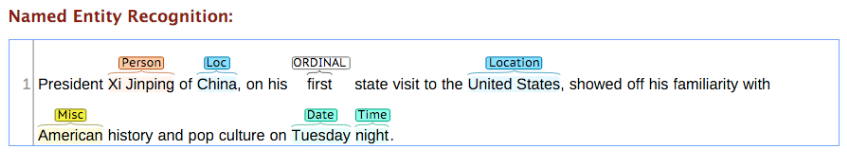
1. **Tổng quan về Named Entity Recognition (NER )**

* Nhận dạng thực thể có tên: là tiến trình xác định và phân loại các phần tử trong văn bản vào các danh mục được định nghĩa trước như tên người, tên tổ chức, địa điểm, giá trị tiền tệ, tỷ lệ phần trăm,… [1]



* Đây là bài toán con được xem là tiền xử lý cho bài toán phức tạp hơn là Relation Extraction, Question answering, Search engine entity-oriented. [2]
* Một công cụ NER tốt có thể xác định các loại thực thể:
* Thực thể định danh: People, Organization, Location, Geo-political entity, Facility, Vehicle.
* Thực thể số: Money, Number
* Thực thể về thời gian: Time, Date
* Mã ký hiệu các lớp trong NER:
* IO encoding: xác định một token là một thực thể xác định hay không là phải thực thể.
* BIO encoding: thêm ký tự B hoặc I trước nhãn để xác định token bắt đầu hoặc bên trong thực thể, dùng ký tự O để xác định các token không thuộc thực thể.
* BMEWO encoding: tương tự như BIO encoding nhưng dùng ký tự M và E để phân biệt token nằm giữa và token kết thúc thực thể, và ký tự W để chỉ những thực thể chỉ gồm một token.
* …

Ví dụ: ta có chuỗi ban đầu: “Steven Paul Jobs was born in California.”

Sau khi thực hiện NER, các token trong câu sẽ được gán nhãn như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| **Token** | **IO** | **BIO** | **BMEWO** |
| Steven | PER | B-PER | B-PER |
| Paul | PER | I-PER | M-PER |
| Jobs | PER | I-PER | E-PER |
| was | O | O | O |
| born | O | O | O |
| in | O | O | O |
| California | LOC | B-LOC | W-LOC |
| . | O | O | O |

* Sự khó khăn phân loại trong NER:
* Nhiều thực thể hiếm khi xuất hiện nên hệ thống chỉ có thể xác định chúng dựa trên nội dung.
* Trong thực tế, một từ có thể thuộc nhiều loại thực thể khác nhau, vì vậy có nhiều sự nhập nhằng trong việc phân loại thực thể cho các từ:
  + People và Organization: Brigham Young, Ford, Philip Morris…
  + People và Location: Jordan, Washington, Georgia,…
  + People và Time: April, June, Sunday,…
  + …

1. **NER trong Stanford CoreNLP C#**

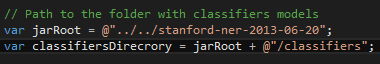
Stanford CoreNLP hỗ trợ người dùng thực hiện giai đoạn NER thông qua lớp CRFClassifier. Lớp này được cài đặt dựa trên các mô hình trình tự chuỗi tuyến tính Conditional Random Field.

Stanford NER cung cấp cho ta 3 model để thực hiện phân loại các phần tử trong văn bản:

* Mô hình 3 lớp: Location, Person, Organization
* Mô hình 4 lớp: Location, Person, Organization, Misc
* Mô hình 7 lớp: Location, Person, Organization, Money, Percentage, Date, Time.

Ta có download các mô hình này tại: <https://nlp.stanford.edu/software/CRF-NER.html>

Để sử dụng lớp CRFClassifier, ta khai báo một biến lưu trữ chuỗi đường dẫn file của mô hình phân lớp:



Sau đó, ta tạo một mô hình bằng mô hình trong đường dẫn trên:



Cuối cùng, ta thực hiện phân lớp các từ của văn bản bằng phương thức *classifyToString( )* của lớp CRFClassifier:



Kết quả hàm trả về một chuỗi các từ đã được gán nhãn lớp.



Với chuỗi kết quả trả về như trên:

* CRF được phân loại vào lớp ORGANIZATION
* Jenny Finkel được phân vào lớp PERSON
* Những từ còn lại không thuộc 3 lớp được định nghĩa trước của mô hình (PERSON, ORGANIZATION, LOCATION) nên được đánh nhãn là O ( Other)

Ngoài ra, lớp CRFClassifier còn cung cấp một số hàm phân lớp với kết quả trả về nhiều định dạng khác:

|  |  |
| --- | --- |
| **Cú pháp hàm** | **Kết quả trả về** |
| String classifyToString ([String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) sentences, [String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) outputFormat, boolean preserveSpacing) | Tùy thuộc vào biến outputFormat. Có 5 kiểu định dạng: "slashTags", "xml", "inlineXML", "tsv", or "tabbedEntities" |
| [String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) classifyWithInlineXML  ([String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) sentences) | Các từ được phân lớp dưới dạng inline XML |
| [String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) classifyToString  ([String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) sentences) | Các từ được phân lớp bằng dấu / |
| [List](https://docs.oracle.com/javase/8/docs/api/java/util/List.html?is-external=true)<[Triple](https://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/util/Triple.html)<[String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true),[Integer](https://docs.oracle.com/javase/8/docs/api/java/lang/Integer.html?is-external=true),[Integer](https://docs.oracle.com/javase/8/docs/api/java/lang/Integer.html?is-external=true)>>  classifyToCharacterOffsets  ([String](https://docs.oracle.com/javase/8/docs/api/java/lang/String.html?is-external=true) sentences) | Trả về danh sách các Triple, mỗi triple chứa thông tin về lớp của thực thể, vị trí đầu tiên và vị trí kết thúc thực thể trong chuỗi ban đầu. |

Tài liệu tham khảo:

[1]: <https://ongxuanhong.wordpress.com/2016/02/05/cac-thuat-ngu-trong-xu-ly-ngon-ngu-tu-nhien/>

[2] <https://ongxuanhong.wordpress.com/2017/08/28/information-extraction-bai-toan-rut-trich-thong-tin-trong-van-ban/#more-13401>

[3] <https://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/ie/AbstractSequenceClassifier.html>

[4] <http://www.surdeanu.info/mihai/teaching/ista555-spring15/materials/lecture2.pdf>