README for Task 2

Overview

This Python script is designed to tokenize, tag, and parse sentences based on parts of speech (POS). It employs libraries such as `ply` for lexing and parsing, `nltk` for natural language processing, and regular expressions for token matching. The script focuses on identifying articles, nouns, verbs, and a specific category of verbs (verbex).

Dependencies

- `ply`: Used for implementing lex and yacc functionalities.
- `nltk`: Utilized for processing natural language data.
- `re`: Regular expressions for pattern matching in tokens.

Implementation Details

Token Definitions

- **ARTICLE**: Detected based on a predefined set of articles extracted from the 'brown' corpus in `nltk`.
- **NOUN**: Identified using a list of nouns extracted from the 'brown' corpus.
- **VERB**: Detected by a list of verbs, also extracted from the 'brown' corpus.
- **VERBEX**: A custom category representing auxiliary verbs like 'is', 'am', 'was', 'were', 'are'.

Regular Expressions (Regex)

- `t_NOUN`, `t_ARTICLE`, `t_VERBEX`, and `t_VERB` are regex patterns created by joining words in their respective categories with the pipe `|` symbol, which denotes an 'OR' condition in regex.

Each word is escaped using `re.escape` to ensure special characters are treated as literals.

Lexing

- The lexer, created using `ply.lex`, tokenizes the input based on these regex patterns.
- `t_ignore` is set to ignore spaces and tabs.
- `t_error` is a function to handle lexing errors, skipping over problematic characters.

Parsing

- The parser, implemented with `ply.yacc`, defines grammar rules for constructing valid sentences.
- The grammar rules define a sentence structure, recognizing subjects followed by verb phrases.
- Subjects can be an article followed by a noun or just a noun.
- Verb phrases can be a combination of `VERBEX` and `VERB` or a standalone `VERB`.

Sentence Validation

- `isvalidsentence` function takes a string, tokenizes it using the lexer, and then parses it to check if it forms a valid sentence based on the defined grammar.
- It prints whether the sentence is valid and its structure if it is valid.

Usage

The script prompts the user to enter a sentence. It then processes this input to determine if it's a grammatically valid sentence according to the defined rules and outputs the result.

Notes

- The `nltk.download('brown')` line ensures that the 'brown' corpus is downloaded, which is necessary for extracting words for the ARTICLE, NOUN, and VERB categories.
- The script is limited to the words and structures defined in its grammar and may not cover all

complexities of the English language.