Bài tập thực hành:

Tiền xử lý với OpenNLP

Thời gian làm bài: 1 tuần (Xem dealine trong link nộp bài trên moodle)

Nộp bài:

- Nộp bài lên moodle.
- Đặt tên bài nộp theo định dạng MSSV.rar. Trong đó bao gồm:
  - Tập tin báo cáo.
  - Các tập tin dữ liệu theo yêu cầu của bài tập
- Nếu sử dụng code trên Internet, trước đoạn code sử dụng phải có chú thích đường dẫn đến trang web chứa đoạn code. Trường hợp 2 bài có đoạn code giống nhau, bài không chú thích đường dẫn đến trang web chứa source code sẽ bị tính là gian lận, và bị 0 điểm.

Các hành vi sử dụng toàn bộ/một phần bài làm của người khác sẽ bị 0 điểm cho toàn bộ phần thực hành

1 Hướng dẫn cài đặt Open NLP

Chi tiết cài đặt và thiết lập: https://www.tutorialspoint.com/opennlp/opennlp_environment.htm

Các mô hình đã được huấn luyện sẵn: http://opennlp.sourceforge.net/models-1.5/

2 Sentence Detector

Công cụ "Sentence detector" giúp chúng ta định ra ranh giới của các câu trong một dòng, một đoạn văn, một bài văn,… Thông thường, các câu thường tách nhau bằng các dấu câu (chấm (.), phẩy (,), chấm cảm (!), chấm hỏi (?), …) hoặc bằng cách xuống dòng. Tuy nhiên, không phải lúc nào các dấu câu này cũng là dấu hiệu phân tách các câu.

Ví dụ: I've been to the city of St. Petersburg before. That place was very beautiful.

Ở đây, ta mong muốn tách được thành 2 câu là "I've been to the city of St. Petersburg before" và "That place was very beautiful". Dấu chấm sau "St" rõ ràng không phải là dấu hiệu phân tách câu.

```
public static void SentenceDetect()throws InvalidFormatException, IOException
{
	String paragraph = "Hi. How are you? This is Mike.";
	// always start with a model, a model is learned from training data
	InputStream is = new FileInputStream("en-sent.bin");
	SentenceModel model = new SentenceModel(is);
	SentenceDetectorME sdetector = new SentenceDetectorME(model);

	String sentences[] = sdetector.sentDetect(paragraph);
	System.out.println(sentences[0]);
	System.out.println(sentences[1]);

	is.close();
}
```

3 Tokenizer

Token là những từ hoặc cụm từ có nghĩa. Thông thường, các từ thường tách nhau bằng khoảng trắng. Tuy nhiên cũng có những trường hợp ngoại lệ khác như từ ghép, từ láy, …

```
public static void Tokenize() throws InvalidFormatException, IOException {
    InputStream is = new FileInputStream("en-token.bin");
    TokenizerModel model = new TokenizerModel(is);
    Tokenizer tokenizer = new TokenizerME(model);
    String tokens[] = tokenizer.tokenize("Hi. How are you? This is Mike.");
    for (String a : tokens)
         System.out.println(a);
    is.close();
}
```

4 Name Finder

Công cụ này sẽ giúp chúng ta tìm tên các thực thể trong văn bản.

```
public static void findName() throws IOException {
        InputStream is = new FileInputStream("en-ner-person.bin");
        TokenNameFinderModel model = new TokenNameFinderModel(is);
        is.close();

        NameFinderME nameFinder = new NameFinderME(model);

        String sentence[] = new String[]{"Mike","Smith","is",
                                         "a","good","person"};

        Span nameSpans[] = nameFinder.find(sentence);

        for(Span s: nameSpans)
               System.out.println(s.toString());
```

5 POS Tagger

Công cụ này giúp chúng ta xác định từ loại của các cụm từ trong văn bản.

Ví dụ: **Hi**._NNP

How_WRB **are**_VBP **you?**_JJ

This_DT **is**_VBZ **Mike**._NNP

```
public static void POSTag() throws IOException {
    POSModel model = new POSModelLoader().load(new File("en-pos-maxent.bin"));
    PerformanceMonitor perfMon = new PerformanceMonitor(System.err, "sent");
    POSTaggerME tagger = new POSTaggerME(model);

    String input = "Hi. How are you? This is Mike.";
    ObjectStream<String> lineStream = new PlainTextByLineStream(new
                                                    StringReader(input));
    perfMon.start();
    String line;

    while ((line = lineStream.read()) != null) {
         String whitespaceTokenizerLine[]
                              = WhitespaceTokenizer.INSTANCE.tokenize(line);
         String[] tags = tagger.tag(whitespaceTokenizerLine);
         POSSample sample = new POSSample(whitespaceTokenizerLine, tags);
         System.out.println(sample.toString());

         perfMon.incrementCounter();
    }
    perfMon.stopAndPrintFinalResult();
}
```

6 Parser

Công cụ này giúp chúng ta xây dựng cây cú pháp từ của một câu.

"Programcreek is a very huge and useful website."

```
(TOP
        (S
                (NP
                        (NN Programcreek)
                )
                (VP
                        (VBZ is)
                        (NP
                                (DT a)
                                (ADJP
                                        (RB very)
                                        (JJ huge)
                                        (CC and)
                                        (JJ useful)
                                )
                        )
                )
                (. website.)
        )
)
```

```
 public static void Parse() throws InvalidFormatException, IOException {

//http://sourceforge.net/apps/mediawiki/opennlp/index.php?title=Parser#Training_Tool
        InputStream is = new FileInputStream("en-parser-chunking.bin");
        ParserModel model = new ParserModel(is);
        Parser parser = ParserFactory.create(model);
        String sentence = "Programcreek is a very huge and useful website.";
        Parse topParses[] = ParserTool.parseLine(sentence, parser, 1);
        for (Parse p : topParses)
                p.show();
        is.close();
}
```

7 Bài tập

Giải nén data.zip để được thư mục data có cấu trúc như sau:

data

in

Child

data3.txt

data1.txt

data2.txt

out

pos_tagger

parse

1. **Duyệt từng file trong data/in (kế cả thư mục con), với mỗi file tách thành từng câu. Gán nhãn từ loại các từ trong câu và lưu lại vào file với cùng tên file input nhưng vào thư mục out/pos_tagger. Vẽ cây cú pháp cho từng câu và lưu lại vào file cùng tên với file input nhưng vào thư mục out/parse.**
2. **Vẽ cây thư mục data (sau khi thực hiện xong câu 1) ra màn hình Console.**

Tài liệu tham khảo:

[1] https://www.tutorialspoint.com/opennlp/index.htm

[2] http://www.programcreek.com/2012/05/opennlp-tutorial/