1-Removes punctuation symbols :

This function likely removes punctuation symbols from a given text string. Punctuation symbols are characters like periods, commas, exclamation marks, question marks, and so on. Removing them can be useful for text processing tasks like sentiment analysis, text classification, or language modeling, where punctuation might not be relevant to the analysis being performed.

من المحتمل أن تقوم هذه الوظيفة بإزالة رموز علامات الترقيم من سلسلة نصية معينة .رموز علامات الترقيم هي أحرف مثل النقاط والفواصل وعلامات التعجب وعلامات الاستفهام وما إلى ذلك .يمكن أن تكون إزالتها مفيدة لمهام معالجة النص مثل تحليل المشاعر، أو تصنيف النص، أو نمذجة اللغة، حيث قد لا تكون علامات الترقيم ذات صلة بالتحليل الذي يتم إجراؤه.

2-Filtering Stop Words

This function probably filters out stop words from a given text. Stop words are common words that are often filtered out during natural language processing tasks because they typically do not carry significant meaning or information about the content of the text. Examples of stop words include "the", "is", "and", "in", etc. Removing them can help focus on the more important words in the text for tasks like sentiment analysis, text summarization, or topic modeling.

من المحتمل أن تقوم هذه الوظيفة بتصفية كلمات التوقف من نص معين .الكلمات المتوقفة هي كلمات شائعة غالبًا ما يتم تصفيتها أثناء مهام معالجة اللغة الطبيعية لأنها لا تحمل عادةً معنى أو معلومات مهمة حول محتوى النص .تتضمن أمثلة كلمات التوقف "the" و"is" و"and" و"in" وما إلى ذلك .ويمكن أن تساعد إزالتها في التركيز على الكلمات الأكثر أهمية في النص لمهام مثل تحليل المشاعر أو تلخيص النص أو نمذجة الموضوع.

3-Stemming

This function likely performs stemming on words in a given text. Stemming is a process in natural language processing where words are reduced to their root or base form, by removing suffixes. For example, the words "running", "ran", and "runner" would all be stemmed to "run". Stemming helps to normalize words so that variations of the same word are treated as the same, which can improve tasks like information retrieval, search, and text analysis.

من المحتمل أن تؤدي هذه الوظيفة إلى الكلمات الموجودة في نص معين .الاشتقاق هو عملية في معالجة اللغة الطبيعية حيث يتم اختزال الكلمات إلى جذرها أو شكلها الأساسي، عن طريق إزالة اللواحق .على سبيل المثال، الكلمات "يجري " و"ران "و"عداء "كلها مشتقة من كلمة "يركض ."يساعد التجذير على تطبيع الكلمات بحيث يتم التعامل مع الاختلافات في نفس الكلمة على أنها نفسها، مما قد يؤدي إلى تحسين المهام مثل استرجاع المعلومات والبحث وتحليل النص.

4-Tagging Parts of Speech

This function likely assigns parts of speech tags to words in a given text. Parts of speech tagging is a process in natural language processing where each word in a sentence is assigned a grammatical category, such as noun, verb, adjective, etc. This helps in understanding the structure and meaning of a sentence, which is useful for various tasks like syntactic analysis, information extraction, and text understanding.

من المحتمل أن تقوم هذه الوظيفة بتعيين أجزاء من علامات الكلام للكلمات الموجودة في نص معين .أجزاء من تمييز الكلام هي عملية في معالجة اللغة الطبيعية حيث يتم تعيين فئة نحوية لكل كلمة في الجملة، مثل الاسم والفعل والصفة وما إلى ذلك .وهذا يساعد في فهم بنية الجملة ومعناها، وهو أمر مفيد لمختلف اللغات .مهام مثل التحليل النحوي واستخراج المعلومات وفهم النص.

5-Lemmatizing

This function performs lemmatization on words in a given text. Lemmatization is a process in natural language processing where words are reduced to their base or dictionary form, known as the lemma. Unlike stemming, which simply chops off affixes to produce a word stem, lemmatization considers the meaning of the word and applies morphological analysis to accurately return the lemma. For example, the word "running" would be lemmatized to "run", "better" would be lemmatized to "good", and so on. Lemmatization is useful for tasks like text normalization, where you want to reduce inflected words to a common base form for analysis.

تقوم هذه الوظيفة بإجراء عملية lemmatization على الكلمات الموجودة في نص معين. Lemmatization هي عملية في معالجة اللغة الطبيعية حيث يتم اختزال الكلمات إلى شكلها الأساسي أو المعجمي، المعروف باسم lemma. على عكس الاشتقاق، الذي يقوم ببساطة بتقطيع اللواحق لإنتاج أصل الكلمة، فإن التجسيد يأخذ في الاعتبار معنى الكلمة ويطبق التحليل المورفولوجي لإرجاع ليما بدقة .على سبيل المثال، سيتم تحويل كلمة "يجري "إلى "تشغيل"، وسيتم تحويل كلمة "يعد" أفضل "إلى "جيد"، وهكذا Lemmatization مفيدًا لمهام مثل تسوية النص، حيث تريد تقليل الكلمات المصروفة إلى نموذج أساسي مشترك للتحليل.

6-Chunking

This function likely performs chunking or shallow parsing on a given text. Chunking is a process in natural language processing where words or tokens in a sentence are grouped together based on their syntactic structure. These groups are called chunks, and they often consist of phrases like noun phrases (NP), verb phrases (VP), etc. Chunking helps in identifying meaningful units in the text beyond individual words, which is useful for tasks like information extraction, named entity recognition, and parsing.

من المحتمل أن تقوم هذه الوظيفة بإجراء تحليل جزئي أو سطحي لنص معين .التقطيع هو عملية في معالجة اللغة الطبيعية حيث يتم تجميع الكلمات أو الرموز المميزة في الجملة معًا بناءً على بنيتها النحوية .تسمى هذه المجموعات قطعًا، وما إلى ذلك .يساعد القطع في تحديد ،(VP) وعبارات الفعل ،(NP) وغالبًا ما تتكون من عبارات مثل عبارات الاسم الوحدات ذات المعنى في النص بما يتجاوز الكلمات الفردية، وهو أمر مفيد لمهام مثل استخراج المعلومات، والمسمى التعرف على الكيانات والتحليل.

7-Chinking

The term "chinking" in natural language processing refers to the process of removing certain parts from a chunk, usually to define what the chunk should not contain. It's essentially the opposite of chunking, where instead of specifying what should be included in a chunk, you specify what should be excluded.

For example, if you have a chunk that represents a noun phrase, you might want to remove certain types of words, like determiners or adjectives, from that chunk. Chinking allows you to define patterns to remove such words from the chunk while still maintaining the overall structure.

Chinking can be useful in various text processing tasks, particularly when you want to customize what information is retained or discarded during chunking.

يشير مصطلح "التقطيع "في معالجة اللغة الطبيعية إلى عملية إزالة أجزاء معينة من قطعة ما، عادةً لتحديد ما لا ينبغي أن تحتويه القطعة .إنه في الأساس عكس التجزئة، حيث بدلاً من تحديد ما يجب تضمينه في القطعة، يمكنك تحديد ما يجب استبعاده.

على سبيل المثال، إذا كان لديك مقطع يمثل عبارة اسمية، فقد ترغب في إزالة أنواع معينة من الكلمات، مثل المحددات أو تحديد الأنماط لإزالة مثل هذه الكلمات من القطعة مع الحفاظ على البنية Chinking الصفات، من هذا المقطع .يتيح لك العامة.

يمكن أن يكون التقطيع مفيدًا في العديد من مهام معالجة النص، خاصة عندما تريد تخصيص المعلومات التي يتم الاحتفاظ بها أو التخلص منها أثناء التقطيع.

8-Named-Entity Recognition

This function performs Named-Entity Recognition (NER) on a given text. Named-Entity Recognition is a natural language processing task where the goal is to identify and classify named entities mentioned in the text into pre-defined categories such as person names, organization names, locations, dates, monetary values, etc.

For example, in the sentence "Apple is headquartered in Cupertino, California", NER would identify "Apple" as an organization name and "Cupertino, California" as a location.

NER is essential for many NLP applications, such as information retrieval, question answering, text summarization, and more, as it helps in extracting and understanding specific entities mentioned in the text.

تقوم هذه الوظيفة بالتعرف على الكيانات المسماة (NER) على نص محدد. التعرف على الكيانات المسماة هي مهمة معالجة لغة طبيعية حيث يكون الهدف هو تحديد وتصنيف الكيانات المسماة المذكورة في النص إلى فئات محددة مسبقًا مثل أسماء الأشخاص وأسماء المؤسسات والمواقع والتواريخ والقيم النقدية وما إلى ذلك.

على سبيل المثال، في الجملة "يقع المقر الرئيسي لشركة Apple في كوبرتينو، كاليفورنيا"، ستحدد NER "Apple" كاسم مؤسسة و"Cupertino، كموقع "كاليفورنيا.

NER يعد ضروريًا للعديد من تطبيقات البرمجة اللغوية العصبية، مثل استرجاع المعلومات، والإجابة على الأسئلة، وتلخيص النص، والمزيد، لأنه يساعد في استخلاص وفهم كيانات محددة مذكورة في النص.

9-Dependency Parsing

This function performs dependency parsing on a given text. Dependency parsing is a natural language processing technique that analyzes the grammatical structure of a sentence to establish relationships between words. It represents these relationships as directed links between words, where each link indicates a syntactic dependency between a head word and its dependent(s).

The head word is typically the main word in a phrase, and the dependents are words that modify or are governed by the head word. These dependencies form a tree-like structure called a dependency tree, which represents the syntactic structure of the sentence.

Dependency parsing is valuable for tasks such as information extraction, text understanding, and question answering, as it provides insights into the relationships between words in a sentence, helping in understanding the meaning and structure of the text.

تقوم هذه الوظيفة بتحليل التبعية على نص معين .تحليل التبعية هو أسلوب معالجة لغة طبيعية يقوم بتحليل البنية النحوية للجملة لإقامة علاقات بين الكلمات .وهو يمثل هذه العلاقات كروابط موجهة بين الكلمات، حيث يشير كل رابط إلى التبعية النحوية بين الكلمة الرئيسية والتابعين لها.

الكلمة الرئيسية هي عادة الكلمة الرئيسية في العبارة، والكلمات التابعة هي الكلمات التي تعدل أو تحكمها الكلمة الرئيسية .
تشكل هذه التبعيات بنية شبيهة بالشجرة تسمى شجرة التبعية، والتي تمثل البنية النحوية للجملة.

يعد تحليل التبعية ذا قيمة لمهام مثل استخراج المعلومات، وفهم النص، والإجابة على الأسئلة، لأنه يوفر نظرة ثاقبة للعلاقات بين الكلمات في الجملة، مما يساعد في فهم معنى النص وبنيته.

10-Rule-Based Matching

This function likely performs rule-based matching on a given text. Rule-based matching is a technique in natural language processing used to identify sequences of tokens in text that match a set of predefined patterns or rules.

These rules can be simple or complex and can involve regular expressions, token attributes, or linguistic features. Rule-based matching is often used for tasks such as named entity recognition, information extraction, and text pattern matching.

For example, if you want to find all instances of dates in a text, you could define a rule that matches sequences of tokens that resemble dates based on their format (e.g., "January 1, 2024" or "12/31/23").

Rule-based matching provides flexibility and control over the patterns you want to extract from the text, making it a powerful tool in natural language processing pipelines.

من المحتمل أن تقوم هذه الوظيفة بالمطابقة المستندة إلى القواعد على نص معين .المطابقة المستندة إلى القواعد هي تقنية في معالجة اللغة الطبيعية تُستخدم لتحديد تسلسلات الرموز المميزة في النص التي تتطابق مع مجموعة من الأنماط أو القواعد المحددة مسبقًا.

يمكن أن تكون هذه القواعد بسيطة أو معقدة ويمكن أن تتضمن تعبيرات عادية أو سمات رمزية أو ميزات لغوية .غالبًا ما يتم استخدام المطابقة المستندة إلى القواعد لمهام مثل التعرف على الكيانات المسماة واستخراج المعلومات ومطابقة أنماط النص.

على سبيل المثال، إذا كنت تريد البحث عن كافة مثيلات التواريخ في نص ما، فيمكنك تحديد قاعدة تطابق تسلسلات الرموز المميزة التي تشبه التواريخ بناءً على تنسيقها )على سبيل المثال، "1 يناير 2024 "أو "31/12/23( " .

توفر المطابقة المستندة إلى القواعد المرونة والتحكم في الأنماط التي تريد استخراجها من النص، مما يجعلها أداة قوية في مسارات معالجة اللغة الطبيعية.

11-Vectorization types

"Vectorization types" likely refers to the different methods or approaches used to convert textual data into numerical vectors that can be processed by machine learning algorithms. Here are some common vectorization types in natural language processing (NLP):

Bag-of-Words (BoW): Represents text data as a sparse matrix where each row corresponds to a document and each column corresponds to a unique word in the entire corpus. The values in the matrix represent word frequencies or presence/absence indicators.

Term Frequency-Inverse Document Frequency (TF-IDF): Similar to BoW, but instead of raw word counts, it computes a weight for each word in a document that depends on its frequency in the document and the inverse frequency across all documents in the corpus.

Word Embeddings: Represent words as dense, low-dimensional vectors in a continuous vector space. Word embeddings capture semantic relationships between words based on their

contextual usage and are typically learned from large text corpora using techniques like Word2Vec, GloVe, or FastText.

Doc2Vec: An extension of Word2Vec that learns fixed-length vector representations for entire documents. Doc2Vec enables comparisons and analysis of document similarity by learning document embeddings.

Character-level Embeddings: Represent text data at the character level rather than the word level. Each character is mapped to a vector, and words are represented as sequences of character vectors. Character-level embeddings can handle out-of-vocabulary words and capture morphological information.

Byte Pair Encoding (BPE): A subword tokenization technique that segments words into smaller subword units based on their frequency in the corpus. BPE iteratively merges the most frequent pairs of characters or character sequences to build a vocabulary of subword units.

These vectorization techniques enable the numerical representation of text data, allowing machine learning algorithms to process and analyze textual information effectively. The choice of vectorization type depends on the specific characteristics of the data and the requirements of the NLP task at hand.

من المحتمل أن تشير "أنواع المتجهات "إلى الأساليب أو الأساليب المختلفة المستخدمة لتحويل البيانات النصية إلى متجهات رقمية يمكن معالجتها بواسطة خوارزميات التعلم الآلي .فيما يلي بعض أنواع التوجيه الشائعة في معالجة اللغة الطبيعية (NLP):

Bag-of-Words (BoW): يمثل البيانات النصية كمصفوفة متفرقة حيث يتوافق كل صف مع مستند وكل عمود يتوافق مع كلمة فريدة في المجموعة بأكملها .تمثل القيم الموجودة في المصفوفة ترددات الكلمات أو مؤشرات الحضور/الغياب.

تكرار المصطلح - تردد المستند العكسي (TF-IDF): يشبه BoW، ولكن بدلاً من عدد الكلمات الأولية، فإنه يحسب وزنًا لكل كلمة في مستند يعتمد على تكرارها في المستند والتكرار العكسي عبر جميع المستندات في المستند .جسم.

تضمين الكلمات :تمثيل الكلمات كمتجهات كثيفة ومنخفضة الأبعاد في مساحة متجهة مستمرة .تلتقط عمليات تضمين الكلمات العلاقات الدلالية بين الكلمات بناءً على استخدامها السياقي ويتم تعلمها عادةً من مجموعات نصية كبيرة باستخدام تقنيات مثل Word2Vec أو GloVe أو FastText.

Doc2Vec: امتداد لبرنامج Word2Vec .يتيح يتعلم تمثيلات المتجهات ذات الطول الثابت للمستندات بأكملها إجراء مقارنات وتحليل تشابه المستندات من خلال تعلم تضمين المستندات Doc2Vec.

التضمين على مستوى الحرف :يمثل البيانات النصية على مستوى الحرف بدلاً من مستوى الكلمة .يتم تعيين كل حرف إلى متجه، ويتم تمثيل الكلمات كتسلسلات من ناقلات الأحرف .يمكن للتضمينات على مستوى الأحرف التعامل مع الكلمات خارج المفردات والتقاط المعلومات المورفولوجية.

تقنية ترميز الكلمات الفرعية التي تقوم بتقسيم الكلمات إلى وحدات كلمات فرعية أصغر بناءً :(BPE) تشفير زوج البايت بشكل متكرر بدمج أزواج الأحرف أو تسلسلات الأحرف الأكثر شيوعًا لبناء مفردات BPE على تكرارها في المجموعة .يقوم وحدات الكلمات الفرعية.

تتيح تقنيات التوجيه هذه التمثيل العددي للبيانات النصية، مما يسمح لخوارزميات التعلم الآلي بمعالجة المعلومات النصية وتحليلها بشكل فعال .يعتمد اختيار نوع التوجيه على الخصائص المحددة للبيانات ومتطلبات مهمة البرمجة اللغوية العصبية (NLP) المطروحة.