Natural Language Processing (NLP)

Course:
INFO-6145 Data Science and Machine Learning

FANSHAWE

Revised by:
Mohammad Noorchenarboo

October 22, 2024

Contents

Current Section

How Natural Language Processing Works

Natural language processing (NLP) is a subfield of artificial intelligence that tries to process and analyze natural language data. It includes teaching machines to interact with humans in a natural language (a language that developed naturally through use).

Importance of NLP

NLP allows machines to understand and interact with human language, facilitating tasks such as translation, chatbots, and more.

Natural Language Understanding (NLU)

Natural Language Understanding (NLU) is a subtopic of natural language processing that deals with machine reading comprehension.

Example of NLU

When a virtual assistant like Alexa answers a question, it uses NLU to interpret the user's query.

Natural Language Generation (NLG)

Natural Language Generation (NLG) is what happens when computers write language. NLG processes turn structured data into text.

Importance of NLG

NLG helps transform complex data into human-readable language, such as generating reports, descriptions, or summaries from datasets.

Example of NLG

A weather app that provides text-based weather forecasts based on numerical weather data is using NLG.

Speech Recognition (SR)

Speech Recognition (SR) is what happens when a computer or machine is able to successfully identify spoken words with a high level of accuracy. SR inputs audio data and outputs textual data.

Importance of SR

SR enables hands-free interaction with devices, which is essential for accessibility, virtual assistants, and transcription services.

Example of SR

Using voice commands like "Hey Google, set a reminder" on your smartphone involves speech recognition.

Speech Synthesis (SS)

Speech Synthesis (SS) is the process of generating an artificial audio voice from natural language text.

Challenges in SS

Speech synthesis may sound unnatural or robotic if the technology isn't advanced, leading to user frustration.

Importance of SS

SS is crucial for applications like screen readers, where the text must be converted to speech for visually impaired users.

Example of SS

Screen readers used by visually impaired individuals rely on speech synthesis to read text aloud.

Current Section

Introduction to Word Embedding

Word embedding is a method used in natural language processing (NLP) to represent words as vectors of real numbers. This transformation allows us to apply mathematical operations on words, making it easier for machines to process and analyze language.

Why Word Embedding is Required

- Computers cannot understand words directly; they need numerical representations to perform calculations.
- Word embedding captures the semantic meaning of words in a way that reflects their relationships in a vector space.

How Word Embedding Works

Word embedding maps each word to a multi-dimensional vector. Words with similar meanings tend to have vectors that are close together in this vector space.

Key Concept

Distributional Hypothesis: Words that appear in similar contexts tend to have similar meanings. Word embeddings rely on this idea to place similar words near each other in the vector space.

How Word Embedding Works

Numerical Example of Word Embedding

- **Word:** "king" **Vector:** [0.23, 0.50, -0.12, 0.44, 0.11]
- **Word:** "queen" **Vector:** [0.21, 0.49, -0.11, 0.45, 0.10]
- **Word:** "apple" **Vector:** [-0.15, 0.12, 0.78, -0.43, 0.02]

The vectors for "king" and "queen" are similar because they are semantically related, whereas "apple" has a very different vector because its meaning is unrelated.

Benefits of Word Embedding

Capturing Semantic Relationships

Word embedding allows us to capture relationships between words beyond just direct synonyms. It can identify patterns such as:

- **Analogy:** "king" is to "queen" as "man" is to "woman." This is reflected in their vector representations, where the difference between "king" and "queen" is similar to the difference between "man" and "woman."

Numerical Example of Analogy

king - man + woman = queen

Vector Calculation:

, 0.50, -0.12, 0.44, 0.11

- [0.17, 0.40, -0.09, 0.38, 0.09] + [0.18, 0.42, -0.10, 0.40, 0.08]

Results in the vector for "queen": [0.21, 0.49, -0.11, 0.45, 0.10]

Common Word Embedding Techniques

1. Word2Vec

This method predicts words based on their context (the words around them). It uses a neural network to learn word vectors that capture the relationships between words.

2. GloVe (Global Vectors for Word Representation)

This method uses word co-occurrence statistics from a large text corpus. It creates vectors where the relationships between word pairs reflect their co-occurrences.

3. FastText

FastText represents words as subword units, making it capable of handling rare or out-of-vocabulary words by analyzing parts of the word.

Why Word Embedding is Crucial for NLP

Efficient Computation

Word embeddings reduce the complexity of text data, allowing computers to efficiently process language for various NLP tasks.

Improved Performance

Models trained on word embeddings perform better in tasks like translation, text classification, and named entity recognition because they can understand the semantic relationships between words.

Real-World Application

Virtual assistants, like Siri or Alexa, rely on word embeddings to understand the context and meaning of spoken language, allowing them to respond appropriately to user queries.

Current Section

Machine Learning for NLP

Machine learning plays a crucial role in Natural Language Processing (NLP), enabling computers to understand and generate human language. Several key tasks allow machines to perform meaningful language analysis.

Key NLP Tasks

- **Syntactic Dependency Parsing:** This task determines the grammatical relationships between words in a sentence, such as which word is the subject or object.
- **Part-of-Speech Tagging:** Involves labeling each word with its appropriate part of speech, like noun, verb, or adjective.
- **Named Entity Recognition (NER):** Identifies and categorizes proper nouns into predefined categories, such as people, organizations, or locations.

Example of Dependency Parsing

In the sentence "The cat sat on the mat," dependency parsing identifies that:

- **Subject:** "The cat" is the subject of the sentence.
- **Verb:** "Sat" is the action performed.
- **Prepositional phrase:** "on the mat" tells us where the cat sat.

Numerical Example of Dependency Tree

Each word is connected by arrows to its related word:

```
   sat
   / \
the cat   on
         / \
      the mat
```

Life Cycle of a Machine Learning System

The machine learning life cycle for NLP systems follows a structured process, typically involving three key steps:

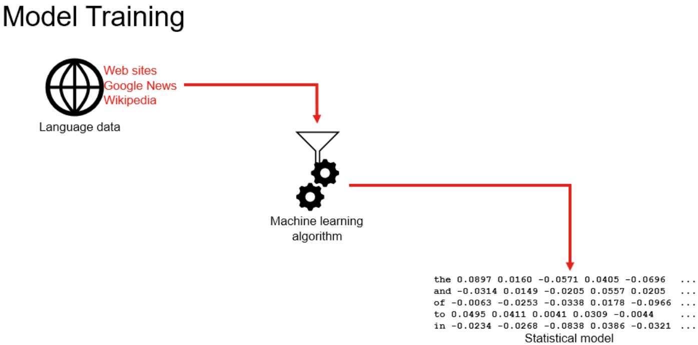
1. Model Training

- During model training, a machine learning algorithm is trained on a dataset to recognize patterns in language.
- Training data can be generic, like data from Wikipedia or Google News, or specific, like customer interactions for a particular business application.

Life Cycle of a Machine Learning System

Example of Training Data

Training a model on a large corpus like Wikipedia enables it to understand general language usage, whereas training it on customer service logs helps the model learn specific patterns relevant to customer inquiries.



2. Model Testing

Once the model is trained, it is tested to evaluate its performance. This process ensures that the model can generalize its knowledge to unseen text and identify semantic similarities, syntactic patterns, and named entities.

How Testing Works

During testing, the model is fed new data (text it hasn't seen before) to determine how well it can apply the knowledge gained during training.

2. Model Testing

Testing Example

A trained model is given the sentence: "Apple Inc. is based in Cupertino." The model should be able to:

- Recognize that "Apple Inc." is an organization (Named Entity Recognition).
- Identify that "is based" indicates a relationship between the company and a location.
- Parse "Cupertino" as the location.

3. Making Predictions

After testing, the model can be used to make predictions. In an NLP application, a common task is predicting the dependency tree structure over a sentence, which illustrates the relationships between words.

Example of Making Predictions

A dependency tree generated by the model can show how words are related. For instance, the model might predict that:

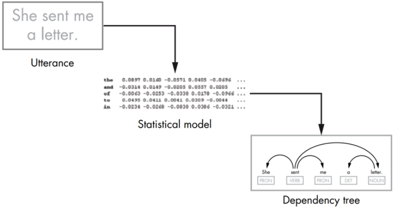
- "John" is the subject of the sentence.
- "bought" is the verb or action.
- "a car" is the object, detailing what was bought.

Numerical Example

Numerical Example of Prediction

In the sentence "John bought a car," the model predicts a dependency tree:

```
  bought
  /   \
John  car
       |
       a
```



Why Model Training, Testing, and Prediction Matter

Key Takeaway

The training, testing, and prediction steps are fundamental to building machine learning systems that can understand and interpret human language accurately.

Real-World Application

In chatbots, models trained on conversation data allow the system to predict and generate appropriate responses, providing smooth human-machine interaction.

Current Section

Why Use Machine Learning for NLP?

Machine learning models aren't perfect, but they offer a powerful approach to building natural language processing (NLP) systems. Why do we use machine learning for NLP instead of a rule-based approach? Let's compare programming languages (e.g., Java, C++) to natural languages (e.g., English, French).

Programming Languages vs. Natural Languages

- **Programming Languages:** Based on "context-free" grammars, with a small set of well-defined reserved words. For example, Java has only 61 reserved words.
- **Natural Languages:** Based on "context-dependent" grammars and have a much larger vocabulary. English has 171,476 words (Oxford English Dictionary, 1989).

Challenges of Natural Language Processing

In natural languages, words often have multiple meanings. For example, the word "count" can be a verb or a noun, and it has different meanings depending on the context:

- "To have value or importance."
- "To say numbers one after another."
- "Count" as a noun, meaning "a title of nobility."

Combined Words: N-Grams

In addition to individual words, natural languages also have combined word phrases, known as N-grams, such as "ice cream" or "power supply."

Loosely Defined Rules in Natural Languages

Many rules in natural languages are defined loosely. Consider the example of a split infinitive:

- **Example 1:** NLP apps allow you to **programmatically extract** the meaning of an utterance.
- **Example 2:** NLP apps allow you to **extract the meaning programmatically**.

These variations show the flexibility of natural language rules, which makes a rule-based approach inadequate for handling all possibilities in real-world language use.

Machine Learning for NLP

Instead of encoding a language using predefined rules, machine learning-based NLP systems use a data-driven approach. Machine learning algorithms detect patterns in large amounts of language data and build statistical models that make predictions about unseen text.

Statistical Model Approach

Rather than assigning each word to a fixed category or rule, machine learning generates a statistical model. This model can predict the syntactic structure, meaning, and relationships between words in new, previously unseen data.

NLP Systems vs. Compilers

NLP System (e.g., ChatGPT)

A natural language processing system generates responses based on underlying statistical models. It predicts the meaning of the input text and formulates a response based on patterns observed in language data.

Compiler for Programming Languages

In contrast, a compiler for programming code applies a strictly defined set of rules to process the input. The grammar of programming languages is rigid and well-defined, making rule-based processing feasible.

What is a Statistical Model in NLP?

A statistical model in NLP contains estimates for the probability distribution of linguistic units, such as words or phrases. This allows you to assign features to words based on the likelihood of their occurrence in a particular context.

Probability Distribution in NLP

In probability theory, a probability distribution maps all possible outcomes of a variable to their probabilities. In NLP, this concept helps determine the likelihood of different parts of speech or word meanings in a given sentence.

Example of Probability Distribution

Consider the word "count" in the sentence "Can we count on them?" The statistical model assigns probabilities to each part-of-speech tag based on context:

- Verb (90%): "Count" is most likely used as a verb here.
- Noun (10%): There's a small chance "count" could be used as a noun, depending on the context.

Numerical Example of Probability Distribution

- **Verb:** 0.90 (90%)
- **Noun:** 0.10 (10%)

Current Section

Recognizing Intent

Suppose you sell clothes, and your online application receives the request: "I want to order a pair of jeans." The goal of the NLP system is to recognize that the user's intent is to place an order for jeans.

Intent Recognition in NLP

NLP systems aim to identify what a user wants to accomplish, which, in this case, is placing an order. This is called intent recognition.

Dependency Parsing for Intent Recognition

Modern NLP systems can perform syntactic dependency parsing, which analyzes the grammatical structure of a sentence. Here's how dependency parsing might work for the sentence "I want to order a pair of jeans":

```
    order
      |
    want
   / |  \
I    to   jeans
        / \
       a  pair
```

Syntactic Structure

The verb "order" is linked to the noun "jeans," while the verb "want" shows the user's desire.

Challenge in Recognizing Intent

Notice that the generated syntactic tree doesn't directly mark the user's intent. That's because generic NLP systems don't know your application's logic or the types of intent you're looking for.

Custom Intent Recognition

To effectively extract intent, you need to tailor the NLP system to your application's logic, understanding the expected keywords and context.

Understanding the Aspects of Intent Recognition

To successfully extract the meaning or intent from an utterance, you need to analyze three key aspects:

- **Keywords:** Identify key words that signal the intent.
- **Context:** Consider the surrounding context to interpret the meaning correctly.
- **Meaning Transition:** Analyze meaning transitions between sentences or phrases.

Keywords

Using syntactic dependency parsing, you can select the most important words for meaning recognition. For example, in the sentence "I want to order a pair of jeans," the key words are likely "order" and "jeans."

Keyword Example

In this case:

- **Verb:** "Order" indicates the action.
- **Noun:** "Jeans" specifies the object.

Context

Context is essential when selecting keywords, as the same phrase might have different meanings depending on the situation.

Context Example

Consider the phrase: "I want the newspaper delivered to my door."

- In one context, the intent could be a request for a newspaper subscription, with "want" and "newspaper" as key words.
- In another context, the intent could be a delivery request, with "delivered" and "door" as the key words.

Meaning Transition

Often, intent is expressed across multiple sentences. For example:

- "I already have a relaxed pair of jeans. Now I want a skinny pair."

The user's intent is spread across both sentences: the first describes the current state, while the second expresses a new desire.

Steps for Finding Intent

To find the keywords that describe the user's intent:

- Step 1: Find a transitive verb in the present tense (e.g., "want").
- Step 2: Identify the direct object of that verb (e.g., "skinny pair").
- Step 3: If the object is a pro-form (e.g., "it"), find its antecedent in the previous sentence.

Intent Recognition Example

Consider the following discourse:

- "I already have a relaxed pair of jeans. Now I want a skinny pair."

Step-by-Step Process

- **Verb:** "Want" expresses the user's desire.
- **Object:** "Skinny pair" is what the user wants to order.
- **Previous Context:** "Relaxed pair" informs us about the existing product, but is not part of the new intent.