Lecture 23: Multilingual NLP

Resources

Multilingual NLP and its Difficulties

Two varieties of Multilingual NLP

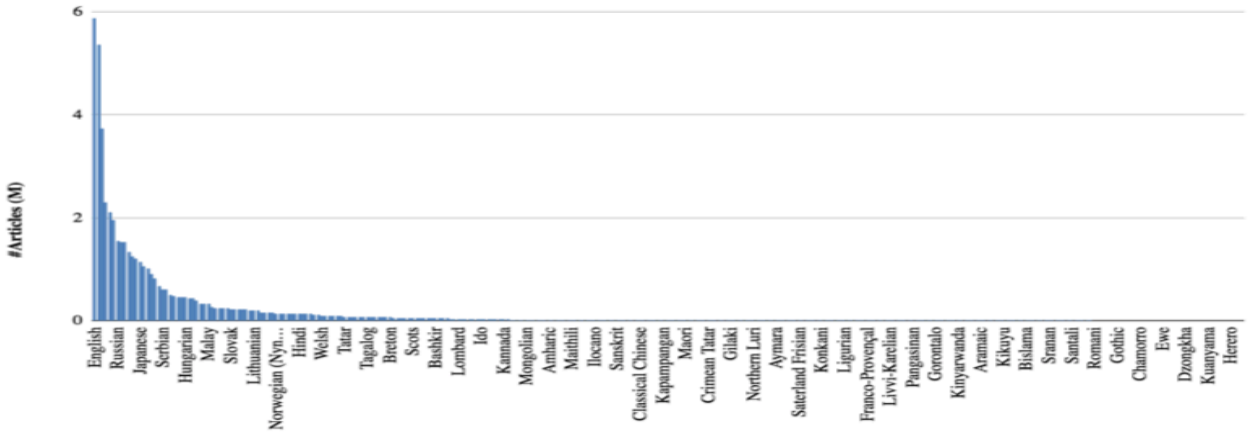
NLP đa ngôn ngữ là xử lý ngôn ngữ tự nhiên trong nhiều ngôn ngữ khác nhau. Có hai loại chính:

- NLP đơn ngữ trong nhiều ngôn ngữ: Điều này có nghĩa là bất kỳ tác vụ nào bạn có thể thực hiện bằng tiếng Anh, bạn cũng có thể thực hiện bằng các ngôn ngữ khác không phải tiếng Anh. Ví dụ như question answering, sentiment analysis, chatbots, code generation, v.v.
- NLP liên ngôn ngữ: Đây là các tác vụ xử lý nhiều ngôn ngữ cùng lúc, chẳng hạn như machine translation, crosslingual QA, v.v. Crosslingual QA là việc trả lời câu hỏi khi tài liệu nguồn ở một ngôn ngữ khác. Ví dụ, nếu tôi đặt câu hỏi bằng tiếng Nhật, hệ thống có thể tìm thông tin bằng tiếng Anh và trả lời câu hỏi bằng tiếng Nhật.

Paucity of data

Hiện nay, nhiều hệ thống của chúng ta được huấn luyện bằng cách sử dụng các tập dữ liệu lớn. Một trong những thách thức lớn nhất trong xử lý ngôn ngữ đa ngữ là sự thiếu hụt dữ liệu ở nhiều ngôn ngữ mà chúng ta quan tâm.

Ví dụ điển hình là số lượng bài viết trên Wikipedia ở các ngôn ngữ khác nhau. Bạn có thể thấy rằng số lượng bài viết giảm đi rất nhanh sau 20 đến 30 ngôn ngữ đầu tiên, với tiếng Anh đứng đầu. Điều này cũng tương tự đối với văn bản chung trên internet, nhưng không quá khắc nghiệt như trên Wikipedia.



Một điểm khác cần lưu ý là dữ liệu đã được chú thích (annotated data) còn ít hơn nữa. Dữ liệu chú thích là tập con của dữ liệu đơn ngữ, vì vậy chúng ta có ít dữ liệu hơn cho dịch máy, gán nhãn chuỗi, đối thoại, trả lời câu hỏi và các nhiệm vụ khác như instruction following.

Linguistic Peculiarities

Xử lý ngôn ngữ tự nhiên đa ngôn ngữ gặp nhiều thách thức do sự khác biệt cơ bản giữa các ngôn ngữ, đặc biệt khi sử dụng các mô hình được huấn luyện chủ yếu trên tiếng Anh. Dưới đây là một số thách thức chính:

1. **Hình thái học (Morphology):**

- Trong tiếng Anh, thay đổi hình thái học ở giữa từ khá hiếm (ví dụ: "goose" thành "geese")
- Nhiều ngôn ngữ khác thường xuyên có các thay đổi ở giữa từ, điều này có thể gây khó khăn cho các công cụ như SentencePiece

2. **Dấu và thanh điệu:**

- Nhiều ngôn ngữ sử dụng dấu và thanh điệu để chỉ âm điệu.
- Ví dụ: tiếng Tây Ban Nha, tiếng Pháp sử dụng dấu
- Ngôn ngữ Yoruba (Nigeria) có nhiều dấu thanh điệu nhưng thường được viết không dấu, tạo ra sự mơ hồ và đa dạng từ vựng
- Pinyin (phiên âm tiếng Trung) sử dụng số ở cuối âm tiết để chỉ thanh điệu

3. **Hệ thống chữ viết khác nhau (CJK - Chinese, Japanese, Korean):**

- Chữ Hán (Trung Quốc): Sử dụng cả chữ La-tinh và biểu tượng ý nghĩa
- Chữ Nhật: Kết hợp cả biểu tượng ý nghĩa và ký tự phiên âm
- Chữ Hàn: Dựa hoàn toàn vào phát âm, kết hợp ba âm tiết trong một ký tự (ví dụ: "한국" = "han" + "guk")

4. **Những thách thức khác:**

- Phương ngữ: Sự khác biệt trong cách nói giữa các vùng miền
- Thiếu hệ thống chữ viết chuẩn hóa: Nhiều ngôn ngữ không có chuẩn viết thống nhất, người dùng có thể viết bằng chữ bản địa hoặc chữ La-tinh

So với các ngôn ngữ khác, tiếng Anh tương đối đơn giản với hệ thống chữ viết chuẩn hóa, hình thái học đơn giản và số lượng ký tự hạn chế.

Multilingual Learning

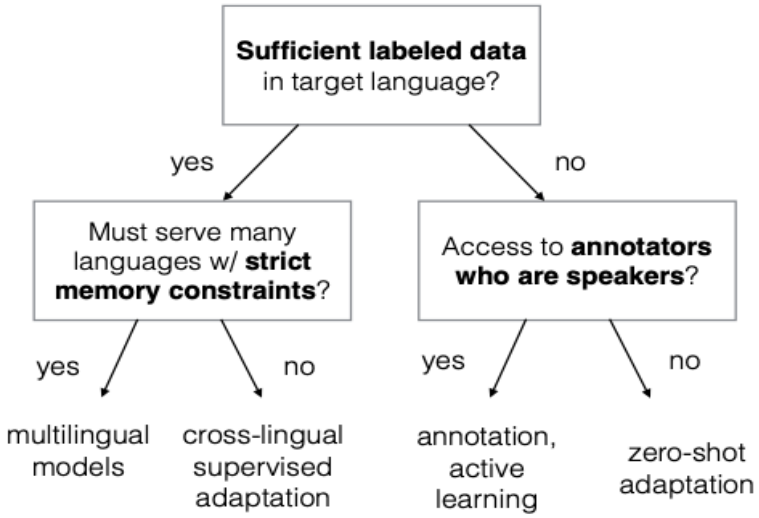
Trong những năm gần đây, một xu hướng nổi bật trong xử lý ngôn ngữ tự nhiên là khả năng học các mô hình xử lý đa ngôn ngữ. Các ngôn ngữ có thể học hỏi lẫn nhau, đặc biệt là những ngôn ngữ ít dữ liệu có thể được cải thiện độ chính xác nhờ vào dữ liệu từ các ngôn ngữ có nhiều tài nguyên hơn. Đây là một dạng của “transfer learning”.

Một lợi ích lớn khác của học đa ngôn ngữ là tính thực tiễn. Trước đây, Google Translate phải triển khai hàng trăm mô hình riêng lẻ, ví dụ như "English to Chinese", "English to Japanese", "English to French", "English to Spanish", và ngược lại. Điều này đòi hỏi phải quản lý nhiều máy chủ khác nhau cho từng mô hình. Tuy nhiên, hiện nay, chỉ cần một mô hình lớn duy nhất có thể

xử lý tất cả các ngôn ngữ cùng lúc. Điều này không chỉ giúp giảm số lần triển khai mà còn cho phép mô hình trở nên lớn hơn và mạnh mẽ hơn nhờ vào học chuyển giao. Do đó, nhiều nơi đang chuyển đổi sang mô hình này để xử lý các vấn đề đa ngôn ngữ.

High-level Multilingual Learning Flowchart

Khi cần xây dựng một mô hình xử lý ngôn ngữ khác tiếng Anh, đây là quy trình ra quyết định để chọn phương pháp phù hợp nhất:



1. Đầu tiên, kiểm tra có đủ dữ liệu được gán nhãn cho ngôn ngữ mục tiêu hay không?

- Với bài toán dịch máy: cần ít nhất 1 triệu câu
- Với bài toán phân loại: khoảng 1000 câu có thể đủ

2. Xem xét yêu cầu về bộ nhớ:

- Nếu cần xử lý nhiều ngôn ngữ với giới hạn bộ nhớ nghiêm ngặt → sử dụng mô hình đa ngôn ngữ
- Nếu không có giới hạn bộ nhớ → có thể dùng mô hình đa ngôn ngữ hoặc điều chỉnh (adapt) mô hình cho ngôn ngữ cụ thể để đạt hiệu quả tốt hơn

3. Trường hợp không đủ dữ liệu gán nhãn:

- Nếu có nguồn lực → Thu thập và gán nhãn dữ liệu mới (khuyến nghị khoảng 1000 mẫu)
- Mặc dù có nhiều nghiên cứu về "zero-shot adaptation" (thích ứng không cần dữ liệu), nhưng trong thực tế triển khai, việc gán nhãn một lượng nhỏ dữ liệu vẫn hiệu quả hơn

Lưu ý: Zero-shot adaptation có thể hữu ích trong giai đoạn thử nghiệm/demo, ví dụ như khi muốn chứng minh khả năng hoạt động của hệ thống nhận dạng giọng nói đa ngôn ngữ tại một quốc gia mới. Tuy nhiên, để xây dựng hệ thống thực tế, việc thu thập và gán nhãn dữ liệu vẫn là phương án tối ưu.

Multilingual Language Modeling

Simple Multilingual Modeling

Mô hình ngôn ngữ đa ngữ, ở mức độ đơn giản nhất, là việc huấn luyện một mô hình ngôn ngữ trên một lượng lớn dữ liệu từ nhiều ngôn ngữ khác nhau. Ví dụ, khi huấn luyện một mô hình như GPT, bạn chỉ cần đưa tất cả dữ liệu vào, huấn luyện từ vựng con trên toàn bộ dữ liệu và chờ xem kết quả. Tuy nhiên, nếu bạn quan tâm đến hiệu suất, có thể cần làm nhiều hơn thế.

Có hai loại mô hình ngôn ngữ đa ngữ:

- Đầu vào đa ngữ: Nếu bạn có đầu vào đa ngữ, bạn không cần làm gì đặc biệt để mô hình hoạt động, miễn là mô hình cơ bản có khả năng xử lý nhiều ngôn ngữ. Ví dụ, nếu bạn muốn dịch sang tiếng Anh, bạn chỉ cần đưa vào các câu và yêu cầu "please translate this into English", GPT sẽ thực hiện dịch một cách khá tốt mà không cần biết câu đó là tiếng Pháp hay tiếng Nhật.

- Đầu ra đa ngữ: Trong trường hợp này, ít nhất bạn cần chỉ định ngôn ngữ mà mô hình cần tạo ra. Có nhiều cách để thực hiện điều này, nhưng cơ bản là bạn có thể thêm một thẻ hoặc gợi ý về ngôn ngữ đích cho các tác vụ sinh. Ban đầu, Google Dịch đã thực hiện điều này bằng cách thêm một thẻ đơn giản ở đầu câu, ví dụ như "<fr>" hoặc "<ja>", sau đó là câu cần dịch, và mô hình sẽ tạo ra đầu ra tương ứng.

Difficulties in Fully Multilingual Learning

Trong lĩnh vực học đa ngôn ngữ, một trong những thách thức lớn nhất là "Curse of Multilinguality". Hiện tượng này thể hiện rõ khi xem xét các mô hình mã nguồn mở, trong đó đa số chỉ được huấn luyện nghiêm túc với tiếng Anh (ví dụ như LLaMa).

Với một mô hình có kích thước cố định, năng lực xử lý mỗi ngôn ngữ sẽ giảm khi tăng số lượng ngôn ngữ được hỗ trợ. Nghiên cứu từ paper (Conneau et al, 2019) đã chứng minh điều này: khi tăng số lượng ngôn ngữ lên đến 100, điểm số của các ngôn ngữ phổ biến giảm dần. Đối với các ngôn ngữ ít phổ biến, điểm số ban đầu tăng nhẹ nhờ transfer learning từ các ngôn ngữ khác, nhưng sau đó cũng giảm do năng lực mô hình bị giới hạn.

Một ví dụ điển hình khác là dự án Bloom của Hugging Face - một mô hình 175 tỷ tham số được thiết kế cho nhiều ngôn ngữ. Tuy nhiên, kết quả cho thấy mô hình này không hoạt động tốt với tiếng Anh và thậm chí còn kém hơn với các ngôn ngữ khác.

Nguyên nhân chính của hiện tượng này:

- Giới hạn về compute budget: Với nguồn lực tính toán cố định, số lượng dữ liệu có thể xử lý bị hạn chế.
- Thời gian huấn luyện cho mỗi ngôn ngữ tỷ lệ thuận với hiệu suất của ngôn ngữ đó. Khi phân bổ thêm thời gian cho ngôn ngữ mới, thời gian dành cho các ngôn ngữ khác sẽ giảm.

Một chiến lược được đề xuất là phân chia nỗ lực nghiên cứu thành ba hướng:

- Phát triển mô hình chuyên sâu cho tiếng Anh
- Xây dựng mô hình riêng cho khoảng 10 ngôn ngữ phổ biến nhất
- Phát triển mô hình đa ngôn ngữ cho nhiều ngôn ngữ cùng lúc

Trong các mô hình đa ngôn ngữ hiện tại, hầu hết các tham số được chia sẻ giữa các ngôn ngữ, ngoại trừ word embeddings và subword embeddings.

Tokenization Disparity

Một vấn đề quan trọng là sự chênh lệch trong việc token hóa giữa các ngôn ngữ khác nhau. Đặc biệt, khi sử dụng các mô hình như GPT-3.5 và GPT-4 của OpenAI, sự khác biệt này có thể ảnh hưởng đến hiệu suất và chi phí xử lý.

Ví dụ, khi token hóa một đoạn văn bản tiếng Anh bằng tokenizer của GPT-3.5 và GPT-4, kết quả cho ra 58 tokens. Tuy nhiên, khi dịch đoạn văn bản này sang tiếng Myanmar bằng Google Translate và token hóa lại, số lượng tokens tăng lên đến 6117. Nguyên nhân chính là do tiếng Myanmar sử dụng một hệ thống ký tự khác, dẫn đến việc mỗi ký tự được mã hóa thành nhiều bytes, và mỗi byte trở thành một token.

Điều này dẫn đến một số vấn đề:

- Chi phí xử lý văn bản tiếng Myanmar với GPT-4 cao gấp 10 lần so với tiếng Anh, do chi phí được tính theo số lượng tokens.
- Tốc độ sinh văn bản chậm hơn 10 lần, vì mô hình tạo ra tokens với tốc độ cố định.
- Việc không thể nhóm các đơn vị ngữ nghĩa lại với nhau khiến mô hình phải sử dụng nhiều tài nguyên hơn để kết hợp các tokens, dễ dẫn đến lỗi và giảm độ chính xác.

Những vấn đề này ảnh hưởng lớn đến hiệu quả chi phí và độ chính xác của mô hình khi xử lý các ngôn ngữ có hệ thống ký tự khác biệt.

Heuristic Sampling of Data

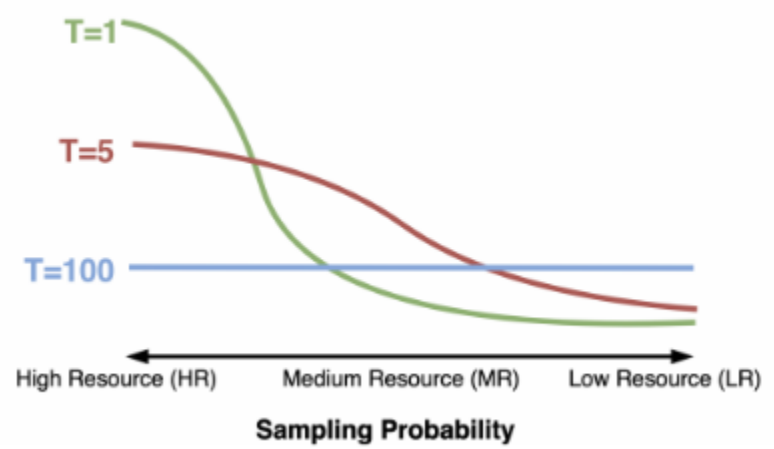
Một trong những giải pháp hiệu quả để cải thiện việc huấn luyện mô hình và xây dựng bộ tokenizer là sử dụng kỹ thuật lấy mẫu dữ liệu, phổ biến nhất là "temperature sampling". Phương pháp này hoạt động bằng cách nhóm dữ liệu (thường theo ngôn ngữ) và lấy mẫu dựa trên tần suất xuất hiện của chúng.

Công thức tính xác suất lấy mẫu cho ngôn ngữ L: $P(L) = e^{freq(L)/T} / \sum_{L'} e^{freq(L')/T}$

Trong đó:

- freq(L): tần suất của ngôn ngữ L
- T: giá trị temperature
- L': tất cả các ngôn ngữ

Với phân phối dữ liệu dạng long-tail, khi:



- T = 5: phân phối sẽ phẳng hơn, tăng tỷ lệ lấy mẫu cho ngôn ngữ ít phổ biến và giảm với ngôn ngữ phổ biến
- T = 100: phân phối gần như đều, các ngôn ngữ được lấy mẫu đồng đều

Temperature sampling được áp dụng ở cả hai giai đoạn:

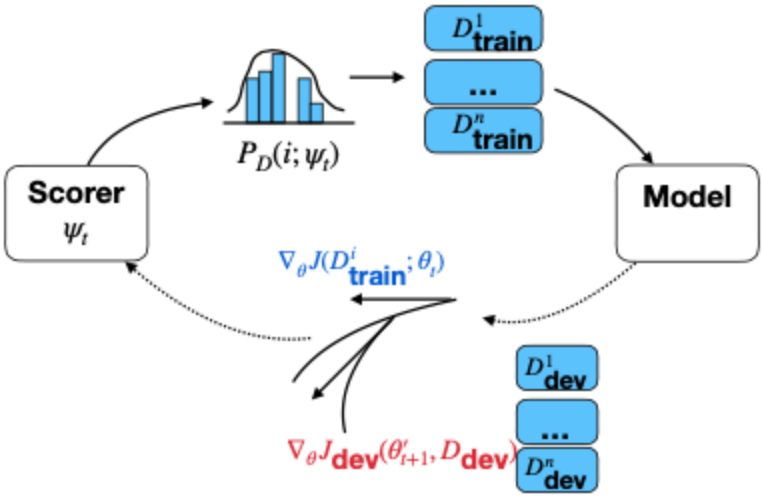
1. Huấn luyện mô hình
2. Xây dựng từ vựng: giảm tỷ trọng tiếng Anh và tăng tỷ trọng các ngôn ngữ ít phổ biến như tiếng Miến Điện

Các mô hình như XLMR và QWEN đã áp dụng phương pháp này. Đặc biệt, QWEN còn mở rộng kích thước từ vựng lên 215K (so với LLaMA là 32K) để cải thiện phân phối cho các ngôn ngữ khác nhau.

Về việc xây dựng batch trong dịch máy, thay vì tạo batch riêng cho từng cặp ngôn ngữ (ví dụ: English-French), người ta thường thực hiện upweighting/sampling trước khi tạo batch. Lý do là nếu mỗi batch chỉ chứa một cặp ngôn ngữ cụ thể, gradient sẽ có độ biến thiên lớn, làm giảm độ ổn định của Stochastic Gradient Descent (SGD). Một cách tiếp cận tốt hơn là tạo nhiều batch đa dạng ngôn ngữ cùng lúc, chạy qua tất cả chúng, sau đó tạo thêm batch mới khi cần thiết.

Learning to Balance Data

Một nghiên cứu thú vị từ Cindy, đã đề xuất một phương pháp học tự động để cân bằng dữ liệu giữa các tập huấn luyện và tập phát triển. Ý tưởng cơ bản là tính toán gradient từ các tập huấn luyện và tập phát triển, sau đó so sánh sự tương đồng giữa chúng. Nếu gradient từ tập huấn luyện và tập phát triển tương đồng, điều đó cho thấy tập huấn luyện đang giúp tối ưu hóa hiệu suất trên tập phát triển. Ngược lại, nếu không tương đồng, tập huấn luyện có thể gây hại cho hiệu suất và cần được điều chỉnh trọng số.



Phương pháp này cho phép tự động học các chiến lược cân bằng dữ liệu phức tạp hơn. Ví dụ, trong giai đoạn đầu của quá trình huấn luyện, có thể cần tăng trọng số cho các ngôn ngữ ít tài nguyên. Tuy nhiên, khi bắt đầu quá tải, trọng số sẽ tự động giảm để tránh gây hại cho mô hình. Khi cần thiết, trọng số sẽ lại được tăng lên.

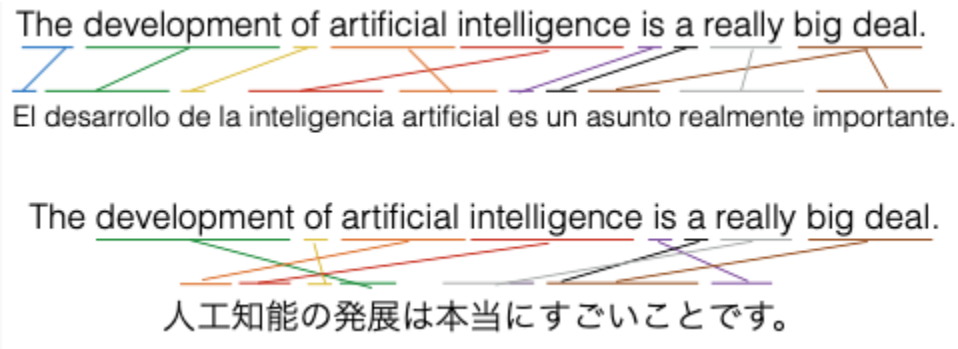
Một số nghiên cứu trước đây, như bài báo NLLB, đã áp dụng chiến lược tương tự bằng cách bắt đầu huấn luyện với các ngôn ngữ có nhiều tài nguyên và dần dần thêm vào các ngôn ngữ ít tài nguyên. Tuy nhiên, đây chỉ là một phương pháp dựa trên kinh nghiệm và có thể có những chiến lược tốt hơn. Việc tự động hóa quá trình này có thể giúp tìm ra các chiến lược tối ưu hơn.

Cuối cùng, một câu hỏi được đặt ra là liệu việc tăng số lượng từ vựng lên 215k có gây ra vấn đề về hiệu suất không. Mặc dù điều này có thể làm chậm quá trình tính toán softmax và embedding từ, nhưng nếu độ dài chuỗi ngắn hơn, ta cũng có thể hưởng lợi từ điều đó. Đặc biệt, khi xử lý dữ liệu đa ngôn ngữ, việc mở rộng từ vựng từ 32k cho tiếng Anh lên 215k cho 100 ngôn ngữ có vẻ là một ý tưởng hợp lý.

Machine Translation

Translation

Trong lĩnh vực dịch máy, một trong những nhiệm vụ đa ngôn ngữ quan trọng nhất là dịch từ ngôn ngữ này sang ngôn ngữ khác. Có hai lý do chính khiến việc dịch thuật trở nên khó khăn. Thứ nhất là sự khác biệt về cú pháp giữa các ngôn ngữ. Ví dụ, câu "the development of artificial intelligence is a really big deal" khi dịch sang tiếng Tây Ban Nha có sự khác biệt về cú pháp. Trong tiếng Tây Ban Nha, cần sử dụng mạo từ mà tiếng Anh không có, và thứ tự của danh từ và tính từ cũng bị đảo ngược. Tiếng Anh thường đặt tính từ trước danh từ, trong khi nhiều ngôn ngữ khác, bao gồm cả tiếng Tây Ban Nha, đặt tính từ sau danh từ. Ngoài ra, cụm từ "big deal" là một biểu đạt thành ngữ và không phải lúc nào cũng được dịch thành "big" và "deal" trong các ngôn ngữ khác.



Khi dịch sang tiếng Nhật, cú pháp còn khác biệt hơn nữa với trật tự chủ ngữ - tân ngữ - động từ, khiến động từ nằm ở cuối câu thay vì giữa câu như trong tiếng Anh. Mặc dù có sự khác biệt lớn về trật tự từ, nhưng đôi khi tiếng Nhật lại có sự tương đồng với tiếng Anh hơn so với tiếng Tây Ban Nha, chẳng hạn như cụm từ "artificial intelligence" vẫn giữ nguyên thứ tự.

Why is it difficult to translate?

Một trong những lý do khiến việc dịch thuật trở nên khó khăn là do sự mơ hồ về mặt từ vựng. Ví dụ, trong tiếng Anh và tiếng Pháp, các từ có thể được dịch khác nhau tùy ngữ cảnh. Chẳng hạn, "a leg of a journey", "a leg of an animal", "a leg of a chair" và "a leg of a human" đều được dịch khác nhau. Một ví dụ khác là từ "run" trong các ngữ cảnh như "run a marathon", "run a program", "run a company" và "a run in a stocking" cũng có cách dịch khác nhau trong hầu hết các ngôn ngữ trên thế giới.

Ngoài ra, còn có những khó khăn đặc thù của từng ngôn ngữ. Chẳng hạn, trong tiếng Nhật, người ta hầu như không đề cập đến chủ ngữ trong giao tiếp hàng ngày. Ví dụ, từ "たべた" (tabeta) có thể có nghĩa là "I ate", "you ate", "he ate", "she ate", "the dog ate", v.v., và không thể xác định được nếu không có ngữ cảnh. Các hệ thống dịch thuật thường gặp khó khăn trong việc xác định ngữ cảnh này. Trong dịch thuật hội thoại, câu hỏi như "Did you eat?" có thể bị dịch thành "I ate" do đó là mặc định, gây ra sự nhầm lẫn. Những đặc thù như vậy làm cho việc dịch thuật trở nên phức tạp hơn.

Translation Tasks

Trong lĩnh vực dịch máy, có một sự kiện quan trọng là WMT (Hội nghị về Dịch máy), nơi diễn ra các nhiệm vụ chia sẻ hàng năm về dịch thuật và đánh giá. Một điểm thú vị là các hệ thống dịch thuật và đánh giá cùng phát triển. Mỗi năm, có hai nhiệm vụ chính: tối ưu hóa độ chính xác của dịch thuật và tối ưu hóa độ chính xác của đánh giá. Nhiệm vụ đánh giá luôn sử dụng các hệ thống từ nhiệm vụ dịch thuật, điều này tạo ra thách thức lớn khi các hệ thống dịch thuật ngày càng cải thiện.

Một nguồn tài nguyên quan trọng khác là FLORES, một tập dữ liệu gồm 200 ngôn ngữ được dịch từ Wikipedia tiếng Anh. Wikipedia là một miền quan trọng cần dịch vì chứa nhiều kiến thức hữu ích. Tuy nhiên, việc có được bản dịch chất lượng cao cho 200 ngôn ngữ là rất khó khăn do

khó khăn trong việc thuê dịch giả giỏi và kiểm soát chất lượng. Tập dữ liệu này được tạo bởi Meta và là tiêu chuẩn cho dịch thuật ngôn ngữ ít tài nguyên.

Cuối cùng, IWSLT cung cấp các nhiệm vụ về dịch thuật giọng nói, là một lựa chọn tốt nếu bạn quan tâm đến lĩnh vực này.

NLLB Translation Model (NLLB Team 2022)

Trong lĩnh vực dịch máy, mô hình NLLB translation là một ví dụ điển hình về việc xây dựng một mô hình dịch máy mạnh mẽ. Đây là một mô hình mã nguồn mở, cung cấp chi tiết về cách thức hoạt động của nó. Để tóm tắt, mô hình này bắt đầu với dữ liệu song ngữ công khai và một lượng nhỏ dữ liệu song ngữ ban đầu, cùng với rất nhiều dữ liệu đơn ngữ.

Quá Trình Huấn Luyện

- Mô Hình Nhúng Đa Ngữ: Quá trình huấn luyện bắt đầu bằng việc xây dựng một mô hình embedding đa ngữ, với mục tiêu xác định các cặp dịch tốt. Mô hình này được áp dụng trên toàn bộ dữ liệu đơn ngữ từ nhiều ngôn ngữ để trích xuất "mind bitext" - dữ liệu song ngữ với điểm tin cậy về chất lượng.

- Nhận Diện Ngôn Ngữ: Một thách thức lớn trong quá trình này là nhận diện ngôn ngữ, đặc biệt khi xử lý tới 200 ngôn ngữ có thể rất giống nhau. Ví dụ, một hệ thống nhận diện ngôn ngữ đã dự đoán sai ngôn ngữ của một chuỗi ký tự emoji là "manip". Để khắc phục, cần có dữ liệu huấn luyện được chọn lọc kỹ lưỡng và các chỉ số tin cậy cho hệ thống nhận diện ngôn ngữ.

Kỹ Thuật Mô Hình

- Mixture of Experts: Mô hình NLLB sử dụng kỹ thuật "mixture of experts", cho phép sử dụng các tham số cụ thể cho từng ngôn ngữ. Điều này đặc biệt hữu ích khi xử lý nhiều ngôn ngữ khác nhau, vì có thể tối ưu hóa các tham số cho từng ngôn ngữ cụ thể.

- Curriculum Learning: Kỹ thuật "curriculum learning" được áp dụng bằng cách bắt đầu huấn luyện với các ngôn ngữ có tài nguyên thấp. Điều này giúp mô hình dần dần thích nghi và cải thiện khả năng dịch cho các ngôn ngữ ít phổ biến.

- Huấn Luyện Tự Giám Sát: Mô hình áp dụng huấn luyện tự giám sát với mục tiêu khử nhiễu, bằng cách thêm nhiễu vào dữ liệu đơn ngữ và sau đó cố gắng tái tạo lại dữ liệu gốc. Điều này giúp cải thiện khả năng của mô hình trong việc xử lý dữ liệu không hoàn hảo.

- Back Translation: Kỹ thuật "back translation" được sử dụng để tạo dữ liệu song ngữ giả lập từ dữ liệu đơn ngữ. Ví dụ, nếu muốn dịch từ tiếng Anh sang tiếng Swahili nhưng không có nhiều dữ liệu song ngữ, có thể huấn luyện một mô hình dịch ngược từ Swahili sang tiếng Anh và tạo ra dữ liệu song ngữ giả lập.

Mô hình NLLB đã được đánh giá và cho thấy hiệu quả cao, đặc biệt là với các ngôn ngữ có tài nguyên thấp, cạnh tranh với các mô hình GPT. Tuy nhiên, với các ngôn ngữ có tài nguyên cao, Google Translate và GPT-4 vẫn có ưu thế hơn. Ví dụ, GPT-4 có thể vượt qua Google Translate ở một số ngôn ngữ như tiếng Romania.

Ngoài ra, còn có mô hình SeamlessM4T hỗ trợ dịch giọng nói và Lego MT, cung cấp thêm lựa chọn cho những ai muốn khám phá các mô hình dịch máy tiên tiến.

Multilingual Pre-trained Models

Multilinguality of Standard LLMs

Trong phần này, tôi muốn thảo luận về các mô hình tiền huấn luyện đa ngôn ngữ. Các mô hình ngôn ngữ lớn (LLMs) đóng như GPT-4 thường có khả năng đa ngôn ngữ một cách ngẫu nhiên do dữ liệu huấn luyện lớn. Ngược lại, các LLMs mở thường thực hiện lọc dữ liệu để đảm bảo hiệu suất tốt cho tiếng Anh, như đã đề cập trước đó. Do đó, hiện tại không có nhiều lựa chọn mở thực sự tốt cho các mô hình tự hồi quy từ trái sang phải tiêu chuẩn. Tôi cho rằng, ở thời điểm hiện tại, "Qwen" là lựa chọn tốt nhất.

Multi-lingual Representation Learning

Tiếp theo, chúng ta sẽ khám phá về các mô hình học biểu diễn đa ngôn ngữ và các mô hình mã hóa-giải mã. Khác với tiếng Anh, nơi mà các mô hình tự hồi quy (Auto regressive) thường được đánh giá cao, các mô hình mã hóa-giải mã và mô hình ngôn ngữ mặt nạ (Mask language models) lại tỏ ra khá cạnh tranh trong các nhiệm vụ đa ngôn ngữ.

Việc tiền huấn luyện mô hình ngôn ngữ, chẳng hạn như BERT, đã được chứng minh là hiệu quả và sử dụng các mục tiêu mô hình ngôn ngữ mặt nạ. Các mô hình như mBERT, XLM và XLM-R mở rộng phương pháp huấn luyện theo phong cách BERT cho việc tiền huấn luyện đa ngôn ngữ. Trước khi đi sâu vào cách thức mà các mô hình này thực hiện điều đó, chúng ta cần hiểu rõ hơn về các khái niệm cơ bản và cách chúng được áp dụng trong bối cảnh đa ngôn ngữ.

Multilingual Representation Evaluation

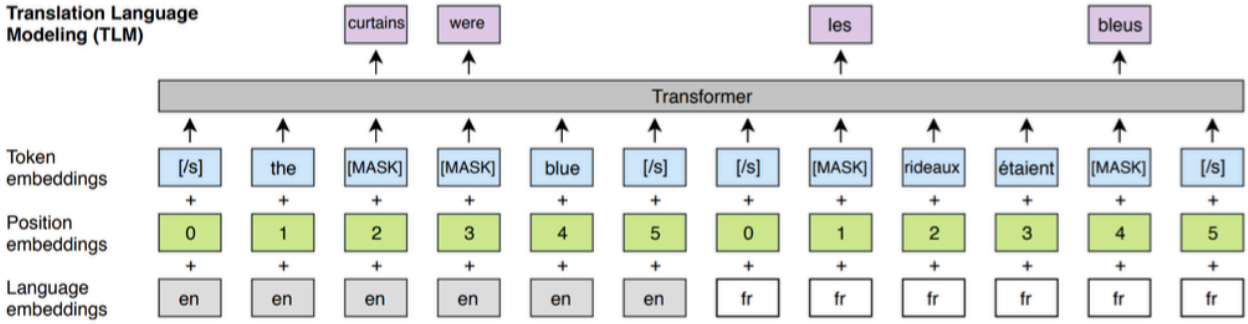
Trong phần này, chúng ta tìm hiểu về việc đánh giá các biểu diễn đa ngôn ngữ và một số bộ tiêu chuẩn đánh giá phổ biến mà mọi người thường sử dụng để đo lường kỹ năng của các mô hình đa ngôn ngữ.

Đầu tiên là bộ tiêu chuẩn "XTREME" và phiên bản nâng cao "XTREME-R". Cách hoạt động của chúng là bao gồm các nhiệm vụ như phân loại câu, dự đoán cấu trúc, truy xuất câu và trả lời câu hỏi trên một loạt các ngôn ngữ đa dạng về mặt ngữ hệ học. Cụ thể, có thể lên đến khoảng 40 ngôn ngữ khác nhau.

Ngoài ra, còn có bộ tiêu chuẩn "XGLUE", tuy ít đa dạng về mặt ngữ hệ học hơn nhưng lại bao gồm các nhiệm vụ theo phong cách tạo sinh.

Multilingual Masked Language Modeling (Lample and Conneau 2019)

Một phương pháp phổ biến để huấn luyện mô hình ngôn ngữ đa ngữ là sử dụng "multilingual mask language modeling". Khác với phương pháp "mask language modeling" truyền thống, nơi mà bạn che đi một phần của đầu vào và cố gắng dự đoán phần bị che đó, phương pháp đa ngữ này hoạt động bằng cách đưa vào mô hình một câu ở một ngôn ngữ này và một câu khác ở ngôn ngữ khác, lý tưởng nhất là các câu song song.



Quá trình này bao gồm việc che đi một số phần của câu, cho phép mã nguồn huấn luyện gần như hoàn toàn giống nhau, chỉ cần thêm các "language embeddings" để phân biệt ngôn ngữ. Mô hình được huấn luyện để có thể dự đoán giữa các ngôn ngữ, từ đó cải thiện sự liên kết giữa các biểu diễn ngôn ngữ khác nhau. Nhiều nghiên cứu đã chỉ ra rằng phương pháp này rất hiệu quả trong việc học các biểu diễn ngôn ngữ.

Multilingual Encoder-decoder

mT5 là một trong những mô hình tốt nhất hiện nay cho xử lý ngôn ngữ đa ngữ, dựa trên kiến trúc T5, một kiến trúc encoder-decoder với mục tiêu tái tạo dữ liệu bị che giấu (masked reconstruction objective).

Cách hoạt động của mô hình này là sử dụng một mô hình encoder-decoder để nhận đầu vào và thực hiện các biến đổi trên đầu ra, như là "dropping words" (bỏ từ) và "reordering words" (sắp xếp lại từ). Mục tiêu là để mô hình tái tạo lại đầu ra gốc. Mô hình này được huấn luyện trên nhiều ngôn ngữ khác nhau, mang lại hiệu suất cao cho nhiều tác vụ khác nhau, đặc biệt là các tác vụ ít phổ biến hơn so với các mô hình ngôn ngữ tiêu chuẩn.

Ngoài mT5, còn có các phiên bản khác như mT0 và ByT5. mT0 được tinh chỉnh thêm với dữ liệu hướng dẫn (instruction tuned), trong khi ByT5 được huấn luyện ở mức độ byte, không sử dụng tokenization, cho phép mô hình hóa bất kỳ hệ chữ nào mà không gặp vấn đề với Unicode. Tuy nhiên, theo kinh nghiệm cá nhân, mT5 dễ sử dụng hơn và cho kết quả tốt hơn so với ByT5.

Gần đây, có một phiên bản tinh chỉnh của mT5, được huấn luyện với một lượng lớn dữ liệu hướng dẫn, tạo ra một phiên bản hiện đại hơn của mT0. Mặc dù chưa có nhiều trải nghiệm cá nhân với phiên bản này, nhưng nó được cho là có khả năng “instruction following” và thực hiện các tác vụ khác khá tốt.

Advanced Modeling Strategies

Cross-lingual Transfer Learning

Trong phần này, tôi muốn thảo luận một chút về một số chiến lược mô hình hóa nâng cao hơn. Đầu tiên là về học chuyển giao đa ngôn ngữ (crosslingual transfer learning), phương pháp này tận dụng dữ liệu từ một hoặc nhiều ngôn ngữ nguồn có tài nguyên phong phú.

Một kỹ thuật khác mà chúng ta thường sử dụng là tiền huấn luyện và tinh chỉnh (pre-training and fine-tuning). Tiền huấn luyện và tinh chỉnh rất hữu ích nếu bạn muốn mô hình của mình hoạt động tốt với một ngôn ngữ cụ thể, vì nó cho phép bạn chuyên biệt hóa cho ngôn ngữ đó. Như đã đề cập, với "the curse of multilinguality", rất khó để có một mô hình thực sự tốt ở tất cả các ngôn ngữ.

Ngoài ra, còn có phương pháp “zero shot transfer”. Cuối cùng, có một phương pháp gọi là “annotation projection” hoặc trong một số tài liệu còn gọi là "translate train".

Pre-train and Fine-tune

Có hai bước quan trọng là "pre-train" và "fine-tune". Cơ chế hoạt động của chúng như sau: đầu tiên, mô hình được huấn luyện trên một lượng lớn dữ liệu từ nhiều ngôn ngữ khác nhau. Sau đó, mô hình được "fine-tune" trên dữ liệu của một ngôn ngữ cụ thể khác.

Phương pháp này thường mang lại kết quả tốt, đặc biệt là khi trong tập dữ liệu huấn luyện ban đầu có ít nhất một số dữ liệu từ ngôn ngữ mà bạn muốn huấn luyện. Ngược lại, nếu không có dữ liệu nào từ ngôn ngữ mục tiêu trong tập dữ liệu huấn luyện ban đầu, việc đạt được hiệu quả tốt sẽ khá khó khăn, vì mô hình chưa có kiến thức nền tảng về ngôn ngữ đó.

Similar Language Regularization

Một trong những thách thức khi làm việc với các ngôn ngữ ít tài nguyên là sự thiếu hụt dữ liệu, đặc biệt là dữ liệu có giám sát. Để khắc phục điều này, một phương pháp là huấn luyện mô hình trên chính ngôn ngữ đó cùng với một vài ngôn ngữ khác có liên quan chặt chẽ.

Ví dụ, nếu bạn muốn huấn luyện mô hình cho một ngôn ngữ cụ thể từ Ấn Độ, thường có một vài ngôn ngữ khác từ Ấn Độ có thể có tài nguyên phong phú hơn. Chẳng hạn, "Hindi" có thể là một ngôn ngữ có nhiều tài nguyên hơn so với ngôn ngữ bạn đang quan tâm. Bạn có thể áp dụng phương pháp này cho bất kỳ ngôn ngữ nào.

Tuy nhiên, đối với một số ngôn ngữ, không có ngôn ngữ nào khác có tài nguyên phong phú hơn, điều này làm cho việc huấn luyện trở nên khó khăn hơn. Nhưng khi có ngôn ngữ liên quan với tài nguyên phong phú, bạn có thể tận dụng điều đó để cải thiện hiệu quả huấn luyện.

Meta-learning for multilingual training

Một phương pháp tiếp cận thú vị để giải quyết vấn đề các ngôn ngữ ít tài nguyên là sử dụng "meta learning". Thay vì áp dụng cách học đa ngôn ngữ tiêu chuẩn, nơi bạn phát triển một mô hình có khả năng xử lý nhiều ngôn ngữ và sau đó tinh chỉnh nó cho ngôn ngữ ít tài nguyên, phương pháp này tập trung vào việc học một mô hình có khả năng thích ứng tốt với các ngôn ngữ ít tài nguyên ngay từ đầu.

Meta learning giúp tối ưu hóa quá trình này bằng cách sử dụng một tập dữ liệu phát triển (development set) để điều chỉnh các mô hình sao cho các gradient từ các cập nhật này phù hợp với gradient trên dữ liệu mục tiêu. Nói cách khác, bạn đang cố gắng học cách cập nhật mô hình theo hướng có lợi cho việc tinh chỉnh đối với ngôn ngữ ít tài nguyên khi bắt đầu huấn luyện trên nó.

Mặc dù không đi sâu vào chi tiết của meta learning trong bài viết này, nhưng nếu bạn đã từng nghe về nó, bạn sẽ hiểu rằng đây là một phương pháp học tập mà mô hình được thiết kế để cải thiện khả năng thích ứng và hiệu quả khi chuyển đổi sang các ngữ cảnh hoặc ngôn ngữ mới.

Zero-shot transfer for pretrained representations

Một trong những chủ đề lớn là "zero-shot transfer" cho các biểu diễn đã được tiền huấn luyện. Phương pháp này hoạt động như sau: đầu tiên, bạn tiền huấn luyện một mô hình ngôn ngữ lớn sử dụng dữ liệu đơn ngữ từ nhiều ngôn ngữ khác nhau. Sau đó, bạn tinh chỉnh mô hình này bằng dữ liệu đã được chú thích trong một ngôn ngữ cụ thể, chẳng hạn như tiếng Anh. Cuối cùng, bạn kiểm tra mô hình trên một ngôn ngữ khác với ngôn ngữ đã tinh chỉnh, ví dụ như tiếng Pháp.

Lợi ích của phương pháp này là tiền huấn luyện đa ngôn ngữ có thể học được một dạng biểu diễn không hẳn là "universal representation" nhưng ít nhất là một biểu diễn có khả năng chuyển giao giữa các ngôn ngữ.

Annotation Projection

Trong lĩnh vực dịch máy đa ngôn ngữ, có một số chiến lược có thể cho kết quả tốt, như “Translate-Train” và “annotation projection”.

Phương pháp “Translate-Train” này rất đơn giản:

- Bạn có dữ liệu đã được gắn nhãn ở ngôn ngữ A (ví dụ tiếng Anh).
- Bạn dịch dữ liệu này sang ngôn ngữ B (ví dụ tiếng Swahili).
- Bây giờ bạn có dữ liệu gắn nhãn ở tiếng Swahili để huấn luyện mô hình.

Mặc dù có thể có một số lỗi dịch, nhưng việc có được ít dữ liệu gắn nhãn vẫn tốt hơn là không có gì cả. Đặc biệt khi chất lượng dịch máy ngày càng được cải thiện.

Phương pháp “annotation projection” phức tạp hơn:

- Bạn có dữ liệu song ngữ (ví dụ tiếng Anh và Swahili) với các nhãn (ví dụ nhãn từ loại) ở ngôn ngữ A (tiếng Anh).
- Bạn cần tìm sự liên kết (alignment) giữa các từ trong hai ngôn ngữ.
- Sau đó, bạn có thể "chiếu" các nhãn từ tiếng Anh sang tiếng Swahili dựa trên sự liên kết này.

Việc tìm sự liên kết từ liên ngôn ngữ (word alignment) rất quan trọng ở đây. Có hai phương pháp chính:

- Không giám sát (unsupervised): Sử dụng thống kê về sự đồng xuất hiện của từ giữa các ngôn ngữ, ví dụ như GIZA++.
- Có giám sát (supervised): Sử dụng mô hình học có tập dữ liệu gắn nhãn, ví dụ như một mô hình Transformer đa ngôn ngữ.

“Annotation projection” có thể được áp dụng cho nhiều loại nhãn khác nhau, như nhận dạng thực thể, phân loại văn bản, v.v. Đây là một giải pháp hữu ích khi không có đủ dữ liệu gán nhãn trực tiếp cho ngôn ngữ mục tiêu.

Một phương pháp khác là dịch trước khi dự đoán (Translate-Test):

- Dịch văn bản đầu vào (ví dụ tiếng Nhật) sang ngôn ngữ cơ sở (ví dụ tiếng Anh).
- Sử dụng mô hình dự đoán dựa trên ngôn ngữ cơ sở.
- Sau đó dịch kết quả dự đoán trở lại ngôn ngữ đầu vào.

Mặc dù không hiệu quả bằng một hệ thống đa ngôn ngữ tốt, phương pháp này vẫn có thể cung cấp kết quả tốt hơn so với một hệ thống đa ngôn ngữ kém.

Which Language to Use?

Trong nghiên cứu về việc lựa chọn ngôn ngữ để chuyển giao mô hình, một phát hiện thú vị đã được đưa ra. Khi muốn huấn luyện một mô hình ngôn ngữ tốt cho một ngôn ngữ đích, việc chọn ngôn ngữ nguồn phù hợp là rất quan trọng.

Nghiên cứu (Lin et al. 2019) đã chỉ ra rằng yếu tố hữu ích nhất để dự đoán ngôn ngữ nào sẽ là tốt nhất để chuyển giao là khoảng cách địa lý giữa các ngôn ngữ. Mặc dù điều này có vẻ kỳ lạ, vì sự gần gũi về mặt địa lý không nhất thiết đồng nghĩa với sự tương đồng về ngôn ngữ, nhưng thực tế cho thấy các ngôn ngữ gần nhau thường có sự tương đồng về từ vựng và cú pháp.

Ví dụ, tiếng Basque và tiếng Tây Ban Nha rất khác nhau, nhưng nhìn chung, các ngôn ngữ gần nhau trên bản đồ có xu hướng tương đồng hơn, làm cho chúng trở thành lựa chọn tốt cho việc chuyển giao ngôn ngữ.

What if language don’t share the same script?

Việc chuyển đổi giữa các ngôn ngữ có thể gặp khó khăn khi chúng không sử dụng cùng một hệ chữ viết. Một ví dụ điển hình là việc "pivoting" từ tiếng Marathi sang tiếng Hindi và sau đó sang

một ngôn ngữ khác để liên kết các thực thể. Một kỹ thuật hữu ích là sử dụng Bảng chữ cái ngữ âm quốc tế (IPA) để chuẩn hóa cách phát âm. Mặc dù các ngôn ngữ có thể sử dụng hệ chữ viết khác nhau, nhưng khi chuẩn hóa qua IPA, cách phát âm có thể tương đồng trong nhiều ngôn ngữ liên quan.

Tuy nhiên, cần lưu ý rằng việc chuẩn hóa này có thể làm giảm độ chính xác trong một số ngôn ngữ. Ví dụ, đối với tiếng Anh và tiếng Pháp, dù người học có thể đọc được các ký tự, nhưng cách phát âm lại khác biệt đáng kể. Điều này cho thấy rằng không thể chỉ dựa vào cách phát âm để chuẩn hóa và mong đợi kết quả tốt. Do đó, việc sử dụng IPA là một kỹ thuật hữu ích nhưng cần được áp dụng cẩn thận.

How to Share Parameters?

Có nhiều kỹ thuật để chia sẻ tham số giữa các ngôn ngữ. Một phương pháp là chia sẻ toàn bộ tham số, tức là sử dụng một mô hình duy nhất với tất cả tham số giống nhau. Trước đây, có những phương pháp chỉ chia sẻ một phần như bộ mã hóa (encoder) hoặc cơ chế chú ý (attention mechanism), cũng như một số ma trận của mô hình Transformer.

Một cách tiếp cận khác là sử dụng bộ tạo tham số để tạo ra tham số cho từng ngôn ngữ. Mặc dù tôi thích bài báo này (Platonios et al. 2018), nhưng nó không thực sự thực tiễn. Cụ thể, chúng tôi đã sử dụng mạng nơ-ron để tạo ra tham số cho mô hình đa ngôn ngữ và cung cấp thông tin như loại ngôn ngữ. Tuy nhiên, phương pháp này khá tham vọng và đòi hỏi nhiều tham số, do đó không thực sự khả thi trong thực tế.

Language Experts

Một xu hướng phổ biến hiện nay là sử dụng các "language experts" hoặc "adapters". "Language experts" là một lớp được chèn vào một phần cụ thể của mô hình, hoạt động theo kiểu "adapter style" để tối ưu hóa việc huấn luyện với ít tham số hơn. Cơ chế này bao gồm việc "down sample" và "up sample", tương tự như "LoRA" hoặc một adapter khác, giúp giảm số lượng tham số cần thiết cho từng ngôn ngữ.

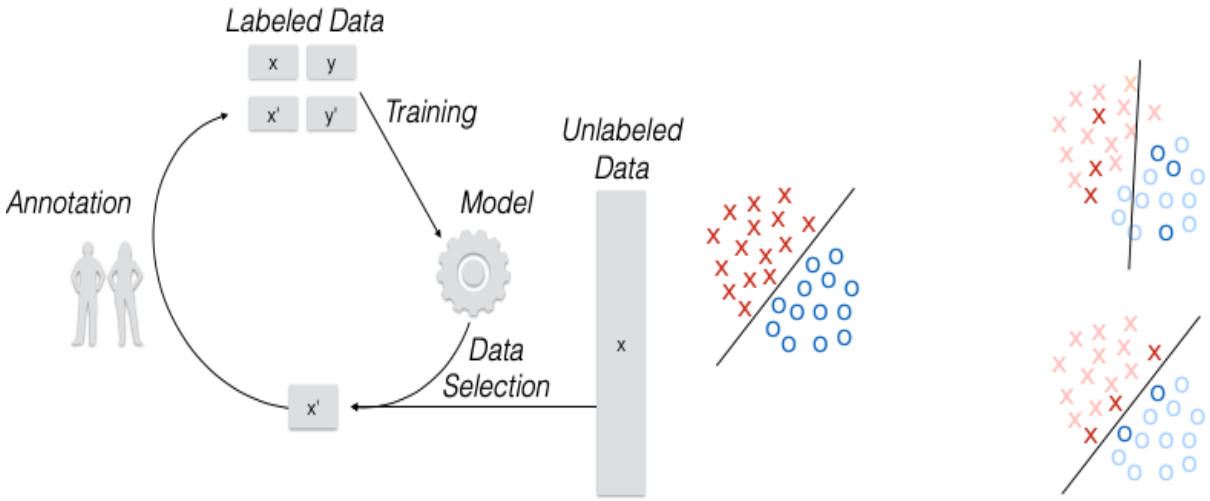
Ngoài ra, còn có các adapter dựa trên nhiệm vụ, cho phép điều chỉnh mô hình theo từng nhiệm vụ cụ thể. Một điểm đáng chú ý là khả năng tiền huấn luyện các mô hình với tham số ngôn ngữ cụ thể đã được tích hợp sẵn. Trong nghiên cứu về tóm tắt văn bản, chúng tôi đã so sánh "prefix tuning" và "LoRA", và nhận thấy rằng có thể huấn luyện một mô hình duy nhất nhưng mỗi ngôn ngữ có tham số "prefix tuning" hoặc "LoRA" riêng. Điều này tỏ ra khá hiệu quả trong việc cải thiện khả năng của mô hình.

Creating New Data

Active Learning Pipeline

Trong NLP, một thách thức lớn là thu thập dữ liệu cho các ngôn ngữ ít tài nguyên. Một giải pháp là sử dụng phương pháp "Active Learning".

Ý tưởng cơ bản của Active Learning là sử dụng dữ liệu đã được gán nhãn để huấn luyện một mô hình. Sau đó, áp dụng mô hình này lên một lượng lớn dữ liệu chưa được gán nhãn và chọn ra những dữ liệu mà mô hình tỏ ra không chắc chắn nhất để tiến hành gán nhãn.



Phương pháp này giúp chọn lọc dữ liệu mà mô hình hiện tại chưa tự tin hoặc chưa hoạt động tốt, từ đó cải thiện chất lượng mô hình. Active Learning không chỉ giới hạn trong học đa ngữ mà còn đặc biệt hữu ích khi chỉ có thể gán nhãn một lượng dữ liệu nhỏ.

Why Active Learning?

Ý tưởng cơ bản được minh họa ở hình trên cho bài toán phân loại nhị phân. Nếu bạn chỉ gán nhãn dữ liệu một cách ngẫu nhiên, có thể bạn sẽ thu được dữ liệu không cung cấp thông tin rõ ràng về ranh giới quyết định cụ thể. Kết quả là, mô hình được huấn luyện trên một số điểm dữ liệu ngẫu nhiên có thể không chính xác. Ngược lại, Active Learning tìm kiếm dữ liệu trực tiếp trên ranh giới quyết định, giúp bạn thu thập các mẫu hiệu quả hơn.

Fundamental Ideas

Có hai ý tưởng cơ bản là sự không chắc chắn (uncertainty) và tính đại diện (representativeness). Khi xây dựng mô hình, chúng ta cần chọn dữ liệu mà mô hình cảm thấy không chắc chắn nhưng vẫn mang tính đại diện. Việc chỉ chọn dữ liệu dựa trên tính đại diện có thể hữu ích, ví dụ như chọn các cụm từ có tần suất cao trong dịch máy để cải thiện độ bao phủ của các cụm từ này.

Tuy nhiên, sự không chắc chắn cũng rất quan trọng vì nó giúp phát hiện ra những điểm mù hiện tại của mô hình. Vấn đề khi chỉ dựa vào sự không chắc chắn là có thể dẫn đến việc chọn phải nhiều dữ liệu không hữu ích, chẳng hạn như các câu chỉ chứa emoji trong dịch máy, điều này không thực sự có ích cho việc huấn luyện mô hình.

Tính đại diện có nghĩa là bạn cần đảm bảo rằng không gian embedding của bạn được bao phủ một cách đầy đủ cho tất cả các embedding mà bạn có trong mô hình.

Resources

1. https://phontron.com/class/anlp2024/lectures/#multilingual-nlp-april-09
2. Reference: Google's Multilingual Translation System (Johnson et al. 2016)
3. Reference: Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT (Wu and Dredze 2019)
4. Reference: Unsupervised Cross-lingual Representation Learning at Scale (Conneau et al. 2019)
5. Reference: Massively Multilingual NMT (Aharoni et al. 2019)
6. Reference: Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges (Arivazhagan et al. 2019)
7. Reference: Balancing Training for Multilingual Neural Machine Translation (Wang et al. 2020)
8. Reference: Multi-task Learning for Multiple Language Translation (Dong et al. 2015)
9. Reference: Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism (Firat et al. 2016)
10. Reference: NLLB (NLLB Team 2022)
11. Reference: Parameter Sharing Methods for Multilingual Self-Attentional Translation Models (Sachan and Neubig 2018)
12. Reference: Contextual Parameter Generation for Universal Neural Machine Translation (Platanios et al. 2018)
13. Reference: MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer (Pfeiffer et al. 2020)
14. Reference: Pre-training Multilingual Experts (Pfeiffer et al. 2022)
15. Reference: Cross-lingual Language Model Pretraining (Lample and Conneau 2019)
16. Reference: Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks (Huang et al. 2019)
17. Reference: Explicit Alignment Objectives (Hu et al. 2020)
18. Reference: XTREME (Hu et al. 2020)
19. Reference: XGLUE (Liang et al. 2020)
20. Reference: XTREME-R (Ruder et al. 2021)
21. Reference: Rapida Adaptation to New Languages (Neubig and Hu 2018)
22. Reference: Meta-learning for Low-resource Translation (Gu et al. 2018)
23. Reference: How multilingual is Multilingual BERT? (Pires et al. 2019)
24. Reference: Inducing Multilingual Text Analysis Tools via Robust Projection across Aligned Corpora (Yarowsky et al. 2001)
25. Reference: Choosing Transfer Languages for Cross-Lingual Learning (Lin et al. 2019)
26. Reference: Phonological Transfer for Entity Linking (Rijhwani et al. 2019)
27. Reference: Handling Syntactic Divergence (Zhou et al. 2019)
28. Reference: Support Vector Machine Active Learning with Applications to Text Classification (Tong and Koller 2001)

29. Reference: Reducing labeling effort for structured prediction tasks (Culotta and McCallum 2005)
30. Reference: Active Learning for Convolutional Neural Networks: A Core-Set Approach (Sener and Savarese 2017)
31. Reference: A Little Annotation does a Lot of Good: A Study in Bootstrapping Low-resource Named Entity Recognizers (Chaudhary et al. 2019)