011a) | 18 | Meurers et al. (2010); Ott et al. (2012). |
| 10 | CoMiC-EN | Meurers et al. (2011b) | 17 | Ziai et al. (2012) |
| 11 | Conort ’12 | Conort (2012) | 20 |  |
| 12 | CoSeC-DE | Hahn and Meurers (2012) | 15 |  |
| 13 | Dzikovska ’12 | Dzikovska et al. (2012) | 21 |  |
| 14 | e-Examiner | G¨utl (2007) | 17 | G¨utl (2008). Latterly “Electronic Assessor” Moser (2009) |
| 15 | eMax | Sima et al. (2009) | 14 | Gy¨orgy and Vajda (2007); Sima et al. (2007). Formerly “EVITA” (Csink et al. 2003). |
| 16 | ETS | Heilman and Madnani (2013) | 22 |  |
| 17 | FreeText Author | Jordan and Mitchell (2009) | 14 | Intelligent Assessment Technologies (2009); Jordan (2007, 2008, 2009a, b); Jordan et al. (2007); Swithenby and Jordan (2008); Willis (2010) |
| 18 | Horbach ’13 | Horbach et al. (2013) | 18 |  |
| 19 | Hou ’11 | Hou and Tsao (2011) | 18 | Hou et al. (2010, 2011, 2012). |
| 20 | IndusMarker | Siddiqi and Harrison (2008b) | 14 | Siddiqi and Harrison (2008a) |
| 21 | Klein ’11 | Klein et al. (2011) | 16 |  |
| 22 | Levy ’13 | Levy et al. (2013) | 22 |  |
| 23 | Madnani ’13 | Madnani et al. (2013) | 18 |  |
| 24 | Mohler ’09 | Mohler and Mihalcea (2009) | 16 | Mohler et al. (2011) |
| 25 | Nielsen ’08 | Nielsen et al. (2008b) | 17 | Nielsen et al. (2009) |
| 26 | PMatch | Jordan (2012a) | 15 | Jordan (2012b) |
| 27 | SAMText | Bukai et al. (2006) | 16 |  |
| 28 | SoftCardinality | Jimenez et al. (2013) | 21 |  |
| 29 | Tandella ’12 | Tandalla (2012 | 19 |  |
| 30 | Thomas ’03 | Thomas (2003) | 14 |  |
| 31 | UKP-BIU | Zesch et al. (2013) | 21 |  |
| 32 | Wang ’08 | Wang et al. (2008) | 13 |  |
| 33 | WebLAS | Bachman et al. (2002) | 13 |  |
| 34 | Willow | P´erez-Mar´ın and Pascual-Nieto (2011) | 16 | Pascual-Nieto et al. (2008, 2011); P´er´ez-Mar´ın (2007); P´er´ez-Mar´ın et al. (2006a, b, c, d, 2007). |
| 35 | Zbontar ’12 | Zbontar (2012) | 20 |  |
| 36 | ASAP ’12 SAS | Hewlett Foundation (2012) | 19 |  |
| 37 | SemEval ’13 Task 7 | Dzikovska et al. (2013) | 20 |  |

Việc ánh xạ khái niệm có thể được thực hiện ở mức độ câu, nhưng cũng có thể được phân tích chi tiết hơn ở cấp độ các đoạn văn nhỏ hơn (thường là các cặp từ hoặc cụm ba từ), và vấn đề này được gọi là ánh xạ mảnh (facet mapping). Ví dụ, (Nielsen et al., 2008a) thực hiện "facet mapping" trong câu hỏi yêu cầu học sinh giải thích về âm thanh tạo ra từ các nhạc cụ dây. Câu trả lời tham chiếu là: "Một dây dài tạo ra âm thanh trầm". Câu trả lời này có thể được chia nhỏ thành bốn mảnh: (1) dây/dài: "Có một dây dài", (2) tạo ra/dây: "Dây tạo ra thứ gì đó", (3) tạo ra/âm thanh: "Một âm thanh đang được tạo ra", và (4) âm thanh/trầm: "Âm thanh là trầm".

Cũng cần lưu ý rằng ánh xạ khái niệm (và ánh xạ mảnh) có mối quan hệ với việc suy diễn văn bản (textual entailment) (Levy et al., 2013). Mối liên kết này được thể hiện trong tài liệu về công cụ c-rater, nơi (Leacock và Chodorow, 2003) chỉ rõ rằng "công cụ chấm điểm phải có khả năng nhận diện khi nào một khái niệm xuất hiện và khi nào thì không". Mối liên kết này cũng được thể hiện thêm bởi (Levy et al., 2013) khi áp dụng chuyên môn về suy diễn vào việc chấm điểm trong hệ thống ASAG.

* **ATM (Automatic Text Marker)**: ATM (Callear et al., 2001) phân tách các câu trả lời của giáo viên và học sinh thành các danh sách khái niệm tối thiểu, mỗi khái niệm chỉ gồm một vài từ, và đếm số lượng khái niệm trùng khớp để chấm điểm. Mỗi khái niệm được xem như là đơn vị nhỏ nhất có thể gán trọng số, và các trọng số này sẽ được cộng lại để tính điểm tổng thể.
* **c-rater**: Công cụ Concept Rater (c-rater) (Leacock và Chodorow, 2003) được phát triển nhằm mục tiêu so khớp càng nhiều khái niệm càng tốt giữa câu trả lời của giáo viên và học sinh ở mức độ câu. Việc so khớp này dựa trên một tập hợp quy tắc và biểu diễn chuẩn của các văn bản, bao gồm các biến thể cú pháp, đại từ, từ đồng nghĩa, và sửa lỗi chính tả. Các câu trả lời của giáo viên được nhập vào dưới dạng các câu riêng biệt cho mỗi khái niệm, đơn giản hóa quá trình chấm điểm, vì mỗi khái niệm được xử lý riêng biệt. Kỹ thuật này tránh được sự cần thiết của các giải pháp gián tiếp, như việc chia câu hỏi thành nhiều phần (Jordan, 2009b), và được cho là có thể dẫn đến độ chính xác cao hơn (Sukkarieh và Blackmore, 2009).
* **Wang et al., 2008**: (Wang et al., 2008) đã so sánh ba phương pháp chấm điểm câu hỏi khoa học trái đất trong giáo dục trung học, bao gồm phương pháp ánh xạ khái niệm, học máy, hoặc kết hợp cả hai. Các phương pháp ánh xạ khái niệm sử dụng các đặc trưng bag-of-words như cosine trên vector TF-IDF hoặc máy vector hỗ trợ (SVM). Phương pháp học máy thuần túy sử dụng hồi quy SVM với unigrams, bigrams, và các cặp từ loại từ để chấm điểm một cách tổng thể, xem tất cả các khái niệm trong câu trả lời như một khối duy nhất.

**Trích xuất thông tin (Information Extraction)**: Trích xuất thông tin liên quan đến việc tìm kiếm và nhận diện các thông tin thực tế trong câu trả lời của học sinh. Các câu trả lời ngắn thường bao gồm các ý tưởng cụ thể, vì vậy trích xuất thông tin bao gồm các thao tác so khớp mẫu hiểu thức, chẳng hạn như sử dụng biểu thức chính quy (regular expressions). Kỹ thuật này có thể áp dụng cho cả dữ liệu có cấu trúc và không có cấu trúc.

* **AutoMark**: AutoMark (Mitchell et al., 2002) thực hiện việc so khớp mẫu như một hình thức trích xuất thông tin trên các biểu diễn cây phân tích cú pháp của câu trả lời giáo viên và học sinh. Hai phương pháp được mô tả là phương pháp "mù" (hoàn toàn tự động) và phương pháp "điều chỉnh" (bao gồm sự tham gia của con người để sửa đổi mô hình sau khi chấm điểm). Phương pháp mù được coi là định nghĩa chính xác nhất của ASAG, vì nó hoàn toàn tự động. Ngược lại, phương pháp điều chỉnh cho phép cải thiện mô hình khi có sẵn nguồn lực con người.
* **WebLAS (33)**: WebLAS (Hệ thống đánh giá ngôn ngữ dựa trên web) (Bachman et al., 2002) xác định các đoạn quan trọng trong câu trả lời của giáo viên thông qua các biểu diễn cây phân tích, sau đó yêu cầu giáo viên xác nhận từng đoạn và gán trọng số cho chúng. Giáo viên cũng được nhắc nhở để chấp nhận hoặc từ chối các lựa chọn thay thế có nghĩa tương tự. Việc so khớp biểu thức chính quy được thực hiện nhằm phát hiện sự hiện diện hoặc vắng mặt của mỗi đoạn trong câu trả lời của học sinh. Chấm điểm theo từng phần là khả thi vì mỗi đoạn được xem xét riêng biệt.
* **Auto-marking (4)**: Auto-marking (Sukkarieh et al., 2003) sử dụng các mẫu thủ công được thiết kế dựa trên tập dữ liệu huấn luyện để xây dựng mô hình. Hai mẫu được tạo cho mỗi câu hỏi, vì mỗi câu hỏi có giá trị 2 điểm. Đánh giá thực nghiệm cho thấy phương pháp này hiệu quả hơn so với mô hình k-nearest neighbor sử dụng các đặc trưng bag-of-words với trọng số TF-IDF. (Sukkarieh et al., 2004) cũng khám phá ý tưởng tạo mẫu thông qua phương pháp bootstrapping, tuy nhiên, lượng dữ liệu không được công bố, khiến việc so sánh với phương pháp thủ công trở nên khó khăn.
* **Thomas ’03 (30)**: Thomas (2003) xem xét ASAG như một bài toán so khớp mẫu boolean với sự hỗ trợ từ từ điển. Cụ thể, các cụm từ yêu cầu được định nghĩa dưới dạng các biểu thức boolean-AND, và các lựa chọn thay thế được chấp nhận thêm vào dưới dạng các biểu thức boolean-OR. Việc cho điểm các giải pháp đúng yêu cầu phải khớp chính xác.
* **eMax (15)**: eMax (Sima et al., 2009) yêu cầu giáo viên đánh dấu các yếu tố ngữ nghĩa cần thiết trong câu trả lời, chấp nhận hoặc từ chối các từ đồng nghĩa của các yếu tố này khi được nhắc nhở, và gán trọng số cho từng yếu tố để tính điểm cuối cùng (Sima et al., 2007). Phương pháp chấm điểm mang tính tổ hợp, trong đó tất cả các cách diễn đạt khả thi được xem xét khi thực hiện so khớp mẫu. Điểm số được gán đi kèm với xếp hạng độ tin cậy, vì vậy các trường hợp khó có thể được chuyển sang đánh giá thủ công.
* **FreeText Author (17)**: FreeText Author (Jordan và Mitchell, 2009) (trước đây là AutoMark như đã nêu) cung cấp giao diện đồ họa để nhập câu trả lời của giáo viên và chấm điểm câu trả lời của học sinh. Câu trả lời của giáo viên được tạo thành dưới dạng các mẫu cú pháp-ngữ nghĩa để khớp với câu trả lời của học sinh. Các mẫu này được tự động tạo từ ngôn ngữ tự nhiên của câu trả lời giáo viên, do đó không yêu cầu chuyên môn về xử lý ngôn ngữ tự nhiên. Thông qua giao diện, giáo viên có thể chỉ định các từ khóa bắt buộc từ câu trả lời của mình và chọn từ các từ đồng nghĩa do hỗ trợ từ điển cung cấp. Cả câu trả lời chấp nhận và không chấp nhận đều có thể được định nghĩa, và câu trả lời của học sinh được chấm điểm theo mức độ khớp mẫu.
* **Auto-Assessor (3)**: Auto-Assessor (Cutrone et al., 2011) tập trung vào việc chấm điểm các câu trả lời dạng một câu đã được chuẩn hóa của học sinh dựa trên việc so khớp tọa độ bag-of-words và các từ đồng nghĩa từ WordNet (Pedersen et al., 2004). So khớp tọa độ trong ASAG liên quan đến việc so khớp các thuật ngữ riêng lẻ giữa câu trả lời của giáo viên và học sinh. Trong Auto-Assessor, mỗi từ khớp chính xác được tính một điểm, các từ liên quan từ WordNet được tính điểm một phần, và các từ không khớp sẽ không được tính điểm.
* **IndusMarker (20)**: IndusMarker (Siddiqi và Harrison, 2008a) được sử dụng để thực hiện so khớp mẫu ở cấp độ từ và cụm từ nhằm chấm điểm câu trả lời của học sinh. Đây được gọi là “so khớp cấu trúc”. Các cụm từ đáng ghi điểm được định nghĩa bằng ngôn ngữ đánh dấu XML gọi là Question Answer Markup Language. Thông qua "trình chỉnh sửa cấu trúc", văn bản và số điểm có thể được nhập cho từng cụm từ.
* **CoSeC-DE (12)**: CoSeC-DE (Comparing Semantics in Context) (Hahn và Meurers, 2012) sử dụng phương pháp Lexical Resource Semantics (LRS) (Richter và Sailer, 2003) để tạo các biểu diễn trừu tượng của văn bản. Ý tưởng này được minh họa qua việc so sánh các câu sau: (1) “Thỏ thắng rùa.” (2) “Rùa thắng thỏ.” (3) “Rùa bị thỏ đánh bại.” Ở đây, (1) và (2) tương đương theo mô hình bag-of-words, nhưng (1) và (3) tương đương theo mô hình LRS. Cụ thể, các biểu diễn LRS của câu trả lời giáo viên và học sinh được mô hình hóa thành đồ thị, và một sự liên kết dựa trên ngưỡng được thực hiện để phát hiện ý nghĩa tương đương.
* **PMatch (26)**: Hệ thống này có khả năng chấm điểm các câu trả lời rất ngắn, dài đến một câu. Hệ thống thực hiện so khớp mẫu ở cấp độ từ, trong đó tất cả các từ bắt buộc, gốc từ và từ đồng nghĩa được phép trong các câu trả lời đúng được so khớp thông qua các biểu thức chính quy với câu trả lời của giáo viên.

**Phương pháp dựa trên Corpus**: Các phương pháp dựa trên corpus khai thác các đặc tính thống kê của các tập văn bản lớn. Mặc dù các phương pháp này thường được sử dụng cho các ứng dụng với văn bản dài (Bukai et al., 2006), nhưng chúng cũng có thể hữu ích trong việc giải thích các từ đồng nghĩa trong các câu trả lời ngắn. Việc chỉ sử dụng từ vựng gốc từ câu trả lời của giáo viên có thể giới hạn các câu trả lời đúng có thể được xác định. Một kỹ thuật phổ biến để mở rộng từ vựng là sử dụng các cặp văn bản song ngữ để phân tích tần suất các cặp từ được dịch sang cùng một bản dịch ngôn ngữ thứ hai. Các từ đồng nghĩa với bản dịch phổ biến có thể được kết hợp vào câu trả lời của giáo viên.

* **Atenea (1)**: Atenea (Alfonseca và Pérez, 2004) ban đầu sử dụng chỉ số BLEU (BiLingual Evaluation Understudy) (Papineni et al., 2002) để chấm điểm. Chỉ số này dựa trên sự trùng lặp n-gram và độ dài mẫu được chuẩn hóa. Giá trị trên thang đo [0,1] sau đó được chia tỷ lệ theo thang điểm thích hợp. Quan trọng là, (Alfonseca và Pérez, 2004) lập luận rằng BLEU nên xem xét cả độ chính xác (precision) và độ hồi tưởng (recall), vì BLEU gốc chỉ tập trung vào độ chính xác. Phần mở rộng này được gọi là ERB (Evaluating Responses with BLEU) (Pérez-Marín, 2004). Atenea được chứng minh là hiệu quả hơn so với các mô hình cơ sở như so khớp tọa độ và mô hình không gian vector (Pérez et al., 2004a; Pérez-Marín, 2004). Sau đó, phân tích ngữ nghĩa (Latent Semantic Analysis - LSA) (Landauer et al., 1998) được thêm vào, và một sự kết hợp có trọng số giữa điểm số BLEU và LSA được thực hiện (Pérez et al., 2005a). LSA là một phương pháp dựa trên corpus tương tự như mô hình không gian vector, giúp xử lý sự biến đổi từ vựng. Phương pháp kết hợp BLEU và LSA mang lại sự cải thiện nhất quán so với các công trình trước đó (Pérez et al., 2005a, c). Vì sự kết hợp có trọng số giữa BLEU và LSA được sử dụng thay vì các đặc trưng riêng lẻ, Atenea được coi là một phương pháp dựa trên corpus.
* **Willow (34)**: Willow (Pérez-Marín và Pascual-Nieto, 2011) là hệ thống kế thừa của Atenea (được mô tả ở trên). Tuy nhiên, nghiên cứu về ASAG trong Willow chủ yếu có tính chất gia tăng, với phần lớn công trình mới mang tính sư phạm hơn. Ví dụ, kết quả hiện tại của học sinh được tham khảo để xác định mức độ khó của câu hỏi mới (Pérez-Marín et al., 2006c), các đề xuất chủ đề được đưa ra cho học sinh để tiếp tục học (Pascual-Nieto et al., 2011), và chức năng tự đánh giá được giới thiệu (Pascual-Nieto et al., 2008).
* **SAMText (27)**: SAMText (Short Answer Measurement of TEXT) (Bukai et al., 2006) áp dụng một biến thể của LSA dựa trên cấu trúc dữ liệu chỉ mục ngược (inverted index), với nội dung được cung cấp từ một web crawl với các tài liệu liên quan đến chủ đề. Ngược lại, LSA thông thường sử dụng cấu trúc ma trận dựa trên các corpus lớn để mô hình hóa sự liên quan ngữ nghĩa. Bukai et al. (2006) lập luận rằng việc sử dụng chỉ mục ngược và crawling phù hợp hơn cho câu trả lời ngắn so với câu trả lời dài, vì web crawl có thể được điều chỉnh cho từng chủ đề thay vì cố gắng mô hình hóa toàn bộ ngôn ngữ cùng một lúc.
* **Mohler ’09 (24)**: Mohler và Mihalcea (2009) phát triển một số hệ thống để nghiên cứu các phương pháp chấm điểm không giám sát bằng cách so sánh riêng lẻ tám phương pháp đo lường ngữ nghĩa dựa trên tri thức và hai phương pháp dựa trên corpus. Các phương pháp dựa trên tri thức bao gồm Hirst và St-Onge (1998), Jiang và Conrath (1997), Leacock và Chodorow (1998), Lesk (1986), Lin (1998), Resnik (1995), shortest path (Mohler và Mihalcea, 2009), và Wu và Palmer (1994). Hai phương pháp dựa trên corpus là phân tích ngữ nghĩa (Explicit Semantic Analysis - ESA) (Gabrilovich và Markovitch, 2006) và LSA. Ngoài việc so sánh các phương pháp đo lường này, (Mohler và Mihalcea, 2009) còn xem xét việc kết hợp các câu trả lời tốt nhất của học sinh với câu trả lời của giáo viên để mở rộng từ vựng câu trả lời của giáo viên, và họ phát hiện rằng điều này mang lại hiệu quả.
* **Klein ’11 (21)**: Klein et al. (2011) triển khai một hệ thống LSA, với ý tưởng chính là sử dụng các câu trả lời của học sinh làm mô hình LSA thay vì các văn bản chung từ các nguồn khác. Một số văn bản sau đó được đánh dấu thủ công, và điều này tạo thành mô hình để tự động chấm điểm các văn bản còn lại. Các vấn đề chính là lựa chọn các văn bản cụ thể để đánh dấu thủ công và xác định số lượng tổng thể. Về số lượng, quá trình này được lặp lại cho đến khi đạt được ngưỡng tương quan cụ thể. Về lựa chọn, ba phương pháp được xem xét: chọn ngẫu nhiên, phân cụm, hoặc chọn văn bản ít tương đồng nhất với những văn bản đã được đánh dấu. Phương pháp thứ ba đã chứng minh hiệu quả nhất. Lợi ích của phương pháp Klein ’11 là việc tự động chọn lựa các bài nộp cần đánh dấu thủ công và giảm thiểu số lượng này. Tuy nhiên, nhược điểm của phương pháp này là mức độ hiệu quả mong muốn chỉ đạt được sau khi đánh dấu thủ công phần lớn các câu trả lời của học sinh, chiếm đến 83% tổng số tình huống. Một vấn đề khác là phương pháp này phụ thuộc vào các tham số, vì phải xác định cả chiều không gian ngữ nghĩa và tham số ngưỡng tương đồng.

**Học máy (Machine Learning):** Học máy thường sử dụng một số phép đo được trích xuất từ các kỹ thuật xử lý ngôn ngữ tự nhiên sau đó được đưa ra một đầu điểm duy nhất hoặc một lớp bằng các sử dụng mô hình phân loại hoặc hồi quy (regression model). Đây cũng là hướng tiếp cận để thực hiện việc chấm điểm tự động (ASAG) cho đồ án này.

* e-Examiner (14): e-Examiner (Gütl, 2007) sử dụng các chỉ số ROUGE (Lin, 2004) làm đặc trưng học máy. Các chỉ số này được kết hợp thành một mô hình hồi quy tuyến tính. Phần lớn nghiên cứu còn lại tập trung vào kiến trúc hệ thống, trong đó thiết kế linh hoạt cho phép dịch vụ được sử dụng dưới dạng độc lập hoặc như một thành phần trong một hệ thống hiện có, chẳng hạn như nền tảng e-learning.
* CAM (8): CAM (Content Assessment Module) (Bailey và Meurers, 2008) sử dụng mô hình k-nearest neighbor và các đặc trưng đo lường mức độ trùng lặp nội dung ở nhiều cấp độ ngôn ngữ giữa câu trả lời của giáo viên và học sinh. Các loại trùng lặp bao gồm unigrams và trigrams từ, các cụm danh từ, ngưỡng tương đồng văn bản, loại từ, gốc từ (lemmas), và từ đồng nghĩa. Điều thú vị là thuật ngữ khác thường được sử dụng để mô tả hai nhiệm vụ đánh giá. Thứ nhất, “phát hiện lỗi ngữ nghĩa” đại diện cho bài toán phân loại nhị phân (đúng/sai). Thứ hai, “chuẩn đoán lỗi ngữ nghĩa” đại diện cho bài toán phân loại 5 nhãn với tập các nhãn mở rộng cho lớp tiêu cực (phân loại 5 lớp). Do đó, các thí nghiệm đại diện cho các phương pháp chấm điểm tổng hợp và hình thành.
* Nielsen ’08 (25): Nielsen et al. (2008b) đánh giá hệ thống học máy của họ trên dữ liệu SciEntsBank, sau này trở thành một phần của cuộc thi SemEval ’13 Task 7 (Dzikovska et al., 2013). Nhiệm vụ phân loại là 5 nhãn, dựa trên các câu hỏi khoa học cấp tiểu học từ lớp 3 đến lớp 6, với các nhãn “hiểu”, “mâu thuẫn”, “mâu thuẫn nội tại”, “lập luận khác”, và “không được đề cập” (những nhãn này sau đó được ánh xạ lại cho SemEval ’13 Task 7). Hệ thống sử dụng các đặc trưng như đặc trưng từ vựng (loại từ, sự khớp gốc từ, và xác suất suy diễn) và đặc trưng cú pháp (loại quan hệ phụ thuộc và khoảng cách chỉnh sửa). Một cây quyết định C4.5 được sử dụng để phân loại.
* CoMiC-EN (10): CoMiC-EN (Meurers et al., 2011b) và CoMiC-DE (hệ thống tiếp theo) xuất phát từ dự án “So sánh ý nghĩa trong ngữ cảnh” (Comparing Meaning in Context - CoMiC). CoMiC-EN là một phiên bản cải tiến của CAM và cách triển khai tương tự. Mục tiêu chính của CoMiC-EN không nhất thiết là hiệu quả hơn CAM, mà là chuyển sang một kiến trúc và bộ công cụ đủ linh hoạt để tích hợp trong các hệ thống dạy học thông minh. Đánh giá được thực hiện trên dữ liệu CAM (Bailey và Meurers, 2008), hiện được gọi là CREE (Corpus of Reading comprehension Exercises in English). Hai bài kiểm tra 2 lớp và 5 lớp được thực hiện với CAM cũng giống nhau, hiện được gọi là “phân loại nhị phân” và “phân loại chi tiết”.
* CoMiC-DE (9): CoMiC-DE (Meurers et al., 2011a) về cơ bản là phiên bản tiếng Đức của CoMiC-EN. Các bước tiền xử lý, lựa chọn đặc trưng và phân loại đều giống nhau, với những thay đổi cần thiết trong bộ công cụ để hỗ trợ ngôn ngữ Đức (ví dụ: sử dụng GermaNet (Hamp và Feldweg, 1997) thay vì WordNet). Bộ dữ liệu đánh giá được thay đổi từ CREE sang CREG (Corpus of Reading comprehension Exercises in German) (Meurers et al., 2010).
* Hou ’11 (19): Hou và Tsao (2011) triển khai một hệ thống cung cấp cho giáo viên chỉ báo tiến độ học tập của học sinh, nhưng có thể được mở rộng thành một hệ thống ASAG điển hình. Bốn loại đặc trưng được trích xuất, bao gồm nhãn từ loại (POS), tần suất thuật ngữ, tf.idf, và entropy, được kết hợp với bộ phân loại SVM. Các thí nghiệm cho thấy hệ thống phù hợp để cung cấp chỉ báo tiến độ, vì thang điểm 10 chỉ được khám phá sơ bộ. Đối với thí nghiệm 2 lớp, các nhóm điểm 0-5 và 6-10 được dự đoán, và đối với thí nghiệm 3 lớp, nhóm điểm trên được chia thành 6-7 và 8-10. Do đó, hồi quy có thể là một lựa chọn tốt để mở rộng công việc này cho ASAG.
* Horbach ’13 (18): Horbach et al. (2013) bao gồm các văn bản đọc hiểu từ các câu hỏi trong tập dữ liệu của họ như một ý tưởng chính. Đối với các câu hỏi ASAG khác, dữ liệu bao gồm ba thành phần: (1) câu hỏi, (2) câu trả lời của giáo viên, và (3) câu trả lời của học sinh. Tuy nhiên, trong các câu hỏi đọc hiểu, có thêm một thành phần: (4) văn bản đọc. Horbach et al. (2013) mô tả điều này là hữu ích vì câu trả lời của học sinh có thể chỉ đề cập đến một phần của văn bản đọc. Vì vậy, thông thường, các hệ thống ASAG khai thác mối quan hệ giữa (2) và (3), nhưng trong bài báo này, các cặp (2)/(4) và (3)/(4) cũng được khai thác. Phần lớn còn lại thực sự dựa trên CoMiC-DE (ở trên). Bộ dữ liệu CREG được sử dụng lại và phương pháp luận dựa trên sự liên kết (với các đặc trưng liên kết câu bao gồm sự đồng thuận đơn giản, entropy và lỗi liên kết trong số lượng câu). Một số công việc mới yêu cầu chú thích bổ sung trên bộ dữ liệu CREG để đánh dấu liên kết câu giữa (2) và (4) nhằm hỗ trợ trích xuất đặc trưng. Tuy nhiên, liên kết cũng được tự động hóa như một cách tiếp cận thay thế, và kết quả cho thấy hiệu quả về cơ bản tương đương hoặc tốt hơn một chút so với CoMiC-DE.
* Madnani ’13 (23): Madnani et al. (2013) triển khai một hệ thống chấm điểm các câu hỏi đọc hiểu về tiêu chuẩn sống. Mỗi văn bản có ba đoạn văn, và câu trả lời của học sinh yêu cầu một câu tóm tắt tổng quát và ba câu khác tóm tắt từng đoạn. Phương pháp học máy bao gồm tám đặc trưng (BLEU, ROUGE, các phép đo về các khía cạnh khác nhau của sao chép văn bản, số lượng câu, và số lượng từ liên kết diễn ngôn được sử dụng phổ biến) làm đầu vào cho bộ phân loại hồi quy logistic.

**Đánh giá (Evaluation)**: Khác với bốn giai đoạn đã được mô tả ở trên, giai đoạn đánh giá (evaluation) chủ yếu tập trung vào các kỹ thuật và phương pháp cụ thể. Đây không phải là một phương pháp đơn lẻ, mà là một cách tiếp cận để đánh giá các hệ thống, mô hình hoặc thuật toán khác nhau trong lĩnh vực chấm điểm tự động (ASAG) hoặc các lĩnh vực liên quan. Các cuộc thi và hội thảo được tổ chức để khuyến khích các nhóm nghiên cứu trên toàn thế giới cạnh tranh nhằm giải quyết các vấn đề cụ thể, từ đó thúc đẩy sự đổi mới và cải tiến trong lĩnh vực này.

## **1.3 Các khía cạnh chung trong hệ thống ASAG**

Các khía cạnh chung xuất hiện trong các hệ thống khác nhau liên quan đến việc phát triển ASAG. Sáu khía cạnh chính bao gồm:

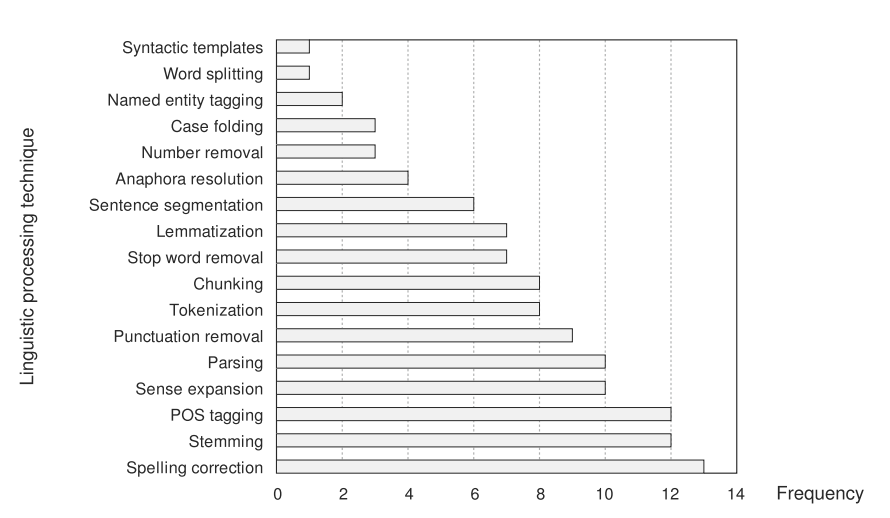
* Dữ liệu (Data Sets): Khía cạnh này tập trung vào các loại dữ liệu được sử dụng bởi các hệ thống khác nhau.
* Xử lý ngôn ngữ tự nhiên (NLP): Khía cạnh này tập trung vào các kỹ thuật NLP khác nhau được áp dụng trong các hệ thống để xử lý và hiểu các câu trả lời của học sinh. Điều này bao gồm các phương pháp phân đoạn văn bản, gắn nhãn phần của lời nói, phân tích phụ thuộc, và các đặc trưng ngôn ngữ khác đóng vai trò trong việc đánh giá chất lượng câu trả lời của học sinh.
* Xây dựng mô hình: Khía cạnh này tập trung vào các mô hình học máy hoặc mô hình thống kê được sử dụng trong các hệ thống ASAG. Nó bao gồm các thuật toán (ví dụ: hồi quy, cây quyết định, mạng nơ-ron) và các đặc trưng hoặc biến đầu vào được sử dụng để huấn luyện các mô hình này. Phần đánh giá sẽ làm nổi bật các chiến lược chung trong việc lựa chọn và huấn luyện mô hình.
* Mô hình chấm điểm: Khía cạnh này liên quan đến cách các mô hình được sử dụng để tạo ra điểm số cho các câu trả lời của học sinh. Nó bao gồm việc xem xét cách các hệ thống tiếp cận việc chấm điểm (ví dụ: phân loại nhị phân, phân loại đa lớp, hồi quy) và cách chúng kết hợp phán đoán của con người, nếu có.
* Đánh giá mô hình: Khía cạnh này xem xét cách hiệu suất của các hệ thống ASAG được đánh giá. Các chỉ số đánh giá phổ biến bao gồm độ chính xác, điểm F1, thống kê kappa và các biện pháp khác để đánh giá sự đồng thuận giữa các đánh giá dự đoán của hệ thống và các đánh giá từ người chấm. Khía cạnh này sẽ so sánh các phương pháp đánh giá khác nhau và hiệu quả của chúng.
* Hiệu quả: Khía cạnh này tập trung vào tác động thực tế của các hệ thống ASAG. Nó xem xét các hệ thống này hoạt động như thế nào trong các ứng dụng thực tế, chẳng hạn như trong lớp học, và thảo luận về các hạn chế hoặc các khu vực cần cải thiện. Điều này cũng bao gồm khả năng các hệ thống này tổng quát hóa tốt như thế nào đối với các loại câu hỏi, lĩnh vực, và ngôn ngữ khác nhau.

### **1.3.1 Tập dữ liệu (Data Sets)**

Về khía cạnh dữ liệu, có bốn thuộc tính định tính quan trọng: Nhóm đối tượng, cấp bậc học, ngôn ngữ và chủ đề.

### **1.3.2 Quá trình xử lý ngôn ngữ tự nhiên (Natural Language Processing)**

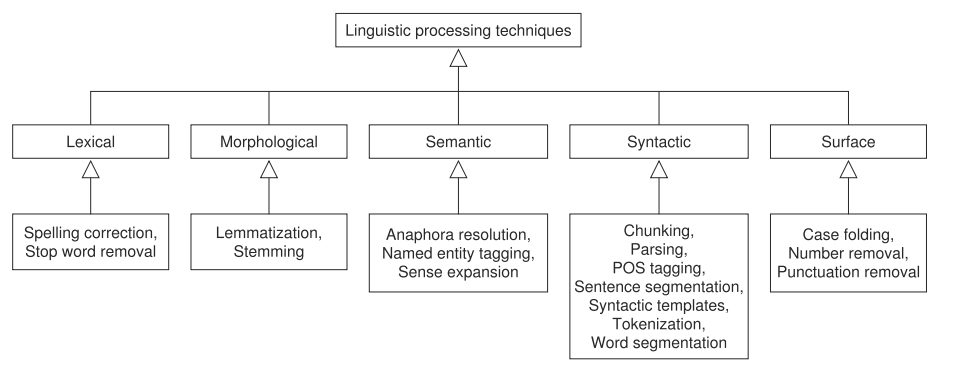
Các kỹ thuật Xử lý Ngôn ngữ Tự nhiên (NLP) đóng vai trò quan trọng trong việc phân tích ngôn ngữ trong các câu trả lời của sinh viên. Những kỹ thuật này có thể là các phương pháp xử lý ngôn ngữ học nhằm thao tác với văn bản, hoặc các kỹ thuật thống kê dựa trên các đặc trưng được trích xuất từ văn bản.



Hình 4: Hình vẽ mô tả các kĩ thuật xử lý ngôn ngữ

Có 17 kỹ thuật xử lý ngôn ngữ học khác nhau, như thể hiện trong Hình 4. Một ví dụ điển hình là phân đoạn từ (tokenization), trong đó văn bản được chia thành các từ. Các kỹ thuật khác bao gồm chuyển đổi chữ hoa thành chữ thường (case folding) và loại bỏ dấu câu. Những kỹ thuật xử lý ngôn ngữ học này thường được áp dụng rộng rãi trong thực tế hơn so với những gì được trình bày trong Hình 4. Trung bình, mỗi hệ thống sử dụng khoảng 3,3 kỹ thuật xử lý ngôn ngữ học.

Một trường hợp ít phổ biến là mẫu cú pháp (syntactic templates), trong đó các mẫu cú pháp (Szpektor và Dagan, 2007) được sử dụng để chuẩn hóa các biểu thức có nghĩa tương đương, như trong ví dụ về thỏ và rùa đã được đề cập trước đó. Một ví dụ khác là tách từ (word splitting), được sử dụng để phân đoạn dữ liệu tiếng Trung Quốc (Wang et al., 2008).



Hình 5: Hình vẽ mô tả các nhóm kĩ thuật xử lý ngôn ngữ

Tóm lại, nhiều kỹ thuật xử lý ngôn ngữ học có thể cần thiết, tùy thuộc vào giai đoạn cụ thể và các khía cạnh khác nhau của vấn đề. Các kỹ thuật này có thể được phân thành năm nhóm lớn: từ vựng (Lexical), hình thái học (Morphological), ngữ nghĩa (Semantic), cú pháp (Syntactic), và bề mặt (Surface), như thể hiện trong Hình 5.

Khác với các kỹ thuật xử lý ngôn ngữ học, kỹ thuật thống kê tạo ra các phép đo đơn lẻ hoặc đặc trưng (features), thường được áp dụng trong các hệ thống học máy. 15 hệ thống mà chúng ta xem xét thuộc kỷ nguyên học máy và tất cả các hệ thống học máy đều nằm trong giai đoạn đánh giá. Các kỹ thuật này bao gồm: tần suất từ, cosine, F1, dịch máy (ROUGE), tương đồng văn bản (edit distance), chồng chéo (greedy string tiling - Wise, 1993), chuỗi con chung dài nhất, sự chồng chéo ở đầu câu, sự suy luận, entropy, và sự hiện diện của các cụm từ cụ thể.

Các hệ thống CAM, CoMiC-EN, CoMiC-DE, và Horbach ’13 đều sử dụng các đặc trưng giống nhau hoặc rất tương tự. Các đặc trưng này nhằm đồng nhất hóa các cấp độ khác nhau và sử dụng các loại trừu tượng ngôn ngữ khác nhau (Meurers et al., 2011b) khi so sánh câu trả lời của giáo viên và sinh viên. Sự liên tục này đã giúp các nhà nghiên cứu tập trung vào những hướng đi khác, như hỗ trợ đa ngôn ngữ, cải tiến kiến trúc hệ thống, và tạo ra các tiêu chuẩn cho bộ dữ liệu.

Một ví dụ thú vị khác là hệ thống UKP-BIU trong cuộc thi SemEval ’13 Task 7, hệ thống này sử dụng một không gian đặc trưng rất rộng. Công nghệ tương đồng văn bản và suy luận tại đây rất quan trọng vì tính công khai và khả năng tái sử dụng của chúng.

### **1.3.3 Xây dựng mô hình (Build Model)**

Ở khía cạnh này, chúng ta sẽ xem xét kích thước bộ dữ liệu và cách thức tổ chức dữ liệu để xây dựng mô hình. Một mô hình được định nghĩa là bất kỳ đại diện nào của câu trả lời sinh viên, cho phép ánh xạ giữa câu trả lời của sinh viên và điểm số đúng với độ chính xác hợp lý. Bảng 2 cung cấp một tóm tắt định lượng về dữ liệu gốc và cách tổ chức dữ liệu cho việc xây dựng mô hình, bao gồm số lượng câu hỏi (Q), số lượng câu trả lời của giáo viên (TA), và số lượng câu trả lời của sinh viên (SA).

Bảng 2: Bảng mô tả kích thước dữ liệu được sử dụng để xây dựng mô hình

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Hệ thống | Tài liệu tham khảo | Q | SA | SA/Q |
| 1 | Atenea | Alfonseca and P´erez (2004) | 7 | 885 | 126 |
| 2 | ATM | Callear et al. (2001) | 9 | 886 | 98 |
| 3 | Auto-Assessor | Cutrone et al. (2011) | 10 | 924 | 92 |
| 4 | auto-marking | Sukkarieh et al. (2003) | 5 | 672 | 134 |
| 5 | AutoMark | Mitchell et al. (2002) | 3 | 266 | 89 |
| 6 | Burstein ’96 | Burstein et al. (1996) | 2 |  |  |
| 7 | c-rater | Leacock and Chodorow (2003) | 9 | 260 | 29 |
| 8 | CAM | Bailey and Meurers (2008) | 2 | 340 | 170 |
| 9 | CoMiC-DE | Meurers et al. (2011a) | 1 | 378 | 378 |
| 10 | CoMiC-EN | Meurers et al. (2011b) | 12 | 3150 | 263 |
| 11 | Conort ’12 | Conort (2012) | 22 | 1029 | 47 |
| 12 | CoSeC-DE | Hahn and Meurers (2012) | 2 | 2000 | 1000 |
| 13 | Dzikovska ’12 | Dzikovska et al. (2012) | 12 | 1640 | 137 |
| 14 | e-Examiner | G¨utl (2007) | 18 | 1370 | 76 |
| 15 | eMax | Sima et al. (2009) | 75 | 566 | 8 |
| 16 | ETS | Heilman and Madnani (2013) | 177 | 1032 | 6 |
| 17 | FreeText Author | Jordan and Mitchell (2009) | 75 | 566 | 8 |
| 18 | Horbach ’13 | Horbach et al. (2013) | 167 | 1032 | 6 |
| 19 | Hou ’11 | Hou and Tsao (2011) | 8 | 184 | 23 |
| 20 | IndusMarker | Siddiqi and Harrison (2008b) | 3 | 611 | 204 |
| 21 | Klein ’11 | Klein et al. (2011) | 7 | 1067 | 152 |
| 22 | Levy ’13 | Levy et al. (2013) | 177 | 1032 | 6 |
| 23 | Madnani ’13 | Madnani et al. (2013) | 9 | 342 | 38 |
| 24 | Mohler ’09 | Mohler and Mihalcea (2009) | 5 | 1396 | 29 |
| 25 | Nielsen ’08 | Nielsen et al. (2008b) | 87 | 19575 | 225 |
| 26 | PMatch | Jordan (2012a) | 7 | 282 | 40 |
| 27 | SAMText | Bukai et al. (2006) | 2 | 2695 | 1348 |
| 28 | SoftCardinality | Jimenez et al. (2013) | 80 | 630 | 90 |
| 29 | Tandella ’12 | Tandalla (2012 | 287 | 2273 | 28 |
| 30 | Thomas ’03 | Thomas (2003) | 11 | 95339 | 332 |
| 31 | UKP-BIU | Zesch et al. (2013) | 2 | 20114 | 1829 |
| 32 | Wang ’08 | Wang et al. (2008) | 4 | 129 | 65 |
| 33 | WebLAS | Bachman et al. (2002) | 10 | 2698 | 675 |
| 34 | Willow | P´erez-Mar´ın and Pascual-Nieto (2011) | 10 | 924 | 92 |
| 35 | Zbontar ’12 | Zbontar (2012) | 187 | 22950 | 2295 |
| 36 | ASAP ’12 SAS | Hewlett Foundation (2012) | 9 | 15923 | 87 |
| 37 | SemEval ’13 Task 7 | Dzikovska et al. (2013) | 70 | 1029 | 92 |

Dữ liệu trong Bảng 4 chỉ ra rằng số lượng câu trả lời của giáo viên thường không được báo cáo rõ ràng. Trong nhiều trường hợp, mặc dù câu trả lời của giáo viên có thể tồn tại, nhưng giả định thường là có một câu trả lời của giáo viên cho mỗi câu hỏi. Hơn nữa, các tác giả có thể công nhận câu trả lời của giáo viên theo cách không thể định lượng, chẳng hạn như gọi chung là “câu trả lời của giáo viên”. Thêm vào đó, việc mô tả câu trả lời của giáo viên cho các hệ thống ánh xạ khái niệm khá phức tạp, vì cần phải cung cấp thêm dữ liệu như số lượng khái niệm và số lượng câu trả lời của giáo viên cho mỗi khái niệm.

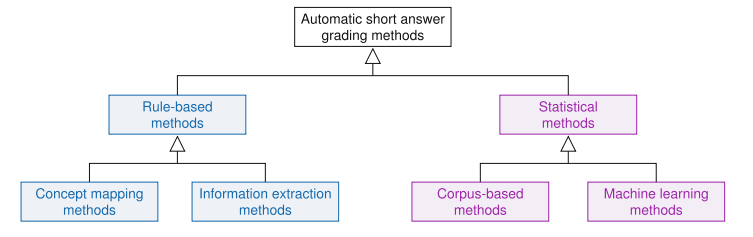
Cách thức dữ liệu được chia cho việc xây dựng mô hình và đánh giá có sự biến động lớn, như thể hiện trong Bảng 2. Lấy ví dụ hệ thống Atenea/Willow, ta thấy rằng việc ánh xạ từ câu trả lời của giáo viên và sinh viên đến dữ liệu mô hình và dữ liệu đánh giá là khá trực tiếp. Cụ thể, TA = M.Dat và SA = E.Dat. Trong trường hợp này, các câu trả lời của giáo viên và sinh viên được so sánh bằng các phương pháp như BLEU và LSA để tính toán điểm số.

Trong nhiều hệ thống khác, câu trả lời của giáo viên không được sử dụng trong quá trình huấn luyện mô hình, mà thay vào đó, chúng được dùng để hướng dẫn việc đánh dấu một số câu trả lời của sinh viên, được sử dụng làm dữ liệu huấn luyện, như trong hệ thống CoMiC-EN. Cách tiếp cận này cũng có thể được áp dụng trong thiết kế thử nghiệm cross-validation như five-fold cross validation (5FCV) trong hệ thống Hou ’11 và leave-one-out cross validation (LOOCV) trong hệ thống Horbach ’13.

Cuối cùng, cũng có thể kết hợp câu trả lời của giáo viên và sinh viên trong việc xây dựng mô hình. Ví dụ, trong hệ thống eMax, từ vựng của các câu trả lời giáo viên được mở rộng bằng cách xem xét từ vựng trong một tập hợp con nhỏ của các câu trả lời sinh viên chất lượng. Ngoài ra, cũng có thể mở rộng tập huấn luyện trong một thử nghiệm học máy bằng cách xem câu trả lời của giáo viên như các trường hợp bổ sung với điểm số hoàn hảo. Tuy nhiên, chúng ta chưa thấy trường hợp này được áp dụng rộng rãi.

### **1.3.4 Mô hình chấm điểm (Grading Model)**

Các mô hình chấm điểm được sử dụng phụ thuộc vào nhiều yếu tố khác nhau, đặc biệt là vào từng giai đoạn cụ thể. Để tổ chức và hình dung rõ hơn về điều này, chúng ta cần tóm gọn lại khái niệm "thời kỳ đánh giá", mà ở đây chúng ta dùng để chỉ sự hợp tác trong cộng đồng, thay vì chỉ tập trung vào công nghệ. Phần lớn các hệ thống hiện nay sử dụng phương pháp học máy, điều này tạo ra một tổ chức mới như thể hiện trong Hình 6.



Hình 6: Hình vẽ mô tả các cách thức khác nhau được sử dụng trong ASAG

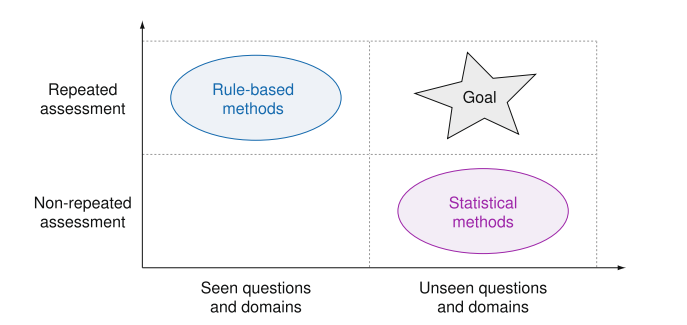
Tổ chức này giúp phân biệt rõ hơn giữa các mô hình chấm điểm "dựa trên quy tắc" và "dựa trên thống kê". Cụ thể, có sự tương đồng giữa các thời kỳ CMap/IE (dựa trên quy tắc) và CBM/ML (dựa trên thống kê). Đối với cặp CMap/IE, chúng ta nhận thấy bốn dòng đầu tiên trong CMap (Bảng 6) tương tự như IE, vì chúng dựa trên các phương pháp đối chiếu mẫu hoặc suy luận. Dòng thứ năm là một trường hợp đặc biệt, trong đó các khái niệm được cung cấp riêng biệt bởi người trả lời, vì vậy phương pháp trích xuất các khái niệm cá nhân từ một câu trả lời tổng hợp không còn cần thiết. Đối với cặp CBM/ML, các điểm số thường được tính toán bằng các phương pháp dựa trên tập hợp văn bản, vốn được xem là các tính năng riêng biệt. Ví dụ, các hệ thống Atenea và Willow sử dụng trung bình có trọng số của BLEU và LSA, cả hai đều có thể được coi là các tính năng học máy. Sau này, hệ thống Mohler '09 đã kết hợp các phương pháp dựa trên tri thức và tập hợp văn bản vào một hệ thống học máy (Mohler et al., 2011), và hệ thống UKP-BIU sử dụng các tính năng dựa trên tập hợp văn bản như một phần của bộ tính năng. Do đó, một cách nhìn khác có thể được hình dung như trong Hình 6.

Với bức tranh tổng thể này, chúng ta sẽ xem xét các sự đánh đổi giữa các phương pháp này. Giả thuyết của chúng ta (như thể hiện trong Hình 7) là phương pháp dựa trên quy tắc sẽ phù hợp hơn cho việc đánh giá lặp lại (khi các công cụ đánh giá được tái sử dụng), trong khi phương pháp thống kê lại phù hợp hơn cho các câu hỏi và lĩnh vực chưa được biết trước. Cụ thể, chúng ta có hai đặc điểm chính: "lặp lại" và "tổng quát hóa". Xem xét lại các tài liệu hỗ trợ cho giả thuyết, chúng ta cho rằng phương pháp dựa trên quy tắc thích hợp cho việc đánh giá lặp lại, vì có thể đầu tư vào các giải pháp cụ thể khi lợi ích có thể thu lại nhiều lần. Trong trường hợp này, các hệ thống thương mại như c-rater từ ETS có thể phát triển mạnh trong việc kiểm tra lặp lại. Ngược lại, các phương pháp thống kê đã phát triển mạnh trong cuộc thi SemEval ’13 Task 7, nơi yêu cầu các giải pháp cho các câu hỏi và lĩnh vực chưa được biết trước. Tình huống này đòi hỏi các giải pháp linh hoạt, vì các thí sinh chỉ có vài tháng để phát triển trên bộ dữ liệu của cuộc thi. Chúng ta cũng cho rằng công trình này đại diện cho việc đánh giá không lặp lại, như là lần đầu tiên và duy nhất có mặt trong cuộc thi ASAG tại SemEval ’13 vào thời điểm viết bài này.

Một điểm cần lưu ý là trong các đánh giá tổng kết quy mô lớn, đã trở nên phổ biến trong 10 năm qua (ví dụ: PISA12, được lặp lại ba năm một lần kể từ năm 2000), chúng ta có ít câu hỏi nhưng lại có một lượng lớn câu trả lời và các xếp hạng từ người đánh giá. Trong trường hợp này, các mô hình chính xác cao có thể được xây dựng bằng cách tạo các mẫu thủ công, như trong nhiều kỹ thuật dựa trên quy tắc. Tuy nhiên, trong các đánh giá hình thành, với sự phát triển của việc cá nhân hóa học tập và các khóa học trực tuyến mở rộng (MOOCs), chúng ta phải xử lý một lượng câu hỏi đa dạng và các câu trả lời, đồng thời đối mặt với nhiễu, chẳng hạn như các đánh giá từ bạn bè. Trong trường hợp này, các mô hình hiệu quả hơn có thể được tạo ra bằng các kỹ thuật thống kê.

Rõ ràng là trong ASAG, hiện nay có nhiều công trình nghiên cứu về các phương pháp thống kê hơn là các phương pháp dựa trên quy tắc. Điều này được thể hiện qua Hình 3 và thực tế là hầu hết các hệ thống từ thời kỳ đánh giá đều là các hệ thống thống kê. Tuy nhiên, câu hỏi cần được đặt ra là liệu các phương pháp này có đủ mạnh để đạt được mục tiêu trong Hình 7 hay không. Chúng ta đề xuất rằng câu hỏi này có thể được trả lời thông qua việc hợp tác nhiều hơn với các đối tác thương mại trong tương lai và các đánh giá lặp lại mà họ yêu cầu.

### **1.3.5 Đánh giá mô hình (Model Evaluation)**



Hình 7: Hình vẽ so sánh giữa phương pháp thống kê và dựa trên quy tắc

Về việc đánh giá**:** Chế độ chấm điểm (grading scheme) rất quan trọng vì loại chỉ số đánh giá được áp dụng phụ thuộc vào chế độ này. Do đó, việc cung cấp dữ liệu này là cần thiết để xác minh tính chính xác của quy trình đánh giá. Như thể hiện trong Bảng 7, dữ liệu này đôi khi không được cung cấp. Khi có dữ liệu, số lượng các hạng mục hoặc điểm số thường bị giới hạn ở mức 5. Một ngoại lệ là hệ thống e-Examiner, sử dụng thang điểm 10 cho 8 câu hỏi trong lĩnh vực khoa học máy tính. Thang điểm này được định nghĩa một cách lỏng lẻo từ 0 (không phù hợp) đến 10 (rất tốt). Tuy nhiên, việc sử dụng thang điểm 10 có thể gây khó khăn trong việc đạt được mức độ đồng thuận cao giữa các giám khảo và tạo ra sự phức tạp không cần thiết đối với các hệ thống chấm điểm đã được xác định rõ trong ASAG. Một ngoại lệ khác là hệ thống Wang ’08, nơi các câu hỏi có thể có đến 30 điểm, nhưng phương pháp chấm điểm này đặc biệt ở chỗ các khái niệm được xử lý theo cách cộng dồn, và người trả lời được thưởng cho việc liệt kê càng nhiều khái niệm càng tốt.

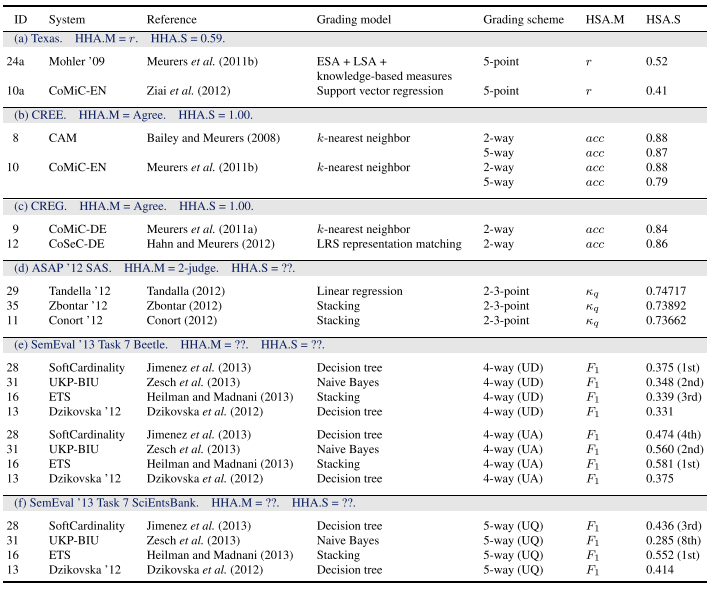
Cấp độ đo lường trong ASAG: Ở cấp độ tổng quát hơn, trong ASAG, chúng ta có dữ liệu từ ba trong bốn cấp độ đo lường nổi tiếng từ hệ thống phân loại của Stevens (1946): dữ liệu danh mục (nominal), dữ liệu thứ tự (ordinal) và dữ liệu tỷ lệ (ratio), nhưng không có dữ liệu khoảng (interval). Dữ liệu danh mục liên quan đến các loại phân loại riêng biệt, nơi không thể áp dụng hiệu ứng xếp hạng. Ví dụ là một phần trong hệ thống chấm điểm năm mức của SemEval ’13 Task 7, nơi không thể xác định liệu câu trả lời "mâu thuẫn" có tốt hơn hay tệ hơn câu trả lời "không liên quan". Dữ liệu thứ tự đề cập đến các phân loại có hiệu ứng xếp hạng, ví dụ là hệ thống chấm điểm hai mức của SemEval ’13 Task 7, với hai loại "đúng" và "sai", trong đó "đúng" rõ ràng tốt hơn "sai". Dữ liệu tỷ lệ liên quan đến các phép đo liên tục có nguồn gốc bằng không, ví dụ như hệ thống e-Examiner (Gütl, 2007), nơi cho phép điểm số phân đoạn cho các câu hỏi 10 điểm. Trong ASAG, tỷ lệ thang đo thường được điều chỉnh để giảm số lượng điểm số có thể được trao cho một bộ dữ liệu rời rạc, ví dụ là hệ thống chấm điểm rời rạc của Madnani ’13, nơi điểm số chỉ được phép thuộc tập hợp chính xác {0, 1, 2, 3, 4}. Cuối cùng, thể loại dữ liệu khoảng không được đề cập trong bài viết này, liên quan đến dữ liệu tỷ lệ nhưng không có nguồn gốc bằng không. Các ví dụ trong thế giới thực là nhiệt độ, ngày tháng, và tọa độ địa lý, có nguồn gốc tùy ý.

Chọn lựa chỉ số đánh giá: Cần lưu ý rằng sự lựa chọn chỉ số đánh giá đôi khi có thể áp dụng cho hai loại dữ liệu cùng lúc. Ví dụ, thang đo thứ tự hai mức "sai" và "đúng" có thể được hiểu như thang đo tỷ lệ {0, 1}. Tương tự, dữ liệu tỷ lệ có thể được coi là dữ liệu rời rạc hoặc liên tục tùy thuộc vào độ chi tiết. Vì vậy, trong một số trường hợp, việc chuyển đổi giữa các loại dữ liệu có thể không gây hại và có thể cho phép so sánh thêm.

Đánh giá các thước đo từ Bảng 7 (theo phân loại của Stevens, 1946):

* Dữ liệu danh mục và thứ tự: Các thước đo này được áp dụng cho cả dữ liệu danh mục và thứ tự. Một thước đo phổ biến là độ chính xác ("accuracy") hoặc sự đồng thuận ("agreement"). Trong tài liệu nghiên cứu, độ chính xác và sự đồng thuận thường được sử dụng như những thuật ngữ tương tự. Tuy nhiên, để tránh sự nhầm lẫn, có thể sử dụng "accuracy" (acc) cho HSA và "agreement" (agr) cho HHA. Điều quan trọng là phải làm rõ chỉ số được sử dụng và cung cấp tham chiếu nếu có thể.
* Kappa của Cohen (Cohen’s κ): Đây là thước đo sự đồng thuận đã được hiệu chỉnh theo xác suất ngẫu nhiên. Kappa là thước đo không trọng số, có nghĩa là tất cả các sự không đồng nhất đều được xử lý một cách ngang nhau (ký hiệu κu cho kappa không trọng số). Thước đo này thích hợp cho dữ liệu danh mục và thứ tự, nơi mức độ khác biệt không thể định lượng được.
* F1 và Precision: Hai chỉ số này được áp dụng trong hệ thống SemEval ’13 Task 7 và Hou ’11. F1 được chọn cho SemEval ’13 Task 7 vì nó phù hợp với mọi sự kết hợp của nhãn lớp tích cực và tiêu cực trong các kịch bản của cuộc thi. Precision, như được ghi nhận trong nghiên cứu của Hou và Tsao (2011), được sử dụng rộng rãi để đánh giá các hệ thống trong lĩnh vực NLP.
* Dữ liệu tỷ lệ (Rời rạc): Đối với dữ liệu tỷ lệ rời rạc, thước đo Cohen’s κ có trọng số được áp dụng (ký hiệu κl hoặc κq cho các trọng số tuyến tính và bậc hai). Thước đo này phù hợp vì có thể áp dụng các mức độ trừng phạt khác nhau đối với các sự không đồng nhất. Ví dụ, sự không đồng nhất 2 điểm nên bị phạt nặng hơn sự không đồng nhất 1 điểm trên thang điểm nhiều mức.
* Dữ liệu tỷ lệ (Liên tục): Đối với dữ liệu tỷ lệ liên tục, hệ số tương quan Pearson (Pearson’s r) là chỉ số duy nhất được sử dụng. Một số ví dụ khác có thể là hệ số tương quan hạng Spearman hoặc Kendall.

### **1.3.6 Hiệu quả (Effectiveness)**



Hình 8: Hình vẽ mô tả ví dụ minh hoạ về hiệu quả của các hệ thống ASAG

Cuối cùng, về các điểm hiệu quả trong Hình 7, các so sánh có ý nghĩa giữa các hệ thống là hạn chế, vì phần lớn các đánh giá được thực hiện trong môi trường khép kín (bubble), nghĩa là số lượng bộ dữ liệu chung giữa hai hoặc nhiều ấn phẩm là khá ít. Điều này có nghĩa là nhiều điểm hiệu quả trong Hình 7 chủ yếu mang tính thông tin. Sau đây, chúng ta sẽ làm rõ một số trường hợp khác trong Hình 8 để thảo luận chi tiết hơn.

Đầu tiên, trong Hình 8a, chúng ta đề cập đến bộ dữ liệu “Texas,” một bộ dữ liệu chưa được nhấn mạnh cho đến nay. Bộ dữ liệu này có sẵn công khai và rất giống với bộ dữ liệu được sử dụng trong ấn phẩm thứ hai của Mohler ’09 (Mohler et al., 2011). Chính Ziai et al. (2012) đã sử dụng bộ dữ liệu này để so sánh hiệu quả của hai hệ thống hiện có: phiên bản mới nhất của Mohler ’09 (Meurers et al., 2011b) và một phiên bản dựa trên hồi quy, CoMiC-EN (Ziai et al., 2012). Do đó, ấn phẩm này rất thú vị vì tác giả không đề xuất một hệ thống mới mà thay vào đó thực hiện một so sánh thực nghiệm giữa các hệ thống hiện có. Chúng ta không tìm thấy bài báo nào tương tự như vậy trong bài đánh giá tài liệu. Kết quả cho thấy phương pháp Mohler ’09 vượt trội hơn phương pháp CoMiC-EN với một khoảng cách rõ rệt (r = 0.52 so với r = 0.41). Hiệu quả của Mohler ’09 cũng khá hợp lý vì nó gần đạt tới giới hạn HHA.S (r = 0.59).

Tiếp theo Hình 8b và Hình 8c đưa ra các so sánh cho bộ dữ liệu CREE và CREG từ dự án CoMiC. Khi xét đến CREE, hệ thống CoMiC-EN, kế thừa của CAM, không hiệu quả hơn, nhưng điều này có thể được giải thích bởi thực tế rằng hệ thống CoMiC-EN tập trung nhiều hơn vào kiến trúc. Khi xét đến CREG, CoSeCDE cung cấp một cải thiện nhỏ so với CoMiC-DE. Trong tất cả các trường hợp, độ chính xác là hợp lý và gần đạt giới hạn HHA.S (agr = 1.00).

Một số kết quả của cuộc thi SAS ASAP ’12 tiếp theo trong Bảng 10d. Hệ thống Tandella ’12 (Tandalla 2012), hệ thống Zbontar ’12 (Zbontar 2012), và hệ thống Conort ’12 (Conort 2012) đã đạt vị trí từ thứ 2 đến thứ 4 trong bảng xếp hạng ban đầu. Các điểm κq tương ứng của chúng rất gần nhau, và hệ thống của đội dẫn đầu chỉ hơn có +0.00077 điểm. Bảng xếp hạng đầy đủ có thể được lấy từ trang web của cuộc thi.

Nhìn chung, thông tin về các hệ thống SAS ASAP ’12 là hạn chế vì mục đích của các bài nộp chiến thắng là cung cấp thông tin cho các đối tác thương mại. Vì vậy, ví dụ, các nhà tổ chức cuộc thi chỉ phát hành 5 bài báo phương pháp luận, ít so với 51 đội đã đạt điểm đánh giá tích cực. Thêm vào đó, có 20 đội đạt được κq ≥ 0.7, một điểm số được coi là xuất sắc theo (Fleiss, 2003). So với đó, chỉ có tám đội tham gia cuộc thi SemEval ’13 Task 7, vì vậy SAS ASAP ’12 đại diện cho sự tham gia lớn nhất trong bất kỳ sáng kiến ASAG nào.

Một vấn đề khác với SAS ASAP ’12 là bộ dữ liệu không thể chia sẻ ngoài bối cảnh của cuộc thi, điều này giới hạn tính hữu ích của nó trong cộng đồng học thuật. Tóm lại, kết quả của cuộc thi SAS ASAP ’12 có phần hạn chế đối với cộng đồng ASAG rộng lớn hơn do hạn chế về bộ dữ liệu và số lượng bài báo phương pháp luận ít ỏi.

Cuối cùng, kết quả F1 trung bình theo vĩ mô từ SemEval ’13 Task 7 được trình bày trong Hình 8e và 8f. Nhìn chung, bộ kết quả đánh giá đầy đủ khó có thể tổng hợp do tính đa dạng: nhiều bộ dữ liệu (Beetle và SciEntsBank), nhiều nhiệm vụ (câu trả lời chưa thấy, câu hỏi chưa thấy, lĩnh vực chưa thấy), nhiều sơ đồ chấm điểm (2 chiều, 3 chiều và 5 chiều), và nhiều chỉ số đánh giá (độ chính xác, F1 trung bình vi mô và F1 trung bình vĩ mô). Như đã mô tả khi đánh giá các hệ thống này, phương pháp tiếp cận của chúng ta trong việc ưu tiên tài liệu này là tập trung vào các chiều của cuộc thi khó khăn và mới mẻ nhất. Điều này đã giúp chúng ta có được bảng xếp hạng trong nửa trên của Bảng 10e dựa trên bài tổng quan (Dzikovska et al., 2013). Tuy nhiên, những bảng xếp hạng này không duy trì được khi so sánh với các bảng xếp hạng mẫu khác từ Hình 8e, 8f và nói chung. Thực tế, ngay cả baseline của Dzikovska ’12 (Dzikovska et al., 2012) cũng đã là người chiến thắng cho một trong các nhiệm vụ con.

Mặt tích cực là các so sánh cho SemEval ’13 Task 7 là duy nhất có kiểm tra ý nghĩa thống kê, trái ngược với phần còn lại của Hình 8. Kiểm tra cụ thể được sử dụng là kiểm tra ngẫu nhiên xấp xỉ với 10.000 lần lặp và ngưỡng p ≤ 0.05 (Dzikovska et al., 2013). Trong việc phân tích dữ liệu này, nhiều giải pháp hiệu quả có thể được xác định cho mỗi nhiệm vụ con, bao gồm các hệ thống có điểm cao nhất cho mỗi nhiệm vụ con và các hệ thống có hiệu quả không khác biệt về mặt thống kê so với hệ thống dẫn đầu.

Ba hệ thống mà chúng ta đã đánh giá hoạt động tốt tổng thể từ góc nhìn này, như hệ thống CoMeT (Ott et al., 2013), một hệ thống khác từ gia đình dự án CoMiC. Tuy nhiên, nhìn chung, chúng ta thiếu một chỉ số duy nhất để chỉ ra phương pháp tiếp cận có hiệu quả tổng thể nhất. Thực tế là điều này có thể thực hiện được, vì rất đơn giản để cộng lại số lượng phân loại chính xác trong tất cả các nhiệm vụ con. Việc báo cáo một chỉ số như vậy sẽ chỉ ra hệ thống có hiệu quả cao nhất trong một loạt các tình huống. Đây là một vấn đề nhỏ nhưng quan trọng cho công việc tương lai.

Một mục còn thiếu khác cho SemEval ’13 Task 7 là một chỉ số HHA. Rất nhiều công việc đã được thực hiện trong việc tạo ra và vận hành cuộc thi này, vì vậy rất thú vị để biết liệu các hệ thống nộp có đang tiến gần đến hiệu quả HHA hay không. Các nỗ lực chú thích ban đầu thực tế có các giá trị κu (Dzikovska et al., 2012), nhưng chúng không có giá trị vì các sơ đồ chấm điểm đã được chỉnh sửa cho cuộc thi. Cụ thể, nỗ lực chú thích Beetle ban đầu được thực hiện với 11 danh mục, và nỗ lực chú thích SciEntsBank ban đầu được thực hiện với 4 danh mục ở cấp độ ánh xạ facet. Cả hai sau đó đã được tái ánh xạ về sơ đồ 5 chiều cho cuộc thi mà không có công việc mới để đo lường lại HHA.

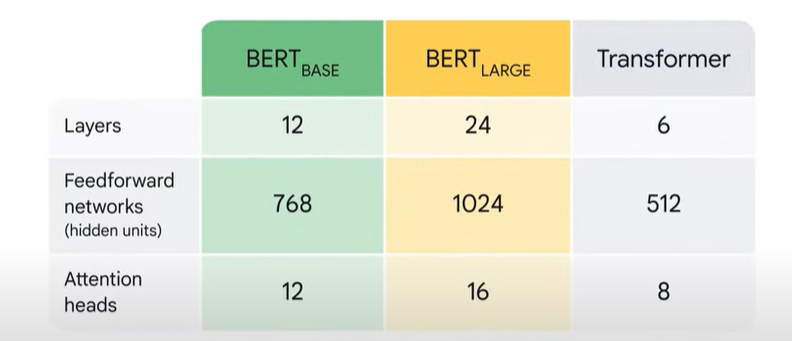
Khi xem xét hiệu quả tổng thể thêm nữa, dữ liệu trong Hình 8 không chỉ ra nhiều sự giao thoa hữu ích giữa các Hình 8a đến f, ngoài việc đánh giá chung giữa Beetle và SciEntsBank. Một ví dụ điển hình là công trình gần đây của Ziai et al. (2012), cung cấp một liên kết giữa Hình 8a và b. Một ví dụ khác là sự tham gia của ETS trong cả ASAP ’12 SAS và SemEval ’13 Task 7, ngoại trừ việc ETS chọn không công bố phương pháp luận của họ tại ASAP ’12 SAS.

# **CHƯƠNG 2:** **ỨNG DỤNG BERT TRONG ASAG**

## **2.1 Giới thiệu về BERT**

BERT (Bidirectional Encoder Representations from Transformers) là một mô hình ngôn ngữ (Language Model) được phát triển bởi Google AI. BERT được xem là một bước đột phá lớn trong lĩnh vực Machine Learning nhờ khả năng ứng dụng rộng rãi vào nhiều bài toán NLP (Natural Language Processing) khác nhau như Question Answering, Natural Language Inference, và nhiều ứng dụng khác với kết quả xuất sắc. BERT là một mô hình học máy dựa trên kiến trúc transformer dành cho xử lý ngôn ngữ tự nhiên, được phát triển và đào tạo bởi Google. Dữ liệu dùng cho quá trình pre-training của BERT bao gồm các quyển sách với 800 triệu từ và Wikipedia với 2.500 triệu từ. BERT có hai phiên bản chính:

* **BERT Base**: 12 layers (transformer blocks), 12 attention heads và 110 triệu tham số.
* **BERT Large**: 24 layers (transformer blocks), 16 attention heads và 340 triệu tham số.

.

Hình 9: Hình vẽ mô tả các loại BERT

BERT được pre-trained qua hai nhiệm vụ chính là Masked Language Modeling (MLM) và Next Sentence Prediction (NSP):

* Masked Language Modeling (MLM): BERT được thiết kế như một mô hình hai chiều, có khả năng lấy thông tin từ cả hai chiều ngữ cảnh—từ trái sang phải và từ phải sang trái.
* Next Sentence Prediction (NSP): BERT cũng được đào tạo để hiểu mối quan hệ giữa các câu, giúp mô hình hiểu được sự liên kết giữa các câu trong văn bản.

Kết quả của quá trình pre-training giúp BERT học và hiểu ngữ cảnh của các từ. Sau khi hoàn thành giai đoạn pre-training, BERT có thể được tinh chỉnh (fine-tuned) với các tập dữ liệu cụ thể để tối ưu hóa hiệu suất cho các nhiệm vụ riêng biệt.

Do đó, hệ thống ASAG có khả năng học và hiểu ngữ cảnh của câu trả lời và đáp án của học sinh cho một câu hỏi cụ thể, từ đó so sánh sự tương đồng về mặt ngữ nghĩa giữa chúng để đưa ra điểm số trong một phạm vi xác định.

## **2.2 Lý do sử dụng BERT**

Một trong những thách thức lớn nhất trong NLP là vấn đề dữ liệu. Mặc dù trên internet có vô vàn dữ liệu, nhưng chúng lại không đồng nhất và mỗi phần dữ liệu thường chỉ được sử dụng cho một mục đích riêng biệt. Khi giải quyết một bài toán cụ thể, ta cần phải trích xuất một bộ dữ liệu phù hợp, và kết quả là ta thường chỉ có một lượng dữ liệu hạn chế. Điều này tạo ra một nghịch lý: các mô hình Deep Learning yêu cầu một lượng dữ liệu rất lớn—thậm chí lên tới hàng triệu dữ liệu—để có thể đạt được hiệu quả tốt. Vậy làm sao để tận dụng được nguồn dữ liệu khổng lồ sẵn có trên internet để giải quyết bài toán của mình?

Đây chính là lý do cho sự ra đời của kỹ thuật Transfer Learning. Với Transfer Learning, các mô hình "chung" được đào tạo với tập dữ liệu lớn có sẵn (pre-training) có thể được "tinh chỉnh" (fine-tune) cho các bài toán cụ thể. Nhờ vào kỹ thuật này, kết quả của nhiều bài toán đã được cải thiện đáng kể, không chỉ trong NLP mà còn trong các lĩnh vực khác như Computer Vision. BERT là một trong những đại diện xuất sắc của Transfer Learning trong NLP. Nó không chỉ gây ấn tượng mạnh bởi kết quả vượt trội trong nhiều bài toán khác nhau, mà còn vì nó hoàn toàn miễn phí, giúp tất cả chúng ta có thể sử dụng BERT cho các bài toán của mình.

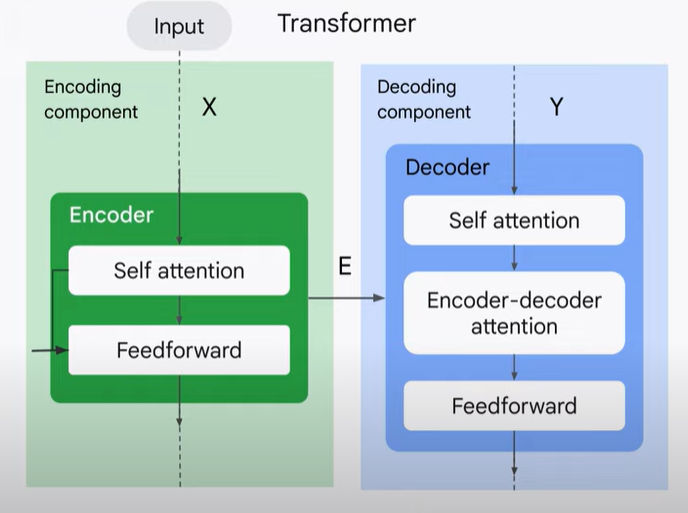
## **2.3 Nền tảng của BERT**

BERT sử dụng Transformer, một mô hình attention (cơ chế chú ý) học mối quan hệ giữa các từ (hoặc một phần của từ) trong văn bản. Transformer gồm hai phần chính: Encoder và Decoder. Tuy nhiên, BERT chỉ sử dụng phần Encoder.

Transformer là một kiến trúc dựa trên khái niệm attention, một kỹ thuật được sử dụng để cung cấp trọng số cho các phần khác nhau của chuỗi đầu vào nhằm đạt được sự hiểu biết tốt hơn về ngữ cảnh của nó. Bên cạnh đó, Transformer xử lý đầu vào một cách song song, giúp mô hình trở nên hiệu quả và dễ dàng mở rộng hơn so với các mô hình tuần tự truyền thống như RNN và LSTM.

Trước khi có Transformer, các mô hình học máy có thể biểu diễn từ dưới dạng vector, nhưng những vector này không mang tính ngữ cảnh. Ví dụ, từ "bank" trong hai ngữ cảnh "money bank" và "river bank" có thể có cùng một vector đại diện, trước khi cơ chế attention được áp dụng. Tuy nhiên, Transformer sử dụng cơ chế attention để hiểu mối quan hệ giữa các từ trong câu, giúp phân biệt rõ ràng các nghĩa của từ trong ngữ cảnh khác nhau.

Transformer có thể xử lý một lượng dữ liệu lớn cùng lúc nhờ vào kiến trúc của nó. Sau khi embedding các từ thành các vector số, các vector này sẽ được xử lý qua hai lớp của Encoder.



Hình 10: Hình vẽ mô tả kiến trúc Transformer

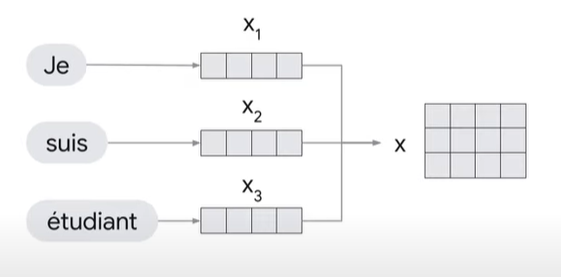
Self attention: là quá trình so sánh các từ trong câu với các từ khác trong chính câu đó để tìm ra mối quan hệ giữa các từ trong câu.

Feedforward: là một mạng nơ-ron với hàng triệu tham số nhận đầu vào input từ self attention layer và trả ra output là các vector đại diện (representations) cho các từ trong câu.

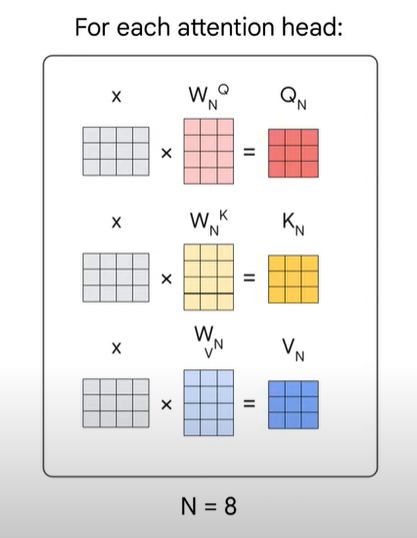
Quá trình xử lý của Encoder:

B1: Đầu vào là những câu dưới dạng ngôn ngữ tự nhiên. Ví dụ: Je suis étudiant

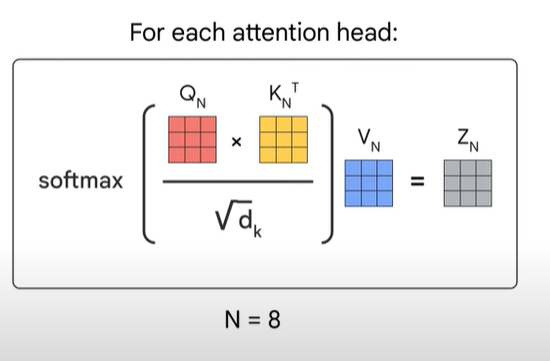
B2: Embedding mỗi từ thành vector.



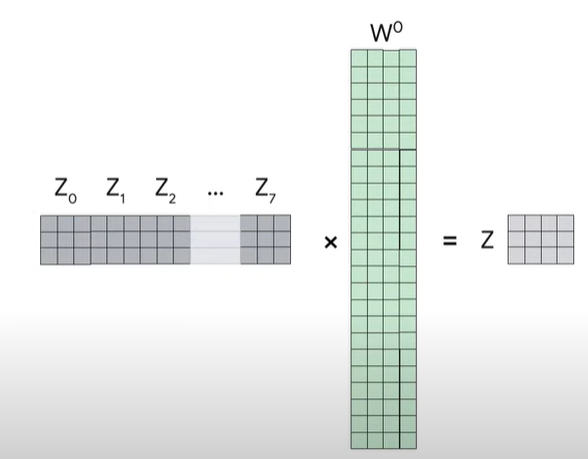
B3: Thực hiện multi-headed attention: nhân với các ma trân trọng số (QKV)



B4: Tính toán attention sử dụng kết quả từ 3 ma trận QKV



B5: Nối các ma trận lại với nhau để tạo thành một ma trận cuối cùng có cùng kích thước.



Tóm lại, attention giúp di chuyển các từ trong câu đến gần các từ liên quan để hiểu ngữ cảnh tốt hơn. Cụ thể, mỗi từ trong câu được chuyển đổi thành một vector số trong không gian đa chiều. Trong không gian này, các từ có nghĩa tương tự sẽ có vector gần nhau, trong khi các từ khác nhau sẽ có vector cách xa nhau.

Ví dụ:

"Money in the bank" và "The bank of the river". Từ "bank" trong *"Money in the bank"* được hiểu là "ngân hàng", trong khi từ "bank" trong *"The bank of the river"* lại có nghĩa là "bờ sông".

Trước khi có attention: Khi mã hóa các từ trong câu thành vector, từ "bank" có thể được mã hóa một cách trung bình hoặc chung chung. Vì vậy, trong không gian vector, "bank" có thể ở một vị trí không phân biệt giữa "ngân hàng" và "bờ sông".

Sau khi áp dụng attention:

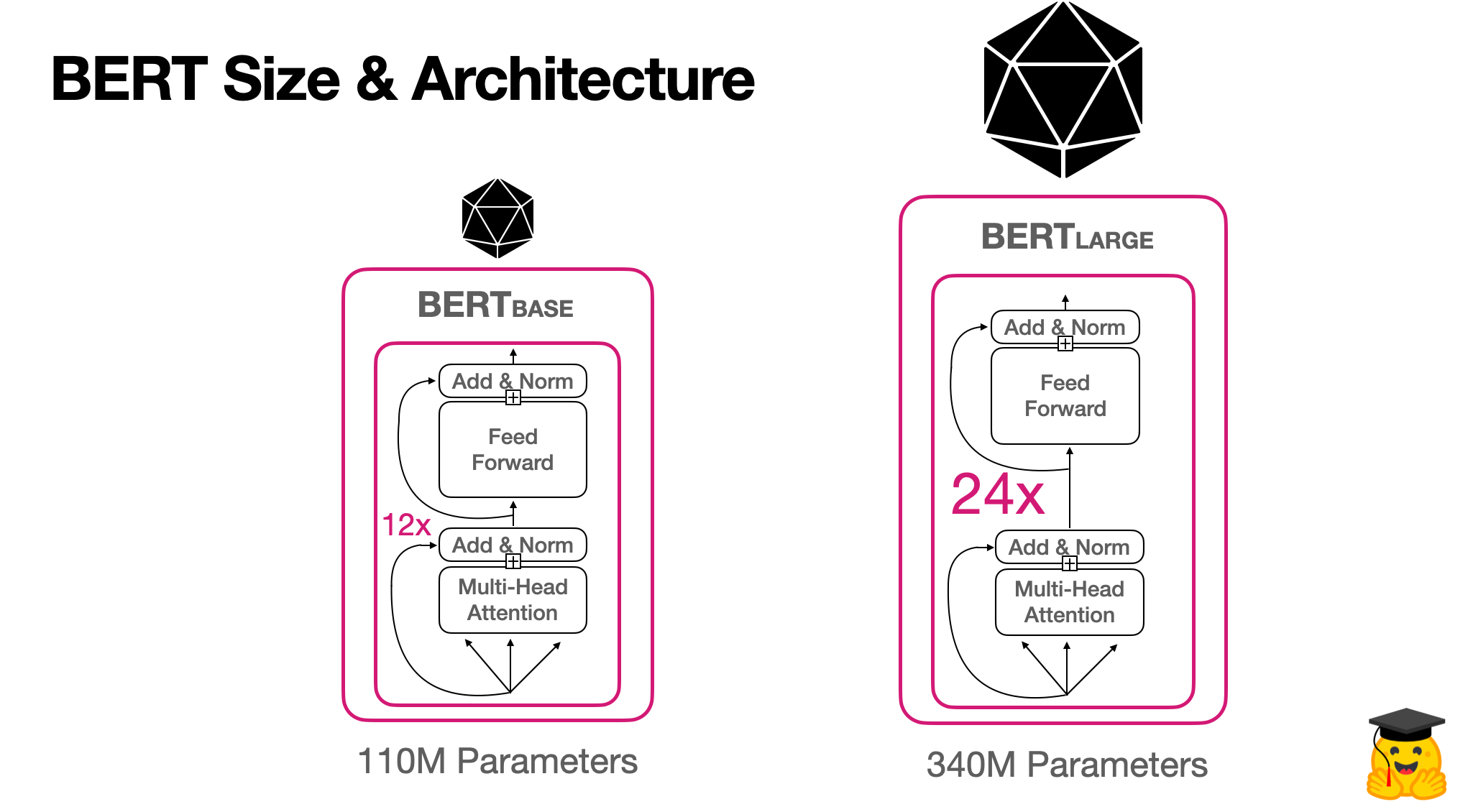
* Trong câu *"Money in the bank"*, từ "bank" sẽ được "đưa gần" hơn với các từ như "money", vì trong ngữ cảnh này, "bank" có nghĩa là ngân hàng.
* Trong câu *"The bank of the river"*, từ "bank" sẽ được "đưa gần" hơn với các từ như "river", "shore", vì trong ngữ cảnh này, "bank" có nghĩa là bờ sông.

## **2.4 Kiến trúc BERT**

BERT được xây dựng dựa tên thành phần Encoder của Transformer. Kiến trúc của BERT gồm nhiều Encoder và tuỳ vào loại BERT khác nhau mà có số lượng Encoder khác nhau.

**Thuật ngữ (ML Architecture Glossary):**

|  |  |
| --- | --- |
| Phần kiến trúc | Định nghĩa |
| Parameters (Tham số) | Số lượng các biến/giá trị có thể học được của mô hình. |
| Transformer Layers (Lớp Transformer) | Số lượng các khối Transformer. Một khối Transformer chuyển đổi một chuỗi các đại diện từ ngữ thành một chuỗi các từ ngữ đã được ngữ cảnh hóa (các đại diện được đánh số). |
| Hidden Size (Kích thước ẩn) | Các lớp của các hàm toán học, nằm giữa đầu vào và đầu ra, có nhiệm vụ gán trọng số (cho các từ) để tạo ra kết quả mong muốn. |
| Attention Heads (Đầu chú ý) | Kích thước của một khối Transformer. |
| Processing (Xử lý) | Loại đơn vị xử lý được sử dụng để huấn luyện mô hình. |
| Length of Training (Thời gian huấn luyện) | Thời gian đã sử dụng để huấn luyện mô hình. |



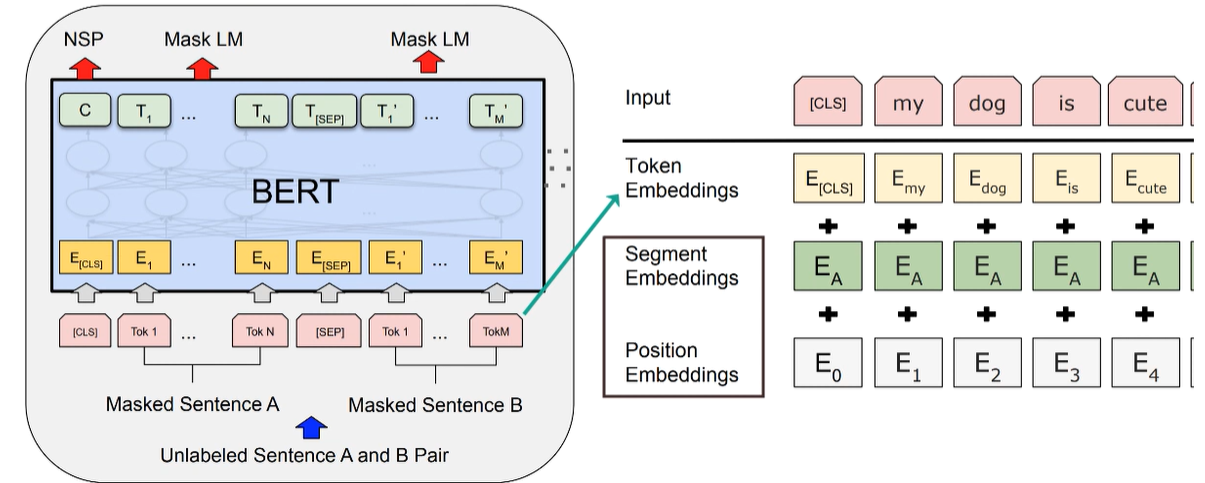
Hình 11: Hình vẽ mô tả kiến trúc của BERT

Dưới đây mô tả chi cấu hình của BERT:

Bảng 3: Bảng mô tả chi tiết kiến trúc của BERT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Transformer Layers | Hidden Size | Attention Heads | Parameters | Processing | Length of Training |
| BERTbase | 12 | 768 | 12 | 110M | 4 TPUs | 4 days |
| BERTlarge | 24 | 1024 | 16 | 340M | 16 TPUs | 4 days |

## **2.5 Các giai đoạn đào tạo BERT**



Hình 12: Hình vẽ mô tả hai nhiệm vụ của quá trình đào tạo BERT

*Pretrain BERT*: Mục đích của giai đoạn này là để BERT hiểu ngôn ngữ, hiểu ngữ cảnh của bài toán cụ thể. Trong hệ thống ASAG thì giai đoạn này giúp BERT hiểu ngữ cảnh câu hỏi câu trả lời. Ở giai đoạn thì có 2 nhiệm vụ chính đó là:

* Masked LM (MLM): Trước khi đưa vào BERT, thì 15% số từ trong chuỗi được thay thế bởi token [MASK], khi đó mô hình sẽ dự đoán từ được thay thế bởi [MASK] với context là các từ không bị thay thế bởi [MASK]. (Trên thực tế, con số 15% không phải là cố định mà có thể thay đổi theo mục đích của bài toán, tuy nhiên nếu cao quá sẽ làm mất ngữ cảnh của câu)
* Next Sentence Prediction (NSP): Trong chiến lược này, thì mô hình sử dụng một cặp câu là dữ liệu đầu vào và dự đoán câu thứ 2 là câu tiếp theo của câu thứ 1 hay không. Trong quá trình huấn luyện, 50% lượng dữ liệu đầu vào là cặp câu trong đó câu thứ 2 thực sự là câu tiếp theo của câu thứ 1, 50% còn lại thì câu thứ 2 được chọn ngẫu nhiên từ tập dữ liệu.

*Fine tune BERT*: Mục đích giai đoạn này là để BERT học một nhiệm vụ cụ thể. Đối với hệ thống ASAG thì nhiệm vụ chính là chấm điểm cho các câu trả lời ngắn. Tùy vào bài toán mà ta có các phương pháp fine-tune khác nhau:

* Đối với bài toán Classification, ta thêm vào một Classification Layer với input là output của Transformer cho token [CLS].
* Đối với bài toán Question Answering, model nhận dữ liệu input là đoạn văn bản cùng câu hỏi và được huấn luyện để đánh nhãn cho câu trả lời trong đoạn văn bản đó.
* Đối với bài toán Named Entity Recognition (NER), model được huấn luyện để dự đoán nhãn cho mỗi token (tên người, tổ chức, địa danh...).

## **2.6 Sử dụng BERT trong NLP**

NLP tasks:

- Sentiment analysis: phân tích đánh giá nhà hàng tích cực hay tiêu cực

- Text classification: phân loại văn bản vào các mục khác nhau như việc phân loại các đánh giá nào thuộc bộ phim ‘Tình cảm’, đánh giá nào thuộc bộ phim ‘Hành động’

- Next word prediction: (There is a beautiful \_\_\_\_\_) BERT có thể dự đoạn từ tiếp theo dựa trên ngữ cảnh trước đó

- Question – Answer: trả lời câu hỏi dựa trên một vài ngữ cảnh

### **2.6.1 So sánh BERT với các mô hình khác**

BERT là một trong những mô hình hiểu ngữ cảnh rất tốt khi được so sánh với các mô hình khác.

Ví dụ:

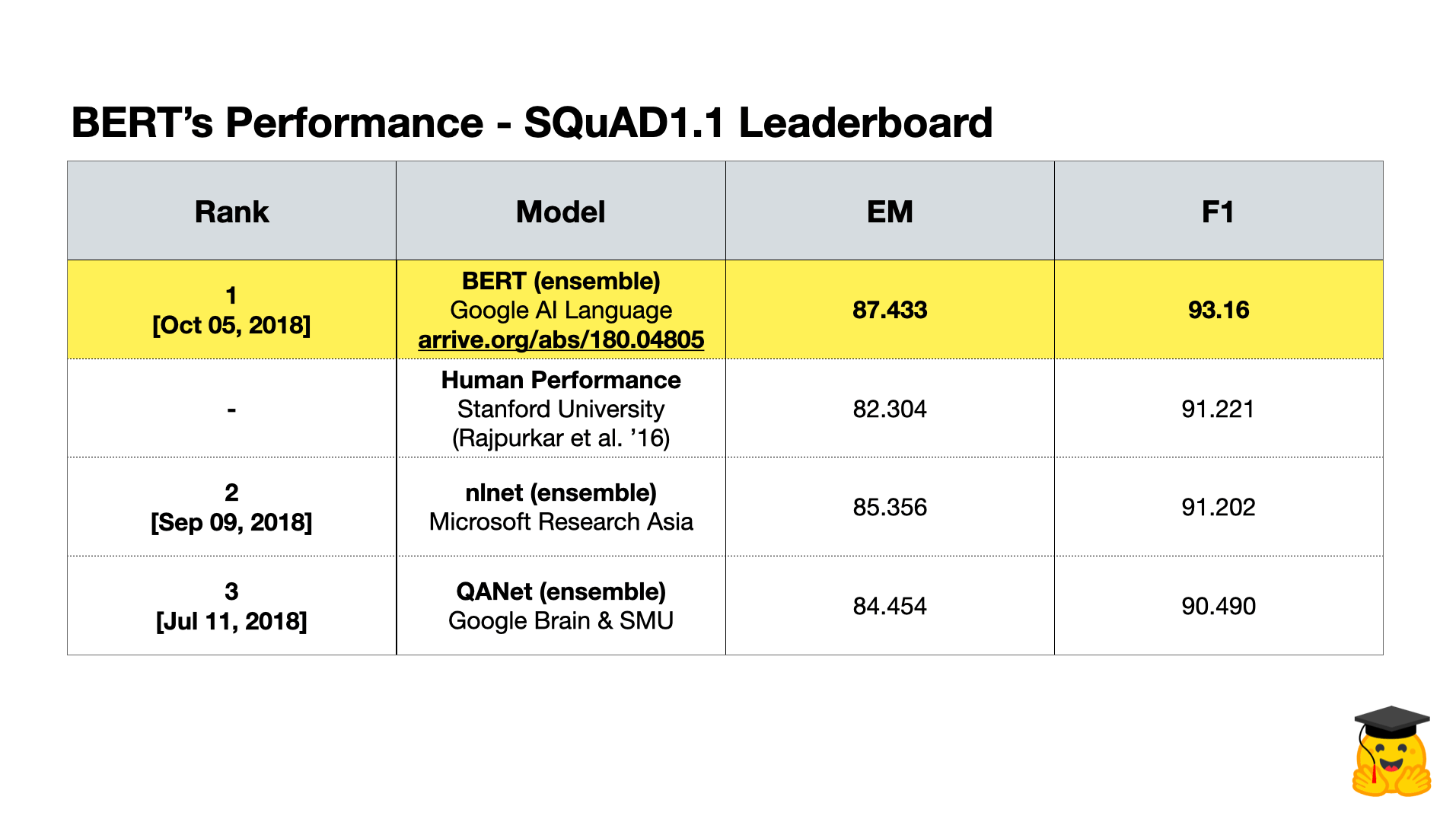
- Con chuột ăn hết lúa trong tải

- Con chuột không dây cần phải dùng pin để chạy

Trong ví dụ trên, con chuột trong 2 câu được hiểu theo 2 cách khác nhau hay còn gọi là có ngữ cảnh khác nhau. Nếu chúng ta sử dụng các mô hình khác như Word2Vec (context-free embedding) thì mô hình này sẽ coi ‘con chuột’ trong 2 câu trên là như nhau. Ngược lại, BERT (context-based embedding) có thể hiểu ngữ cảnh trong hai câu trên.

BERT đã đạt được thành công độ chính xác tiên tiến trên 11 nhiệm vụ NLP phổ biến, vượt trội so với các mô hình NLP hàng đầu trước đó và là mô hình đầu tiên vượt trội hơn con người. Tuy nhiên, những thành tựu này được đo lường như thế nào?

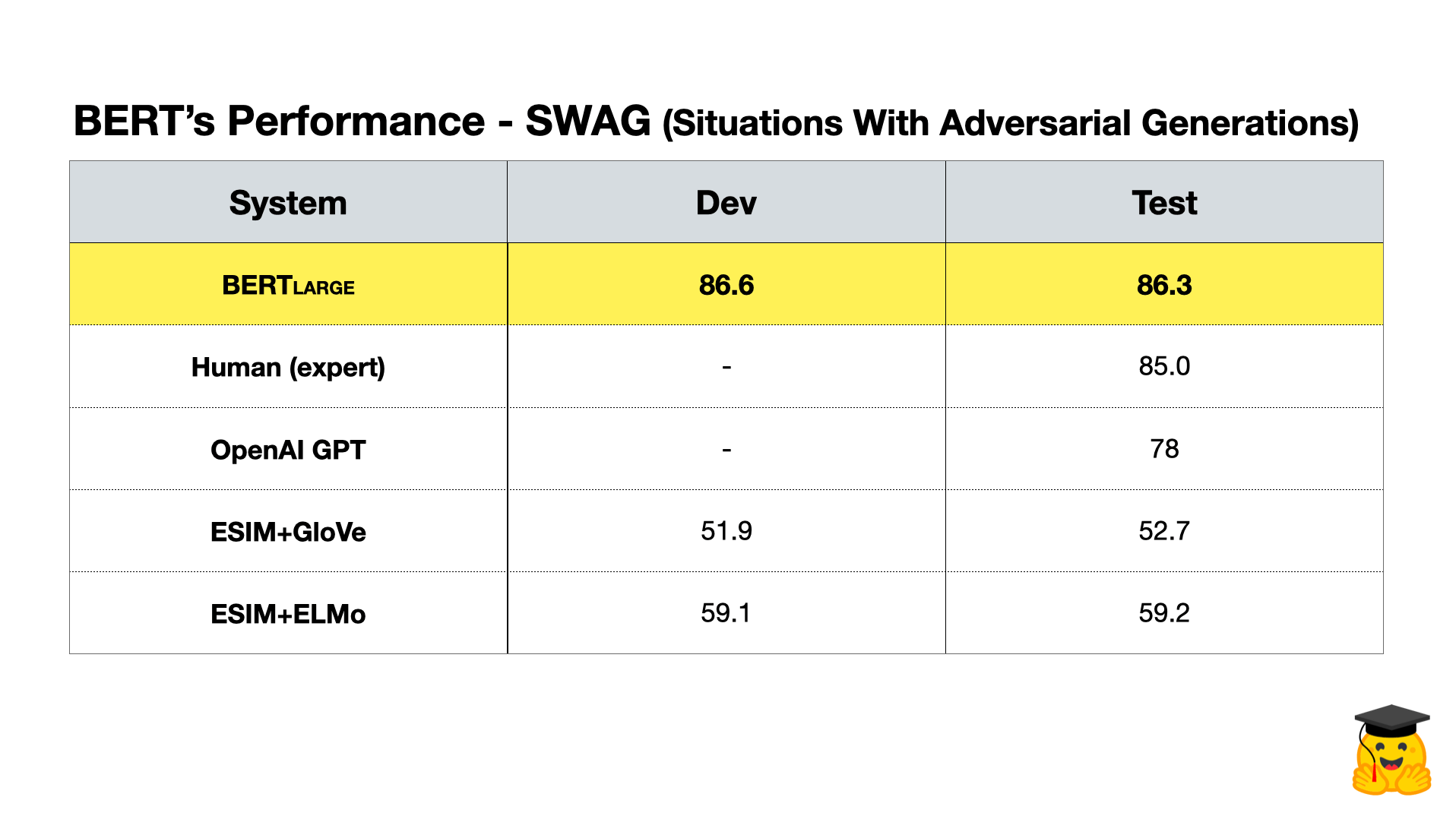
SQuAD (Stanford Question Answering Dataset) là bộ dữ liệu đọc hiểu gồm khoảng 108 nghìn câu hỏi có thể được trả lời thông qua một đoạn văn bản Wikipedia tương ứng. Hiệu suất của BERT theo phương pháp đánh giá này là một thành tựu lớn khi đánh bại các mô hình tiên tiến nhất và hiệu suất ở cấp độ con người trước đó:



Hình 13: Hình vẽ so sánh hiệu suất của BERT so với các mô hình khác đối với tập dữ liệu SQuAD

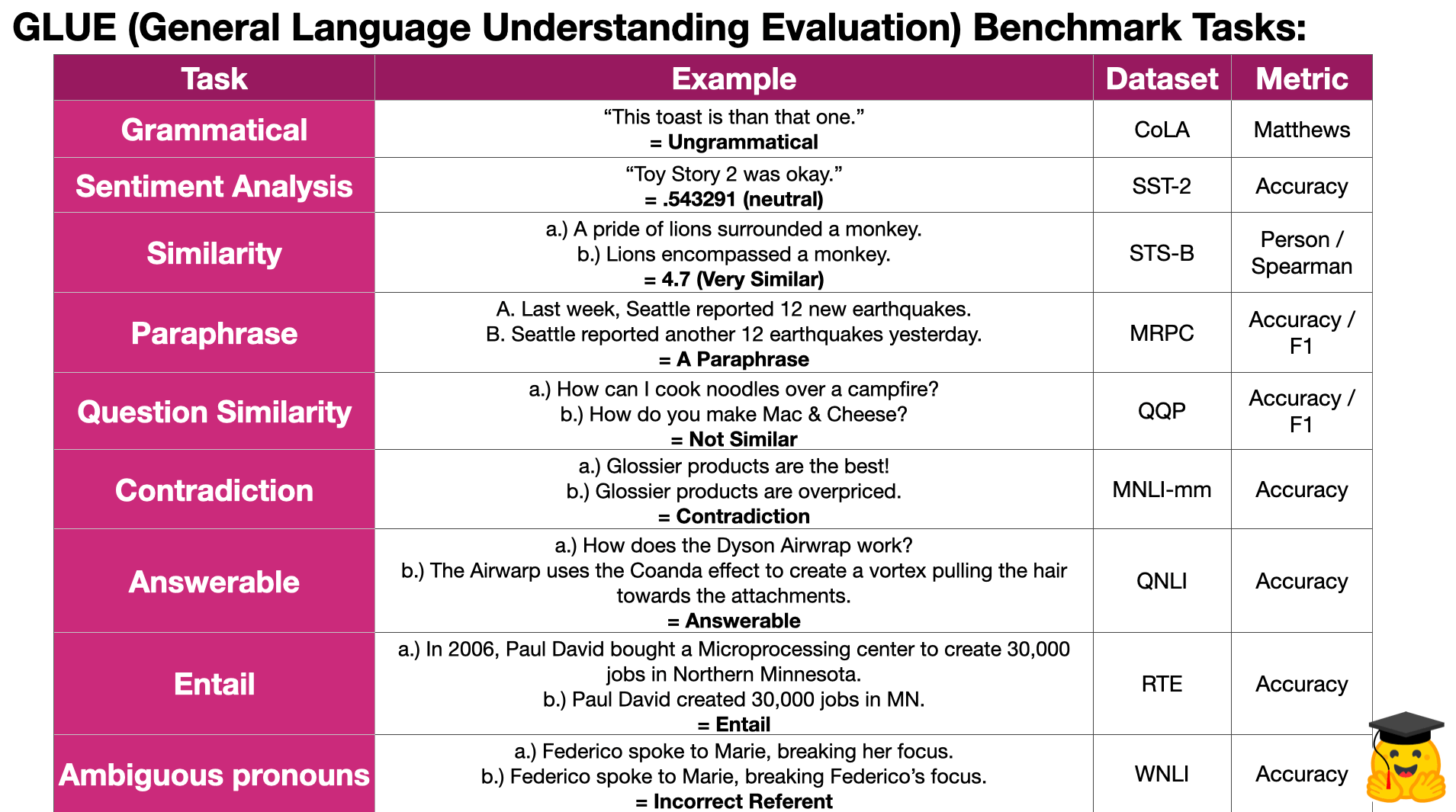
SWAG (Situations With Adversarial Generations) là một đánh giá rất hay ở chỗ nó phát hiện khả năng suy luận của một mô hình. Nó thực hiện điều này thông qua một tập dữ liệu quy mô lớn gồm 113 nghìn câu hỏi trắc nghiệm về các tình huống thông thường. Những câu hỏi này được chép lại từ một cảnh/tình huống video và SWAG cung cấp cho mô hình bốn kết quả có thể xảy ra trong cảnh tiếp theo. Sau đó, mô hình sẽ dự đoán câu trả lời đúng.

BERT vượt trội so với các mô hình hàng đầu trước đó, bao gồm cả hiệu suất ở cấp độ con người:

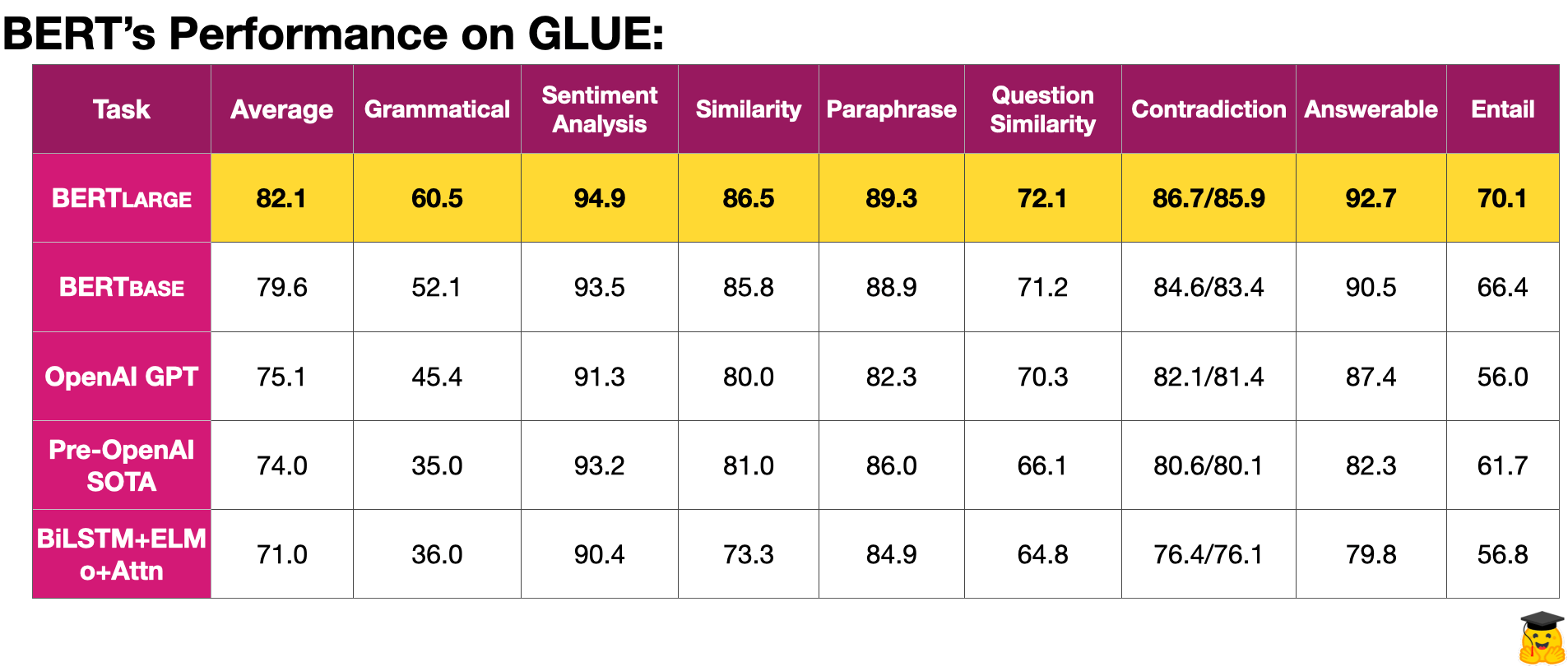


Hình 14: Hình vẽ mô tả hiệu suất của BERT so với các mô hình khác đối với tập dữ liệu SWAG

GLUE (General Language Understanding Evaluation) là một nhóm tài nguyên để đào tạo, đo lường và phân tích các mô hình ngôn ngữ so sánh với nhau. Các tài nguyên này bao gồm chín nhiệm vụ khó được thiết kế để kiểm tra sự hiểu biết của mô hình NLP. Dưới đây là bản tóm tắt của từng nhiệm vụ đó:



Hình 15: Hình vẽ mô tả các nhiệm vụ để kiểm tra sự hiểu biết của mô hình NLP



Hình 16: Hình vẽ mô tả hiệu suất của BERT với các mô hình khác dối với tập dữ liệu GLUE

### **2.6.2 Cách sử dụng BERT**

Đầu tiên chúng ta hãy cài đặt Transformers bằng đoạn code sau:

pip install transformers

Sử dụng BERT để dự đoán từ còn thiếu. Sử dụng token [MASK] thay cho từ còn thiếu:

from transformers import pipeline

unmasker = pipeline('fill-mask', model='bert-base-uncased')

unmasker("Artificial Intelligence [MASK] take over the world.")

Output:

[{'score': 0.3182411789894104,

'sequence': 'artificial intelligence can take over the world.',

'token': 2064,

'token\_str': 'can'},

{'score': 0.18299679458141327,

'sequence': 'artificial intelligence will take over the world.',

'token': 2097,

'token\_str': 'will'},

{'score': 0.05600147321820259,

'sequence': 'artificial intelligence to take over the world.',

'token': 2000,

'token\_str': 'to'},

{'score': 0.04519503191113472,

'sequence': 'artificial intelligences take over the world.',

'token': 2015,

'token\_str': '##s'},

{'score': 0.045153118669986725,

'sequence': 'artificial intelligence would take over the world.',

'token': 2052,

'token\_str': 'would'}]

*Ví dụ khác:*

unmasker("The man worked as a [MASK].")

Output:

[{'score': 0.09747546911239624,

'sequence': 'the man worked as a carpenter.',

'token': 10533,

'token\_str': 'carpenter'},

{'score': 0.052383411675691605,

'sequence': 'the man worked as a waiter.',

'token': 15610,

'token\_str': 'waiter'},

{'score': 0.04962698742747307,

'sequence': 'the man worked as a barber.',

'token': 13362,

'token\_str': 'barber'},

{'score': 0.037886083126068115,

'sequence': 'the man worked as a mechanic.',

'token': 15893,

'token\_str': 'mechanic'},

{'score': 0.037680838257074356,

'sequence': 'the man worked as a salesman.',

'token': 18968,

'token\_str': 'salesman'}]

BERT là một trong những mô hình đầu tiên về NLP được đào tạo theo cách hai bước:

* BERT đã được đào tạo về số lượng lớn dữ liệu chưa được gắn nhãn (không có chú thích của con người) theo kiểu không được giám sát.
* BERT sau đó được đào tạo trên một lượng nhỏ dữ liệu do con người chú thích bắt đầu từ mô hình được đào tạo trước đó, mang lại hiệu suất tiên tiến.

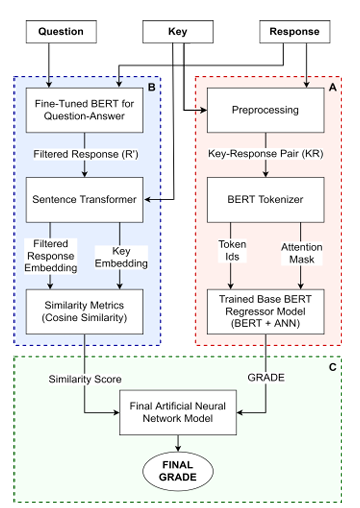
Tóm lại, BERT là một mô hình ngôn ngữ rất phức tạp và tiên tiến giúp mọi người tự động hóa việc hiểu ngôn ngữ. Khả năng đạt được hiệu suất hiện đại của nó được hỗ trợ bằng cách đào tạo về lượng dữ liệu khổng lồ và tận dụng kiến ​​​​trúc Transformers để cách mạng hóa lĩnh vực NLP.

## **2.7 BERT trong ASAG**

Loại câu hỏi yêu cầu câu trả lời ngắn (short answer) thường yêu cầu một vài thông tin bổ sung cùng câu hỏi để chấm điểm. Câu trả lời của học sinh sẽ được chấm điểm dựa trên ngữ cảnh. Cách truyền thống là chấm điểm thủ công, tuy nhiên khi tăng số lượng học sinh thì khối lượng công việc của giáo viên tang rất nhiều. Đôi khi việc chấm điểm thủ công cũng đưa ra một vài sự không nhất quán hoặc thiên vị do nhiều lý do khác nhau. Vì vậy giải pháp cho vấn đề này là chấm điểm tự động. Automatic Short Answer Grading (ASAG) có thể được sử dụng trong trường hợp này. Hệ thống ASAG sẽ tìm hiểu ý nghĩa ngữ cảnh câu trả lời của học sinh với đáp án đối với một câu hỏi cụ thể. Vì vậy có thể xem đây là một nhiệm vụ xử lý ngôn ngữ tự nhiên (Natural Language Processing Task)

Bảng 4: Bảng so sánh các phương pháp được sử dụng với các hệ thống ASAG đã tồn tại trước đó

|  |  |  |
| --- | --- | --- |
| **Models** | **Pearson’s R** | **RMSE** |
| Final Proposed Model (BERT Regressor + Similarity Score) | 0.777 | 0.732 |
| Base Model (BERT Regressor) | 0.760 | 0.753 |
| Tulu et al (2021) [10] | 0.949 | 0.040 |
| Sultan et al. (2016) [7] | 0.630 | 0.850 |
| Tf-Idf (2016) [7] | 0.320 | 1.020 |
| Ramachandran et al. (2015) [6] | 0.610 | 0.860 |
| Mohler et al. (2011) [2] | 0.518 | 0.978 |



Hình 17: Hình vẽ mô tả các bước sử dụng BERT trong hệ thống ASAG

Mục đích của hệ thống ASAG là chịu trách nhiệm tạo ra điểm số khi được cung cấp câu hỏi (Question - Q), đáp án (Key - K), câu trả lời của học sinh (Response - R)

(Question, Key, Response) -> Grade

Phương pháp được sử dụng trong ASAG là phương pháp kết hợp giữa mô hình Regressor BERT và mô hình Fine-tuned Question-Answer BERT

* Bước 1: Mô hình BERT Regressor được tinh chỉnh sử dụng một tập dữ liệu dataset ASAG cụ thể.
* Bước 2: Liên quan đến việc xác định sự tương đồng giữa đáp án (K) và câu trả lời của học sinh (R). Mô hình Question-Answer BERT
* Bước 3: Kết hợp điểm tương đồng ở Bước 2 cùng với điểm được dự đoán của mô hình BERT Regressor để cung cấp điểm cuối cùng cho câu trả lời của học sinh với một câu hỏi cụ thể.

### **2.7.1 Mô hình BERT Regressor cho hệ thống ASAG**

Mô hình BERT Base kết hợp với mô hình hồi quy dựa trên mạng nơ-ron nhân tạo (ANN) cho nhiệm vụ ASAG. Mô hình BERT Base được sử dụng để tạo embedding, với mỗi chuỗi đầu vào (sequence input) được chuyển đổi thành một embedding vector có 768 chiều. Mô hình hồi quy bao gồm một lớp đầu vào, hai lớp ẩn, một lớp đầu ra và các lớp dropout để ngăn ngừa overfitting. Hàm kích hoạt ReLU được sử dụng trong các lớp ẩn. Mô hình BERT Regressor sẽ được đào tạo với một tập dữ liệu đặc thù để dự đoán điểm số trong một phạm vi cụ thể.

Ban đầu, mỗi cặp câu hỏi-đáp án (key-response) sẽ được chuyển thành một chuỗi token thông qua BERT tokenizer. Chuỗi token bắt đầu với token [CLS], và các câu trong chuỗi sẽ được ngăn cách bằng token [SEP]. Những chuỗi token này sẽ được đưa vào mô hình BERT Regressor trong quá trình đào tạo. Nhiệm vụ của mô hình là dự đoán điểm số trong một khoảng giá trị cụ thể. Mô hình BERT sẽ tạo ra embedding cho mỗi cặp key-response, và embedding này sẽ là đầu vào cho mô hình hồi quy để dự đoán điểm cho cặp key-response đó.

**Ví dụ về Cặp K-R và Token IDs**:

* **Câu hỏi (Q)**: Vai trò của chương trình nguyên mẫu trong giải quyết vấn đề là gì?
* **Đáp án (K)**: Mô phỏng hành vi của các phần của sản phẩm phần mềm mong muốn.
* **Câu trả lời của học sinh (R)**: Được định nghĩa trong giai đoạn Đặc tả, một nguyên mẫu kích thích hành vi của các phần của sản phẩm phần mềm mong muốn. Nghĩa là, vai trò của một nguyên mẫu là một giải pháp tạm thời cho đến khi chương trình được hoàn thiện để sử dụng rộng rãi trong giải quyết vấn đề.
* **Cặp K-R**: [CLS] Mô phỏng hành vi của các phần của sản phẩm phần mềm mong muốn [SEP] Được định nghĩa trong giai đoạn Đặc tả, một nguyên mẫu kích thích hành vi của các phần của sản phẩm phần mềm mong muốn. Nghĩa là, vai trò của một nguyên mẫu là một giải pháp tạm thời cho đến khi chương trình được hoàn thiện để sử dụng rộng rãi trong giải quyết vấn đề [SEP]
* **Token IDs của K-R** (không padding): [101, 104…]

### **2.7.2 Xác định độ tương đồng giữa Key-Response**

Sự tương đồng giữa đáp án (K) và câu trả lời của học sinh (R) là yếu tố quan trọng trong việc chấm điểm. Độ tương đồng càng cao, điểm số càng lớn. Tuy nhiên, việc xác định sự tương đồng giữa K và R gặp nhiều khó khăn. Ví dụ, từ khóa trong đáp án có thể xuất hiện trong câu trả lời của học sinh nhưng lại không đúng ngữ cảnh. Ngoài ra, câu trả lời của học sinh có thể dài hơn đáp án nhưng vẫn đúng về mặt ngữ nghĩa. Trong trường hợp này, việc cắt ngắn câu trả lời để phù hợp với đáp án có thể làm mất đi những thông tin quan trọng, dẫn đến sai lệch trong đánh giá.

Để giải quyết vấn đề này, thay vì đơn giản cắt ngắn câu trả lời, một "câu trả lời đã lọc" (filtered response) sẽ được xây dựng từ câu trả lời ban đầu của học sinh. Câu trả lời đã lọc này sẽ chỉ giữ lại những phần quan trọng nhất và loại bỏ các phần không cần thiết hoặc không liên quan, giúp làm rõ thông tin để so sánh với đáp án. Câu trả lời đã lọc sẽ được sử dụng để tính toán độ tương đồng với đáp án (key), giúp nâng cao độ chính xác trong quá trình chấm điểm.

Để xây dựng câu trả lời đã lọc, mô hình BERT tinh chỉnh cho nhiệm vụ Question-Answering sẽ được sử dụng. Mô hình này sẽ tạo ra giá trị bắt đầu và giá trị kết thúc cho mỗi từ trong câu trả lời. Từ có giá trị bắt đầu lớn nhất sẽ được chọn làm từ bắt đầu, và từ có giá trị kết thúc lớn nhất sẽ là từ kết thúc. Tuy nhiên, phương pháp này có thể tạo ra một câu trả lời đã lọc rất ngắn, đôi khi thiếu thông tin quan trọng, khiến mô hình đánh giá không chính xác. Vì vậy, giải pháp cho vấn đề này là đảm bảo chiều dài của câu trả lời đã lọc không nhỏ hơn chiều dài của đáp án, nhằm tránh mất đi thông tin quan trọng.

Câu trả lời đã lọc (filtered response) chỉ chứa các thông tin quan trọng từ câu trả lời của học sinh, đảm bảo chất lượng đánh giá được ưu tiên hơn số lượng. Cả câu trả lời đã lọc và đáp án sẽ được sử dụng để tính toán độ tương đồng (similarity score).

Similarity Score (A, B) =

Tóm lại, phương pháp này được sử dụng để tạo ra điểm tương đồng giữa các thông tin quan trọng trong câu trả lời của học sinh (R) và đáp án (K), giúp nâng cao độ chính xác trong quá trình chấm điểm.

### **2.7.3 Tích hợp Similarity Score với BERT Regressor**

Similarity Score được tích hợp với BERT Regressor thông qua mạng nơ-ron nhân tạo. Mạng này sẽ được đào tạo để dự đoán điểm cuối cùng dựa trên sự kết hợp giữa điểm tương đồng và các đặc trưng khác của dữ liệu.

# **CHƯƠNG 3: SỬ DỤNG CHATGPT TRONG ASAG**

## **3.1 Giới thiệu ChatGPT**

ChatGPT là một công cụ mạnh mẽ được phát triển bởi OpenAI, dựa trên mô hình ngôn ngữ GPT (Generative Pre-trained Transformer) – một trong những mô hình ngôn ngữ lớn hiện nay. Được huấn luyện trên một lượng dữ liệu văn bản khổng lồ từ internet, ChatGPT có khả năng hiểu và tạo ra văn bản giống như con người. Mô hình này có thể trả lời câu hỏi, viết bài luận, sáng tạo nội dung, dịch thuật, và thậm chí tham gia vào các cuộc trò chuyện dài. Cốt lõi của ChatGPT là khả năng xử lý ngôn ngữ tự nhiên (NLP), một lĩnh vực của trí tuệ nhân tạo (AI) chuyên nghiên cứu việc máy tính hiểu và tương tác với ngôn ngữ con người.

Phiên bản GPT-4 mà chúng ta đang đề cập đến trong nghiên cứu này có quy mô ấn tượng với hàng trăm tỷ tham số, cho phép mô hình xử lý các tác vụ phức tạp từ trả lời câu hỏi đơn giản đến sáng tạo nội dung phức tạp. Bên cạnh đó, GPT-4 cũng có khả năng xử lý nhiều ngôn ngữ, trong đó có tiếng Việt – ngôn ngữ chính được sử dụng trong bộ dữ liệu của đồ án này.

Điều khiến ChatGPT nổi bật so với các công cụ AI khác chính là khả năng "học" từ các cuộc trò chuyện và phản hồi linh hoạt theo ngữ cảnh. Mô hình có thể điều chỉnh cách thức trả lời dựa trên ngữ cảnh câu hỏi, mục đích của người sử dụng, và các yếu tố môi trường khác. Tuy nhiên, mặc dù rất mạnh mẽ trong việc tạo ra văn bản tự nhiên, ChatGPT cũng có một số hạn chế. Một trong những yếu tố quan trọng cần lưu ý là ChatGPT không có khả năng hiểu biết "thực tế" như con người. Mô hình này không thể tự mình trải nghiệm thế giới mà chỉ hoạt động dựa trên các mẫu dữ liệu đã được huấn luyện, điều này đôi khi dẫn đến thông tin sai lệch hoặc không chính xác.

Tuy vậy, với những cải tiến liên tục và sự ra đời của các phiên bản mới như GPT-4, ChatGPT ngày càng trở thành công cụ hữu ích trong nhiều lĩnh vực:

* Giáo Dục

+ Hỗ trợ học tập và giảng dạy: ChatGPT giúp học sinh, sinh viên giải đáp câu hỏi, cung cấp giải thích chi tiết về các chủ đề phức tạp, và giúp giảng viên tạo bài tập, đề thi, hoặc chấm điểm câu trả lời ngắn của học sinh.

+ Giảng viên ảo: ChatGPT có thể đóng vai trò là giảng viên ảo, giải thích các khái niệm trong các môn học như khoa học, toán học, ngữ văn, v.v.

* Dịch Vụ Khách Hàng

+ Hỗ trợ khách hàng tự động: ChatGPT có thể được sử dụng trong chatbot hỗ trợ khách hàng trực tuyến để trả lời các câu hỏi thường gặp (FAQs) hoặc giải quyết vấn đề của khách hàng.

+ Hỗ trợ trực tuyến trong ngân hàng: ChatGPT cung cấp hỗ trợ khách hàng 24/7 trong các ngân hàng, giúp kiểm tra số dư tài khoản, xử lý giao dịch, và giải đáp câu hỏi về sản phẩm tài chính.

* Chăm Sóc Sức Khỏe

+ Tư vấn sức khỏe ảo: ChatGPT giúp người dùng tra cứu triệu chứng, cung cấp thông tin về các bệnh lý, thuốc men, và các phương pháp điều trị. Mặc dù không thay thế bác sĩ, nó vẫn hỗ trợ trong việc cung cấp thông tin chung.

+ Hỗ trợ chăm sóc tâm lý: ChatGPT cung cấp hỗ trợ về tâm lý, giúp người dùng nhận diện cảm xúc và giảm căng thẳng.

* Sáng Tạo Nội Dung

+ Viết bài và sáng tạo nội dung: ChatGPT có thể tạo nội dung tự động cho blog, mạng xã hội, hoặc bài viết thương mại.

+ Sáng tác văn học và nghệ thuật: ChatGPT hỗ trợ sáng tác các tác phẩm văn học, thơ ca, truyện ngắn, và kịch bản phim.

* Tự Động Hóa Công Việc và Quản Lý Dữ Liệu

+ Quản lý và phân tích dữ liệu: ChatGPT giúp doanh nghiệp phân tích và tổng hợp dữ liệu, tạo ra các báo cáo tài chính hoặc kinh doanh nhanh chóng.

+ Tự động hóa quy trình công việc: ChatGPT tích hợp vào các hệ thống CRM hoặc ERP để tự động hóa các nhiệm vụ như trả lời email, tạo cuộc hẹn, hoặc phân loại yêu cầu của khách hàng.

* Phát Triển Phần Mềm

+ Hỗ trợ lập trình viên: ChatGPT trả lời các câu hỏi kỹ thuật, hỗ trợ tạo mã nguồn, kiểm tra và tối ưu hóa mã, và tìm kiếm lỗi.

+ Hướng dẫn lập trình: ChatGPT cung cấp hướng dẫn về cách sử dụng các công nghệ và framework như React, Django, TensorFlow.

* Dịch Thuật và Hỗ Trợ Ngôn Ngữ

+ Dịch thuật tự động: ChatGPT hỗ trợ dịch văn bản từ một ngôn ngữ sang ngôn ngữ khác, cung cấp bản dịch nhanh chóng và chính xác trong nhiều ngữ cảnh.

+ Chuyển ngữ ngữ cảnh: ChatGPT giúp chuyển ngữ ngữ cảnh trong các bài viết, giúp người sử dụng hiểu ngữ nghĩa trong các tình huống phức tạp.

* Quản Lý Dự Án và Tư Vấn Kinh Doanh

+ Lập kế hoạch và tổ chức dự án: ChatGPT hỗ trợ nhà quản lý trong việc lập kế hoạch, theo dõi tiến độ, và phân bổ nguồn lực cho dự án.

+ Tư vấn kinh doanh: ChatGPT cung cấp các chiến lược marketing, kế hoạch phát triển sản phẩm, hoặc các quyết định tài chính dựa trên phân tích dữ liệu.

* Giải Trí

+ Tạo câu đố, trò chơi và câu chuyện tương tác: ChatGPT có thể tạo câu đố trí tuệ, trò chơi văn bản và câu chuyện tương tác.

Tóm lại, ChatGPT hiện nay được áp dụng rộng rãi trong nhiều lĩnh vực từ giáo dục, dịch vụ khách hàng, chăm sóc sức khỏe, sáng tạo nội dung, đến quản lý công việc. Nhờ khả năng hiểu ngữ nghĩa và tạo văn bản tự nhiên, ChatGPT không chỉ nâng cao hiệu suất công việc mà còn tạo ra các ứng dụng sáng tạo, tiện ích trong đời sống hàng ngày.

## **3.2 Kiến trúc GPT**

ChatGPT, được phát triển bởi OpenAI, thể hiện bước nhảy vọt đáng kể trong lĩnh vực AI đàm thoại. Nó dựa trên kiến ​​trúc Generative Pre-training Transformer (GPT), cụ thể là GPT-3.5 và được thiết kế để tạo ra văn bản giống con người dựa trên đầu vào mà nó nhận được.

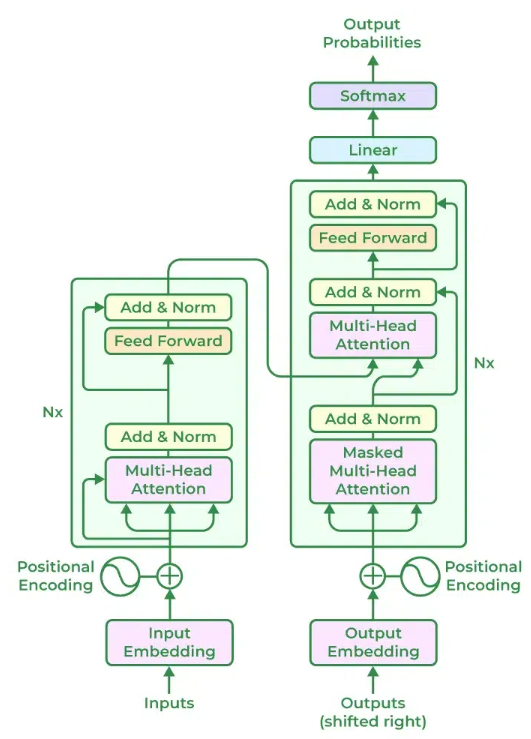
Kiến trúc GPT là một loại mô hình transformer dựa chủ yếu vào cơ chế attention. Transformers là một bước ngoặt lớn trong việc xử lý ngôn ngữ tự nhiên (NLP) nhờ khả năng xử lý hiệu quả của chúng trong việc đào tạo trên các tập dữ liệu lớn. Các mô hình GPT, bao gồm ChatGPT, dựa trên kiến ​​trúc này nhưng được đào tạo trước về dữ liệu văn bản và được tinh chỉnh cho các tác vụ cụ thể.

Sự phát triển của ChatGPT đã tiến bộ đáng kể trong những năm qua, từ khi thành lập ban đầu cho đến trạng thái hiện tại.

* GPT-1: Mô hình đầu tiên, GPT-1, được giới thiệu vào năm 2018. Nó bao gồm 117 triệu tham số và được đào tạo bằng bộ dữ liệu BooksCorpus. GPT-1 đã chứng minh tiềm năng của các mô hình transformer cho các nhiệm vụ NLP, nhưng nó có những hạn chế trong việc tạo ra văn bản dạng dài mạch lạc.
* GPT-2: Năm 2019, OpenAI đã phát hành GPT-2, một model lớn hơn nhiều với tối đa 1,5 tỷ tham số. GPT-2 cho thấy những cải tiến đáng kể trong việc tạo văn bản mạch lạc và phù hợp với ngữ cảnh. Nó được đào tạo trên một tập dữ liệu đa dạng có tên WebText, bao gồm dữ liệu từ nhiều trang web khác nhau. Việc phát hành GPT-2 đã nêu bật khả năng của mô hình trong việc thực hiện nhiều nhiệm vụ NLP mà không cần đào tạo về nhiệm vụ cụ thể.
* GPT-3: Bước nhảy vọt lớn tiếp theo đến với GPT-3, được phát hành vào năm 2020. GPT-3 tự hào với 175 tỷ tham số chưa từng có, khiến nó trở thành mẫu mô hình transformer lớn nhất vào thời điểm đó. Sự gia tăng lớn về tham số này cho phép GPT-3 tạo ra văn bản có độ phức tạp cao và nhận biết ngữ cảnh. Nó đã thể hiện những khả năng vượt trội trong đó mô hình có thể thực hiện các nhiệm vụ với mức độ đào tạo tối thiểu hoặc không cần đào tạo cụ thể về nhiệm vụ.
* GPT-3.5 và ChatGPT: ChatGPT dựa trên GPT-3.5, một phiên bản lặp lại của GPT-3 bao gồm các cải tiến và tối ưu hóa cho AI đàm thoại. GPT-3.5 tập trung vào việc cải thiện tính mạch lạc, khả năng lưu giữ bối cảnh và tính an toàn trong các phản hồi. Mô hình này được tinh chỉnh với trọng tâm cụ thể là các cuộc đối thoại tương tác, giúp mô hình này thành thạo hơn trong việc xử lý các cuộc hội thoại và đưa ra các phản hồi phù hợp, giống con người.

*Những thành phần của ChatGPT:*

* Khối Transformer: Thành phần chính của ChatGPT là nhiều khối transformer. Mỗi khối gồm 2 thành phần chính:
  + - Multi-Head Self-Attention**:** Cơ chế này cho phép mô hình tập trung đồng thời vào các phần khác nhau của văn bản đầu vào, nắm bắt các mối quan hệ theo ngữ cảnh khác nhau.
    - Feed-Forward Neural Network**:** áp dụng các phép biến đổi phi tuyến tính để tinh chỉnh thêm cách biểu diễn.
* Positional Encoding: Không giống như recurrent neural networks (RNN), transformer không xử lý dữ liệu một cách tuần tự. Để giữ thứ tự của các từ, ChatGPT sử dụng mã hóa vị trí, bổ sung thông tin về vị trí của từng token trong chuỗi. Điều này giúp mô hình hiểu ngữ cảnh tốt hơn.
* Pre-training và Fine-turning: ChatGPT trải qua quá trình đào tạo gồm hai bước:
  + - Pre-training: Mô hình được đào tạo trên một kho dữ liệu văn bản lớn, học cách dự đoán từ tiếp theo trong câu. Giai đoạn này giúp mô hình hiểu ngữ pháp và một số khả năng suy luận.
    - Fine-turning: Sau khi pre-training, mô hình được tinh chỉnh trên tập dữ liệu hẹp hơn với người đánh giá tuân theo các hướng dẫn cụ thể. Bước này giúp tinh chỉnh mô hình cho các bài toán cụ thể.



Hình 18: Hình vẽ mô tả chi tiết thành phần của ChatGPT

ChatGPT bao gồm nhiều lớp transformer xếp chồng lên nhau. Mỗi lớp có hai thành phần chính:

* Self-Attention Mechanism**:** Mỗi token trong đầu vào liên quan đến mọi token khác, cho phép mô hình hiểu ngữ cảnh từ tất cả các phần của đầu vào.
* Feed-Forward Networks: Các mạng nơ-ron này áp dụng các phép biến đổi đối với thông tin đầu vào, cho phép mô hình hiểu các mẫu phức tạp.

Cơ chế attention là một cơ chế cực kì quan trọng trong ChatGPT giúp cho khả năng hiểu và tạo văn bản của ChatGPT. Nó liên quan đến: Query, Key, and Value Matrices**,** Scaled Dot-Product Attention**,** Multi-Head Attention

Sau khi đi qua các lớp transformer, trạng thái ẩn cuối cùng được sử dụng để tạo token đầu ra. Mô hình sử dụng lớp softmax để dự đoán phân bố xác suất theo từ vựng cho token tiếp theo, tạo văn bản theo từng bước.

Kiến trúc của ChatGPT, dựa trên mô hình GPT mạnh mẽ, thể hiện tiềm năng của các mô hình biến áp trong AI đàm thoại. Bằng cách tận dụng cơ chế attention, pre-trained và fine-turned, ChatGPT đạt được hiệu suất vượt trội trong việc tạo ra văn bản giống con người. Khi những tiến bộ tiếp tục diễn ra, ChatGPT và những phiên bản kế nhiệm của nó đã sẵn sàng trở nên tích hợp hơn nữa trong nhiều lĩnh vực khác nhau, thay đổi cách chúng ta tương tác với máy móc.

## **3.3 ChatGPT trong ASAG**

Trong phần này, chúng ta sẽ đánh giá khả năng sử dụng chatbot dựa trên mô hình ngôn ngữ lớn (LLM), cụ thể là GPT-3.5 của OpenAI, để chấm điểm tự động cho các câu trả lời thi dạng câu hỏi ngắn. Kết quả đánh giá dưới đây được dựa trên các thực nghiệm thực tế và cho thấy rằng việc sử dụng ChatGPT để chấm điểm các câu trả lời ngắn là hoàn toàn khả thi.

Loại câu hỏi tối ưu: Các câu hỏi dạng câu trả lời ngắn, đặc biệt là các câu hỏi tập trung vào nội dung, được cho là khả thi nhất để chấm điểm tự động bằng LLM. Các câu hỏi yêu cầu sửa lỗi ngữ pháp hoặc trích dẫn thường kém hiệu quả hơn, và các bài luận dài thường dẫn đến việc mô hình dự đoán điểm cao hơn mức thực tế.

Thiết kế prompt:

* Trình bày dữ liệu (câu hỏi, câu trả lời, điểm số) dưới dạng mô tả ngôn ngữ tự nhiên thay vì các định dạng chính thức (như JSONL) cho kết quả tốt hơn.
* Mô hình hoạt động hiệu quả hơn trong cài đặt one-shot, khi một ví dụ về điểm số được cung cấp, thay vì zero-shot, khi không có ví dụ nào.
* Việc đưa vào nhiều câu trả lời trong một prompt làm giảm hiệu suất của mô hình. Do đó, mỗi câu trả lời nên được đánh giá riêng lẻ để đạt kết quả tối ưu.

Ảnh hưởng của số lượng ví dụ: Việc đưa vào ví dụ về các câu trả lời đã được chấm điểm giúp cải thiện hiệu suất của mô hình, nhưng hiệu quả giảm dần nếu có quá nhiều câu trả lời trong một prompt. Điều này cho thấy mô hình hoạt động tốt hơn khi chỉ tập trung vào việc đánh giá một câu trả lời tại một thời điểm.

Đánh giá LLM cho chấm điểm dựa trên nội dung: Nghiên cứu kết luận rằng mặc dù mô hình LLM có triển vọng trong việc chấm điểm dựa trên nội dung, đặc biệt là với các câu hỏi ngắn, nhưng cần phải có thêm các tiêu chí chấm điểm chi tiết (ví dụ: cách xử lý các loại lỗi hoặc tiêu chuẩn chấm điểm) để đảm bảo hệ thống chấm điểm chính xác và đáng tin cậy hơn.

Đánh giá này chỉ ra rằng việc sử dụng mô hình LLM như GPT-3.5 để chấm điểm tự động cho các câu hỏi dạng câu trả lời ngắn là khả thi. Tuy nhiên, có một số yếu tố (ví dụ: loại câu hỏi, thang điểm, thiết kế prompt) có thể ảnh hưởng đến hiệu suất của mô hình. Trường hợp sử dụng hiệu quả nhất có vẻ là các câu hỏi ngắn, tập trung vào nội dung, nơi việc dự đoán điểm số có thể được tự động hóa một cách đáng tin cậy. Tuy nhiên, cần tiếp tục cải tiến để đảm bảo tính nhất quán giữa các ngành học và loại câu hỏi.

**Cách hướng dẫn ChatGPT chấm điểm trong ASAG**:

Bạn hãy mô phỏng quá trình chấm điểm của giáo viên. Bạn đang phải chấm điểm cho bài kiểm tra và câu hỏi cần được trả lời dưới dạng:

"""Câu hỏi"""

"""Câu trả lời của học sinh"""

"""Điểm trong khoảng từ 0-5 cho mỗi câu hỏi"""

*Câu hỏi*: Vai trò của chương trình nguyên mẫu trong giải quyết vấn đề là gì

*Đáp án*: Mô phỏng hành vi của các phần của sản phẩm phần mềm mong muốn

*Câu trả lời của học sinh*: Các vấn đề rủi ro cao được giải quyết trong chương trình nguyên mẫu để đảm bảo tính khả thi của chương trình. Một nguyên mẫu cũng có thể được sử dụng để chứng minh cho công ty thấy rằng phần mềm có thể được lập trình

**Kết quả nhận được từ ChatGPT**:

*Câu hỏi*: Vai trò của chương trình nguyên mẫu trong giải quyết vấn đề là gì?

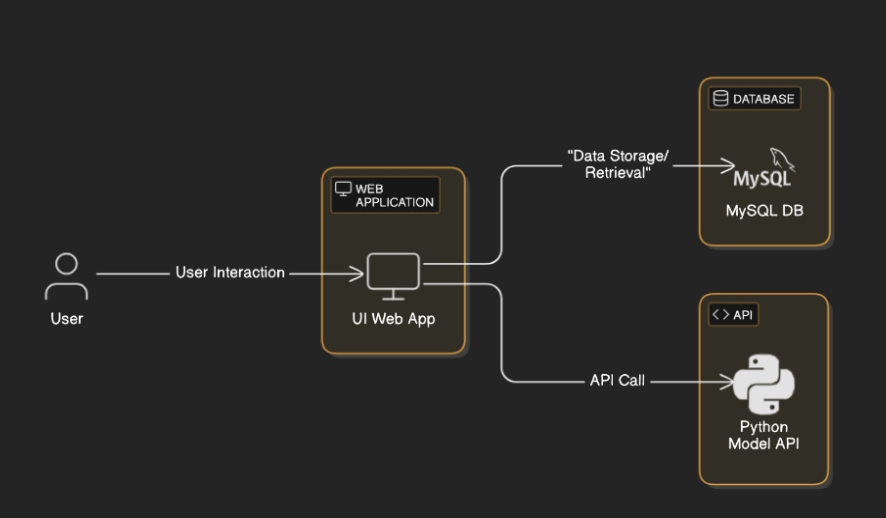
*Câu trả lời của học sinh*: "Các vấn đề rủi ro cao được giải quyết trong chương trình nguyên mẫu để đảm bảo tính khả thi của chương trình. Một nguyên mẫu cũng có thể được sử dụng để chứng minh cho công ty thấy rằng phần mềm có thể được lập trình."

*Điểm*: 4/5 (Câu trả lời của học sinh có ý chính đúng, nêu rõ vai trò của nguyên mẫu trong việc giải quyết các rủi ro và tính khả thi của sản phẩm.

Tuy nhiên, câu trả lời chưa đề cập rõ đến mục đích mô phỏng hành vi các phần của sản phẩm phần mềm như trong đáp án gốc, nên bị thiếu một chút chi tiết về khía cạnh này.)

# **CHƯƠNG 4: XÂY DỰNG HỆ THỐNG ASAG**

## **4.1 Kiến trúc tổng quan hệ thống**



Hình 19: Hình vẽ mô tả kiến trúc tổng quan hệ thống ASAG

WebApp (Ứng dụng Web):

* Chức năng chính: Đây là phần giao diện người dùng, nơi người dùng tương tác với hệ thống. Ứng dụng web sẽ cung cấp các chức năng để người dùng nhập câu trả lời, hiển thị kết quả chấm điểm, và tương tác với hệ thống.
* Công nghệ sử dụng:

+ Java: Được sử dụng để xây dựng ứng dụng web.

+ Spring Boot: Cung cấp một framework nhẹ và mạnh mẽ để xây dựng các ứng dụng web.

+ Thymeleaf: Là một thư viện template engine cho Java, giúp tạo giao diện người dùng động cho ứng dụng.

* Luồng dữ liệu WebApp:

+ Người dùng nhập câu trả lời vào form trên giao diện.

+ Client (Browser) gửi yêu cầu đến webapp thông qua API

+ Webapp sử dụng Spring Boot để xử lý yêu cầu và gọi Model (BERT, ChatGPT, Transformer) để chấm điểm câu trả lời.

+ Sau khi nhận được kết quả từ Model, dữ liệu sẽ được trả về cho giao diện người dùng để hiển thị điểm số hoặc kết quả chấm.

Database (Cơ sở dữ liệu)

* Chức năng chính: Lưu trữ các thông tin liên quan đến câu hỏi, đáp án cho hệ thống.
* Công nghệ sử dụng:

+ MySQL: Là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) được sử dụng để lưu trữ các bảng dữ liệu của ứng dụng. MySQL sẽ lưu trữ thông tin câu hỏi đáp án

+ Spring Data JPA: Sử dụng để kết nối và thao tác với cơ sở dữ liệu thông qua các đối tượng Java, giúp giảm bớt các thao tác SQL trực tiếp.

* Luồng dữ liệu Database:

+ Khi người dùng gửi câu trả lời, webapp sẽ kết nối với CSDL để lấy đáp án tương ứng với các câu hỏi trước khi gửi yêu cầu chấm điểm đến model

Model (Mô hình AI)

* Chức năng chính: Mô hình AI sẽ thực hiện việc chấm điểm tự động các câu trả lời của học sinh dựa trên mô hình ngôn ngữ tự nhiên như BERT, ChatGPT
* Công nghệ sử dụng:

+ BERT: Mô hình ngôn ngữ dựa trên Transformer, có thể được fine-tune cho bài toán chấm điểm câu trả lời ngắn (ASAG).

+ ChatGPT: Sử dụng API của ChatGPT để hỗ trợ việc xử lý và chấm điểm câu trả lời ngắn (ASAG).

* Luồng dữ liệu Model:

+ Webapp nhận câu trả lời từ người dùng và gửi dữ liệu này đến Model (BERT, ChatGPT).

+ Mô hình sẽ tính toán điểm số dựa trên câu trả lời.

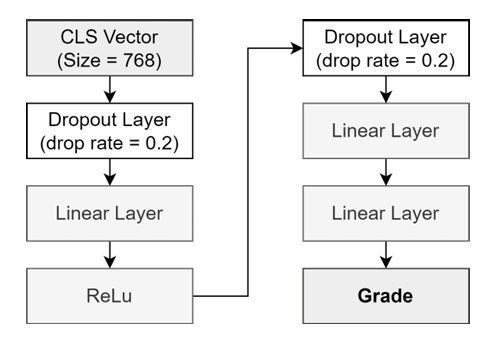
+ Kết quả (điểm số) từ mô hình sẽ được gửi lại về webapp và trả về cho người dùng.

Giao tiếp giữa các thành phần:

* Người dùng nhập câu trả lời trên WebApp.
* WebApp truy xuất đáp án tương ứng cho mỗi câu hỏi từ cơ sở dữ liệu.
* WebApp gửi câu hỏi, câu trả lời của học sinh và đáp án thông qua API đến mô hình AI.
* Mô hình AI chấm điểm và gửi kết quả trở lại WebApp qua API, sau đó hiển thị kết quả lên giao diện người dùng.

## **4.2 Xây dựng mô hình**

### **4.2.1 Kiến trúc BERT Regressor**



Hình 20: Hình vẽ mô tả kiến trúc ANN của BERT Regressor

- Input Layer: Đầu vào của mô hình là vector [CLS] từ mô hình BERT, có kích thước 768. Đây là vector đại diện cho toàn bộ câu trả lời sau khi qua lớp tokenization và BERT embedding. Vector này mang đặc trưng ngữ nghĩa tổng hợp từ câu trả lời của học sinh.

- Dropout Layer (drop rate = 0.2): Lớp dropout với tỉ lệ 0.2 giúp tránh overfitting, bằng cách ngẫu nhiên loại bỏ 20% các neurals trong quá trình huấn luyện. Điều này giúp mô hình tổng quát hơn và không phụ thuộc quá nhiều vào các đặc trưng cụ thể.

- Linear Layer (Fully Connected): Sau lớp dropout, vector CLS được đưa qua một lớp tuyến tính (fully connected), giúp giảm số chiều của vector từ 768 xuống một kích thước nhỏ hơn, thường là một giá trị trung gian để tạo ra các đặc trưng mới từ các đặc trưng ban đầu.

- ReLU Activation: Sau lớp tuyến tính đầu tiên, chúng ta áp dụng hàm kích hoạt ReLU. ReLU giúp mô hình học các mối quan hệ phi tuyến, cho phép mô hình có khả năng học được các đặc trưng phức tạp. Formula: ReLU(x) = max(0, x)

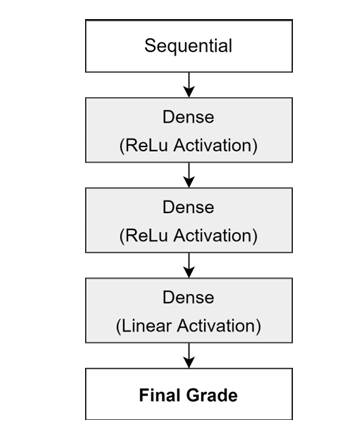
- Dropout Layer (drop rate = 0.2): Lớp dropout thứ hai với tỉ lệ 0.2 tiếp tục giúp giảm overfitting trong quá trình huấn luyện. Đây là một kỹ thuật quan trọng để tăng tính tổng quát của mô hình.

- Linear Layer: Lớp tuyến tính thứ hai tiếp tục chuyển các đặc trưng từ lớp trước đó thành một không gian có số chiều thấp hơn, giúp mô hình dự đoán chính xác hơn.

- Linear Layer (Final Output Layer): Cuối cùng, đầu ra được đưa qua một lớp tuyến tính để dự đoán điểm số cuối cùng. Đây là lớp đưa ra kết quả dự đoán cho điểm số của câu trả lời.

- Grade: Điểm số cuối cùng được đưa ra từ lớp tuyến tính cuối cùng

### **4.2.2 Kiến trúc ANN**



Hình 21: Hình vẽ mô tả kiến trúc của ANN được sử dụng để final grade

- Sequential(): Đây là mô hình tuần tự, nơi các lớp được xếp chồng tuần tự từ đầu vào đến đầu ra.

- Dense (ReLU Activation): Lớp đầu tiên là một lớp Dense (fully connected layer) sử dụng hàm kích hoạt ReLU (Rectified Linear Unit). ReLU giúp mạng học được các đặc trưng phi tuyến, giữ các giá trị đầu ra dương và gán 0 cho các giá trị âm.

- Dense (ReLU Activation): Lớp thứ hai cũng là một lớp Dense với hàm kích hoạt ReLU. Mục tiêu là tiếp tục trích xuất các đặc trưng từ dữ liệu đầu vào.

- Dense (Linear Activation): Lớp cuối cùng sử dụng hàm kích hoạt tuyến tính (Linear Activation). Điều này phù hợp cho bài toán hồi quy, vì đầu ra cần là một giá trị thực (ví dụ: điểm số cuối cùng).

- Final Grade: Đầu ra của mô hình là giá trị điểm cuối cùng ("Final Grade") được dự đoán từ dữ liệu đầu vào.

### **4.2.3 Tích hợp ChatGPT**

Prompts trong ChatGPT là hướng dẫn hoặc câu lệnh mà người dùng nhập vào giao diện để nhận phản hồi.

Để nhận được phản hồi cho một prompt cụ thể, phải sử dụng Completion API:

curl --location 'https://api.openai.com/v1/chat/completions' \

--header 'Content-Type: application/json' \

--header 'Authorization: Bearer $OPENAI\_API\_KEY' \

--data '{

"model": "gpt-3.5-turbo",

"messages": [

{

"role": "system",

"content": "What is springboot?"

}

],

"temperature": 1

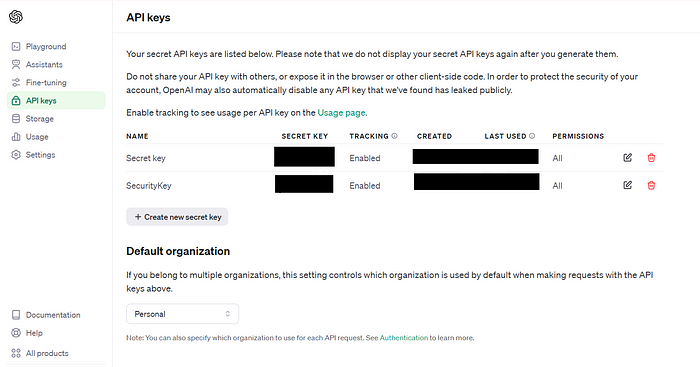
"max\_tokens" :256

}'

Các tham số được sử dụng:

* Model: Tham số này xác định phiên bản mô hình sẽ nhận các yêu cầu. Có nhiều phiên bản mô hình khác nhau, và cho tác vụ này, chúng ta sẽ sử dụng mô hình “gpt-3.5-turbo”, đây là phiên bản mới nhất có sẵn cho công chúng.
* Messages: Mỗi tin nhắn có hai phần quan trọng: “role” và “content.” “Role” cho biết tin nhắn này là của người dùng hay trợ lý. “Content” là nội dung thực tế của tin nhắn.
* Temperature: Mặc định, tham số này được đặt ở mức 1, và nó có thể dao động từ 0 đến 2. Tham số này điều chỉnh mức độ ngẫu nhiên của phản hồi. Giá trị cao hơn làm cho phản hồi trở nên ngẫu nhiên hơn, trong khi giá trị thấp hơn làm cho phản hồi tập trung và dễ dự đoán hơn.
* max\_tokens: Thông thường, không có giới hạn, nhưng bạn có thể sử dụng cài đặt này để quyết định số lượng từ hoặc ký tự tối đa trong phản hồi. Nó giúp kiểm soát các phản hồi dài và giữ chi phí trong tầm kiểm soát.

Để có thể call API trên, trước hết phải tạo API\_KEY để thực hiện quá trình xác thực:



Hình 22: Hình vẽ mô tả giao diện để lấy API\_KEY chatGPT

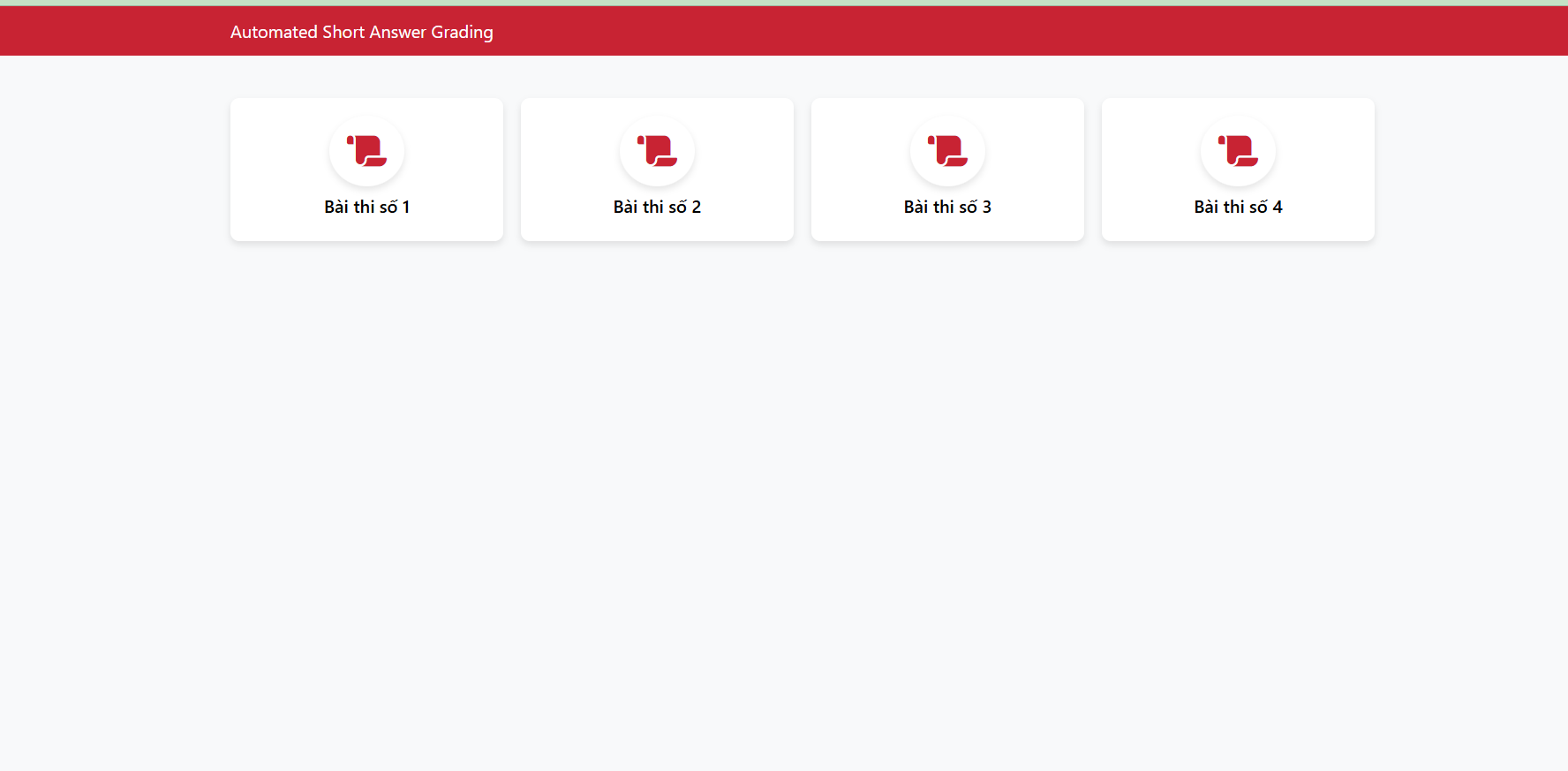
### **4.2.4 Xây dựng API chấm điểm dựa trên mô hình BERT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Endpoint: {{host}} /predict  Mục đích: Để chấm điểm cho các câu trả lời của học sinh tương ứng với từng câu hỏi  Method: POST  Body: application/json | | | | |
| STT | Tham số | Bắt buộc | Kiểu dữ liệu | Mô tả |
| 1 | requests |  | List<Object> |  |
|  | question\_id | x | String | Mã câu hỏi |
|  | question | x | String | Nội dung câu hỏi |
|  | response | x | String | Câu trả lời của học sinh |
|  | key | x | String | Nội dung đáp án |
| Response: | | | |  |
| STT | Tham số | | Kiểu dữ liệu | Mô tả |
| 1 | results | | List<Object> |  |
|  | question\_id | | String | Mã câu hỏi |
|  | result | | Double | Điểm cho câu hỏi tương ứng |

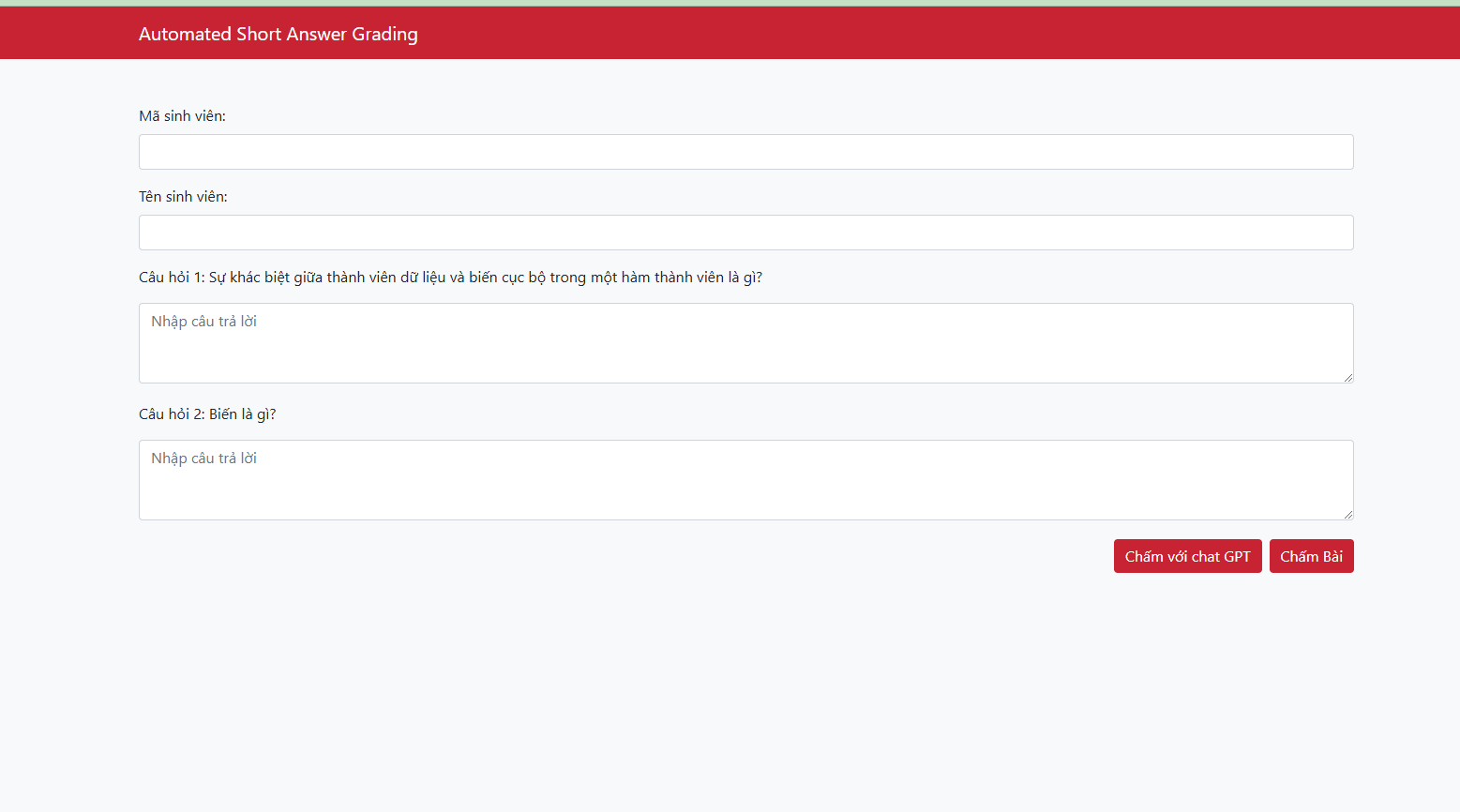
## **4.3 Triển khai ứng dụng web cho ASAG**

Giao diện cho ứng dụng Web bao gồm bốn màn hình chính: Homepage của webapp, Màn hình chi tiết bài thi, Màn hình hiển thị kết quả chấm điểm, Màn hình hiển thị kết quả chấm điểm dựa trên chatGPT

Màn hình 1: Homepage của webapp



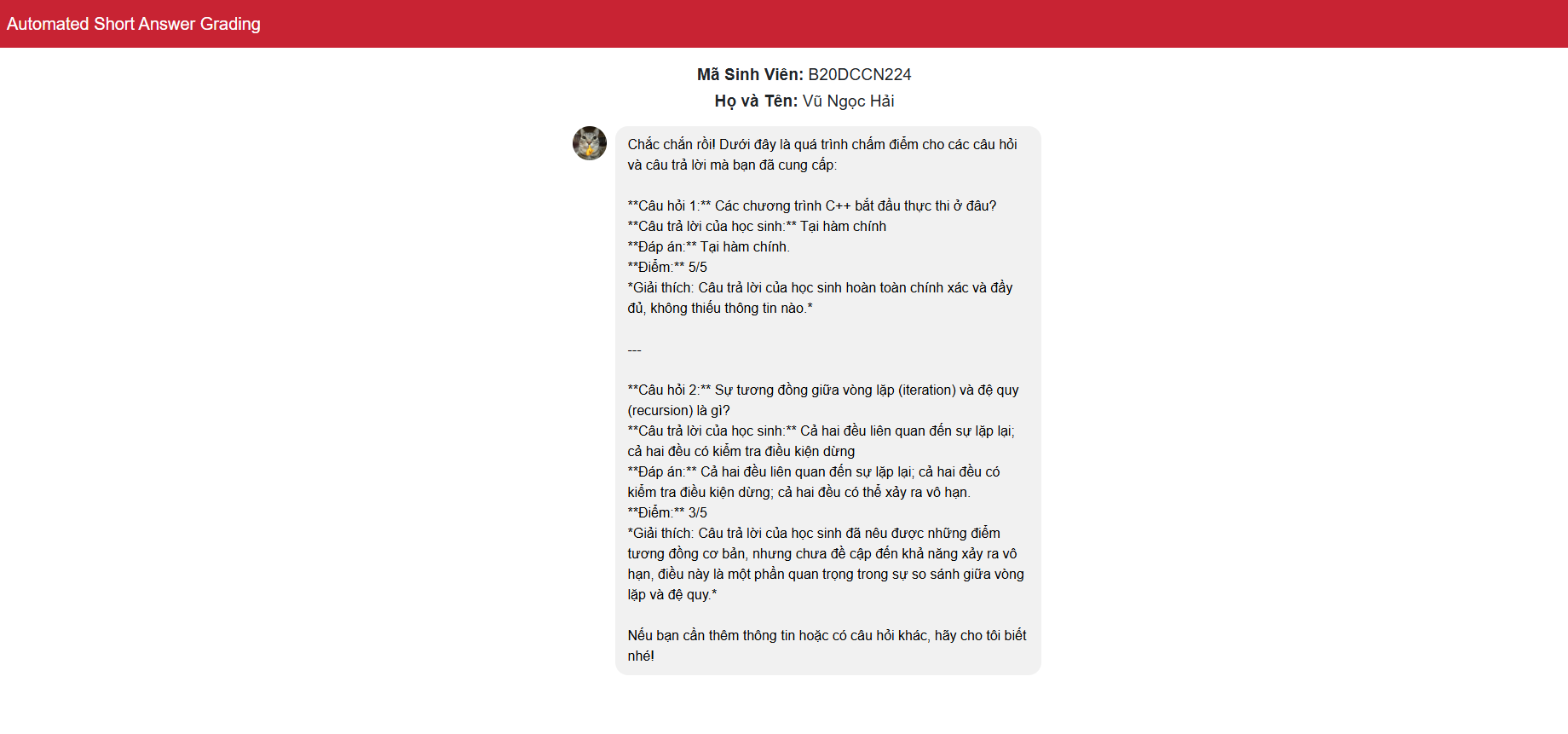
Màn hình 2: Màn hình chi tiết bài thi



Màn hình 3: Màn hình hiển thị kết quả chấm điểm



Màn hình 4: Màn hình hiển thị kết quả chấm điểm dựa trên chatGPT



Xây dựng CSDL cho webapp: Bao gồm bảng questions (id, question, key) được sử dụng để lưu trữ các câu hỏi và đáp án cho các bài kiểm tra hiển thị trên webapp.

## **4.4 Kiểm thử và đánh giá hệ thống ASAG**

**Chấm điểm với BERT:**

**Case 1:** Không trả lời

**Câu hỏi: “**Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?”

**Đáp án: “**Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.”

**Câu trả lời học sinh:** “”

**Điểm hệ thống chấm:** 0.37 / 5

**Case 2:** Trả lời giống đáp án

**Câu hỏi:** Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?

**Đáp án:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Câu trả lời học sinh:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Điểm hệ thống chấm:** 4.75 / 5

**Case 3:** Trả lời đúng 1 ý

**Câu hỏi:** Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?

**Đáp án:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Câu trả lời học sinh:** “Cả hai đều liên quan đến sự lặp lại”

**Điểm hệ thống chấm:** 4.39 / 5

**Chấm điểm với ChatGPT:**

**Case 1:** Không trả lời

**Câu hỏi:** Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?

**Đáp án:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Câu trả lời học sinh:** “”

**Điểm hệ thống chấm:** 0/5 (Học sinh không cung cấp câu trả lời)

**Case 2:** Trả lời giống đáp án

**Câu hỏi:** Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?

**Đáp án:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Câu trả lời học sinh:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Điểm hệ thống chấm:** 5/5

**Case 3:** Trả lời đúng 1 ý

**Câu hỏi:** Sự tương đồng giữa vòng lặp (iteration) và đệ quy (recursion) là gì?

**Đáp án:** Cả hai đều liên quan đến sự lặp lại; cả hai đều có kiểm tra điều kiện dừng; cả hai đều có thể xảy ra vô hạn.

**Câu trả lời học sinh:** Cả hai đều liên quan đến sự lặp lại.

**Điểm hệ thống chấm:** 2/5

Qua kết quả kiểm thử, chúng ta có bảng sau phân tích độ chính xác giữa BERT và GPT:

Bảng 5: Bảng so sánh kết quả kiểm thử đánh giá mô hình BERT và GPT

|  |  |  |
| --- | --- | --- |
| Yếu tố | BERT | ChatGPT |
| Độ chính xác | Chưa cao, đôi khi điểm không phù hợp | Cao, phản ánh đúng thực tế |
| Xử lý câu trả lời trống | Điểm không về 0, thiếu hợp lý | Chính xác, điểm về 0 |
| Phân tích chi tiết | Thiếu nhạy bén, điểm cao không hợp lý | Đánh giá đầy đủ mức độ đúng/sai |

# **KẾT LUẬN**

Trí tuệ nhân tạo (AI) đang ngày càng được ứng dụng rộng rãi, đặc biệt trong lĩnh vực giáo dục. Hệ thống Automatic Short Answer Grading (ASAG) được phát triển nhằm tự động chấm điểm các câu trả lời thi dạng câu hỏi ngắn, thông qua việc sử dụng các mô hình ngôn ngữ mạnh mẽ như BERT và ChatGPT. Hệ thống ASAG áp dụng phương pháp kết hợp giữa mô hình BERT Regressor và Fine-tuned Question-Answer BERT để tính toán điểm số cho mỗi câu trả lời, dựa trên các yếu tố như sự tương đồng giữa đáp án chuẩn (Key - K) và câu trả lời của học sinh (Response - R).

Kết quả nghiên cứu cho thấy mô hình BERT có khả năng tạo ra các điểm số dự đoán rất chính xác khi kết hợp với mô hình Question-Answer BERT, giúp xác định mức độ tương đồng giữa câu trả lời của học sinh và đáp án chuẩn. Mô hình này không chỉ tính toán điểm số dựa trên nội dung câu trả lời mà còn giúp giảm thiểu sai lệch khi đánh giá các câu trả lời dài hơn đáp án chuẩn.

Ngoài ra, việc sử dụng ChatGPT, một mô hình ngôn ngữ tiên tiến của OpenAI, trong việc chấm điểm câu trả lời ngắn cũng được chứng minh là khả thi. Đặc biệt, các câu hỏi ngắn, tập trung vào nội dung, có thể được ChatGPT xử lý rất tốt, giúp giảm thiểu sai sót trong quá trình chấm điểm tự động. Tuy nhiên, khi số lượng câu trả lời trong một prompt quá nhiều, hiệu suất của mô hình sẽ giảm, vì vậy việc đánh giá từng câu trả lời riêng biệt là cần thiết để đạt kết quả tối ưu.

Tóm lại, hệ thống ASAG kết hợp với các mô hình BERT và ChatGPT đã mở ra một hướng đi mới trong việc tự động chấm điểm câu trả lời thi ngắn, với những kết quả khả quan trong các bài kiểm tra tập trung vào nội dung. Tuy nhiên, vẫn cần tiếp tục nghiên cứu và cải thiện các tiêu chí chấm điểm cũng như thiết kế prompt để nâng cao tính chính xác và khả năng áp dụng rộng rãi của hệ thống trong các lĩnh vực giáo dục khác nhau.

**TÀI LIỆU THAM KHẢO**

[1] Steven Burrows·Iryna Gurevych·Benno Stein, *The Eras and Trends of Automatic Short Answer Grading.*

[2] Jai Garg·Kumar·Apurva·Jatin Papreja·Dr. Goonjan Jain, *Domain-Specific Hybrid BERT based System for Automatic Short Answer Grading*

[3] Chul Sung·Tejas Indulal Dhamecha·Swarnadeep Saha·Tengfei Ma·Vinay Reddy· Rishi Arora*, Pre-Training BERT on Domain Resources for Short Answer Grading*

[4] Li-Hsin Chang·Filip Ginter, *Automatic Short Answer Grading for Finnish with ChatGPT*

[5] Lan Jiang·Nigel Bosch, *Short answer scoring with GPT-4*