US 20090089058A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2009/0089058 A1

(43) Pub. Date: Apr. 2, 2009

Bellegarda

(54) PART-OF-SPEECH TAGGING USING LATENT ANALOGY

(76) Inventor: **Jerome Bellegarda**, Los Gatos, CA (US)

Correspondence Address:
James C. Scheller
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
1279 Oakmead Parkway
Sunnyvale, CA 94085-4040 (US)

(21) Appl. No.: **11/906,592**

(22) Filed: **Oct. 2, 2007**

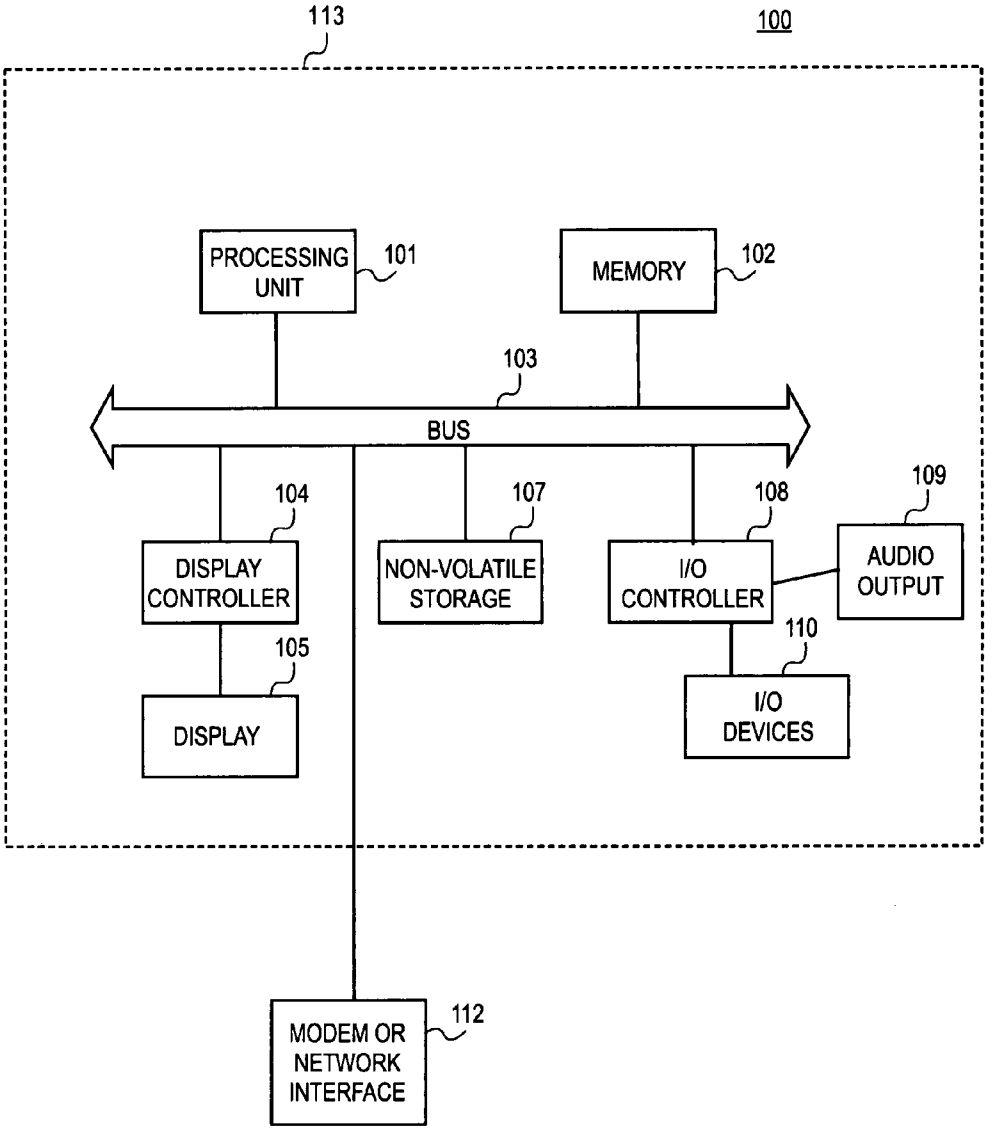
Publication Classification

(51) **Int. Cl.**
G10L 15/00 (2006.01)

(52) **U.S. Cl.** **704/251**

(57) **ABSTRACT**

Methods and apparatuses to assign part-of-speech tags to words are described. An input sequence of words is received. A global fabric of a corpus having training sequences of words may be analyzed in a vector space. A global semantic information associated with the input sequence of words may be extracted based on the analyzing. A part-of-speech tag may be assigned to a word of the input sequence based on POS tags from pertinent words in relevant training sequences identified using the global semantic information. The input sequence may be mapped into a vector space. A neighborhood associated with the input sequence may be formed in the vector space wherein the neighborhood represents one or more training sequences that are globally relevant to the input sequence.



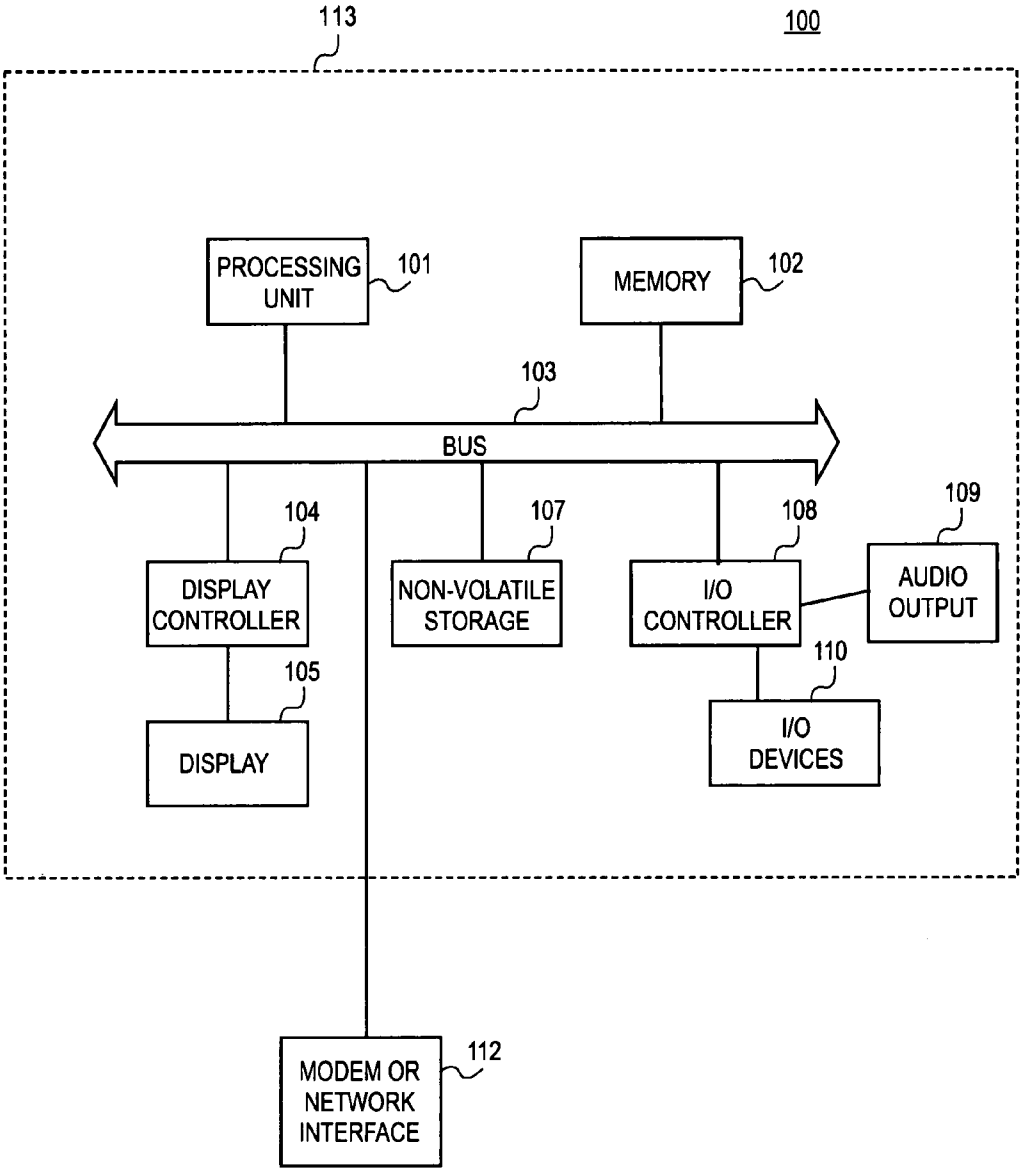
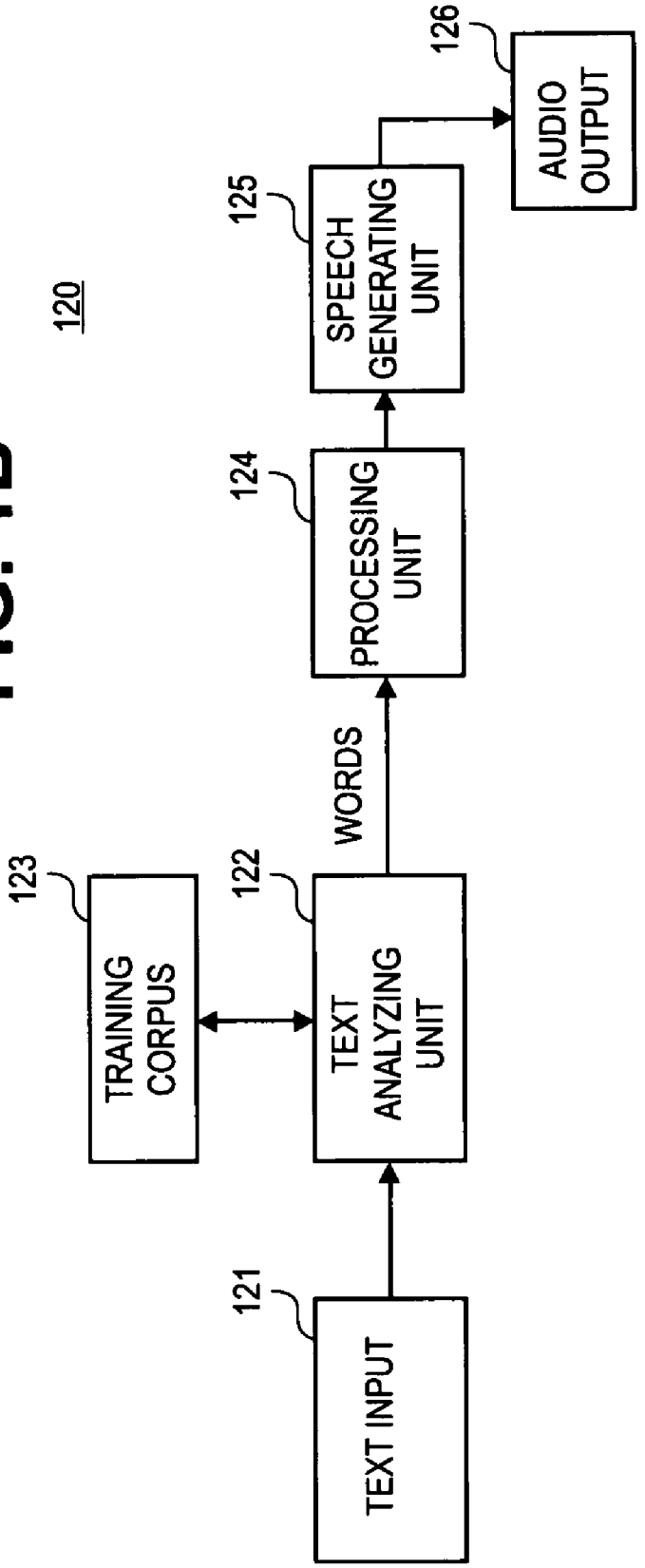

113
100
PROCESSING UNIT
101
MEMORY
102
103
BUS
104
107
108
109
DISPLAY CONTROLLER
NON-VOLATILE STORAGE
I/O CONTROLLER
AUDIO OUTPUT
105
110
DISPLAY
I/O DEVICES
MODEM OR NETWORK INTERFACE
112

FIG. 1A

FIG. 1B


120
121
TEXT INPUT
122
TEXT ANALYZING UNIT
123
TRAINING CORPUS
WORDS
124
PROCESSING UNIT
125
SPEECH GENERATING UNIT
126
AUDIO OUTPUT

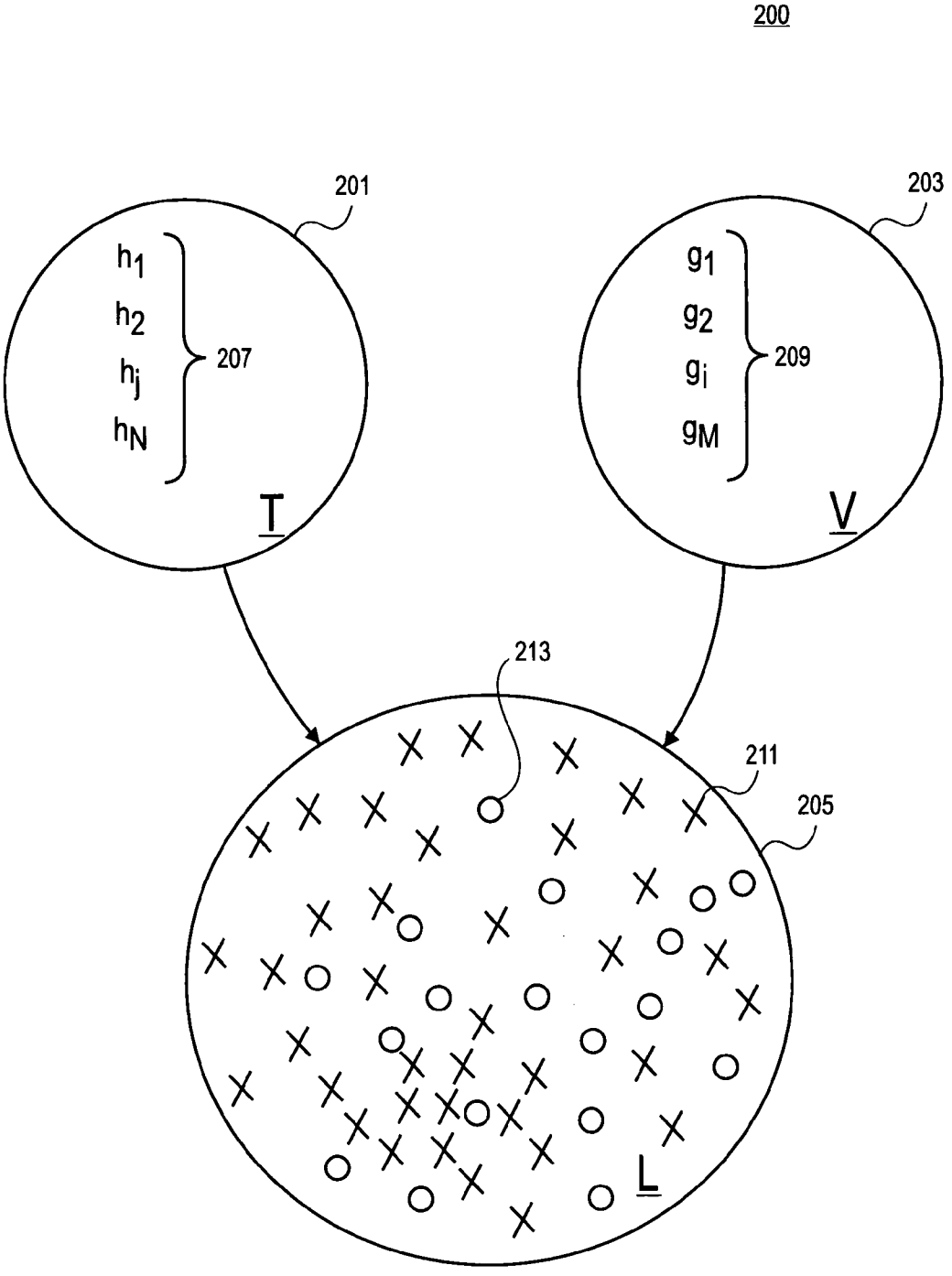


FIG. 2

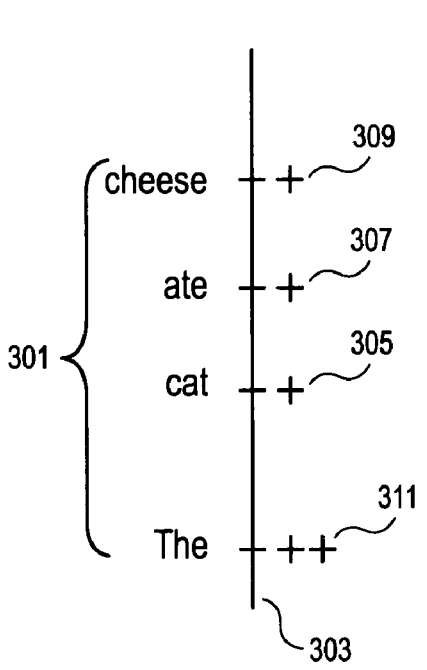

300
309
cheese
307
ate
305
301
cat
311
The
303

FIG. 3

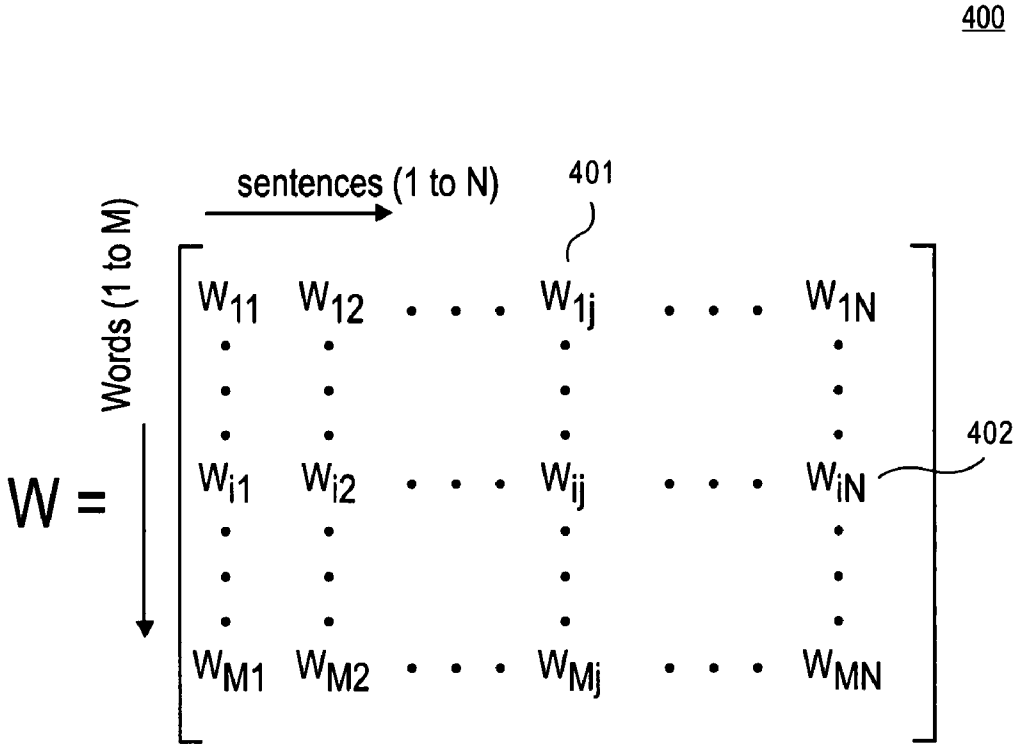

400
sentences (1 to N)
401
Words (1 to M)
W =
W11 W12 W1j W1N
Wi1 Wi2 Wij WiN
WM1 WM2 WMj WMN
402

FIG. 4

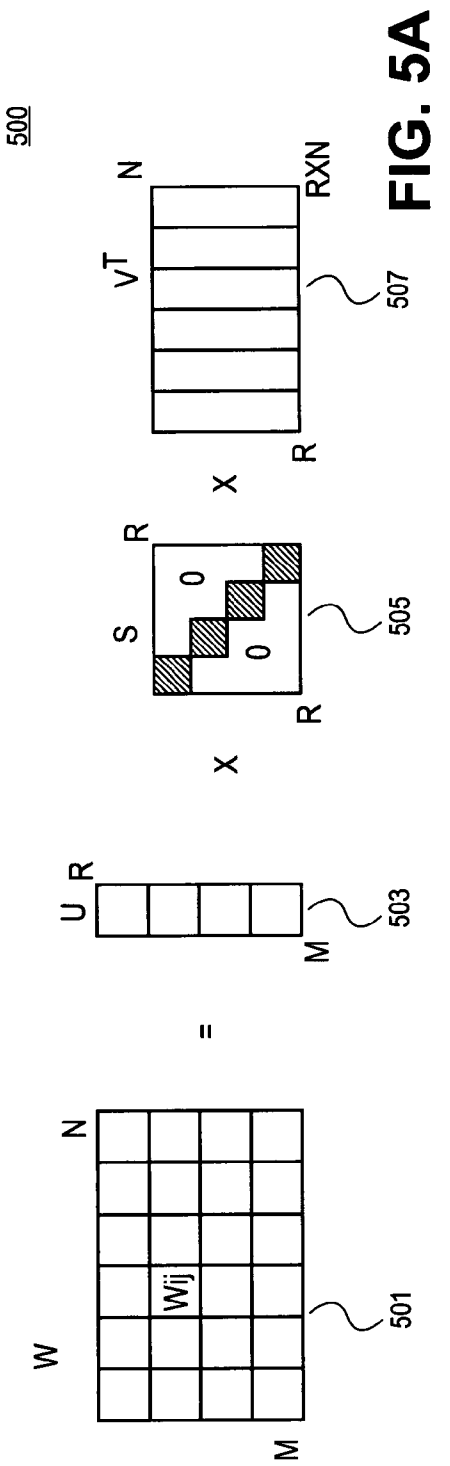

500
W
N
M
Wij
501
=
U
R
M
503
X
S
R
R
0
0
505
X
V^T
N
R
RXN
507

FIG. 5A

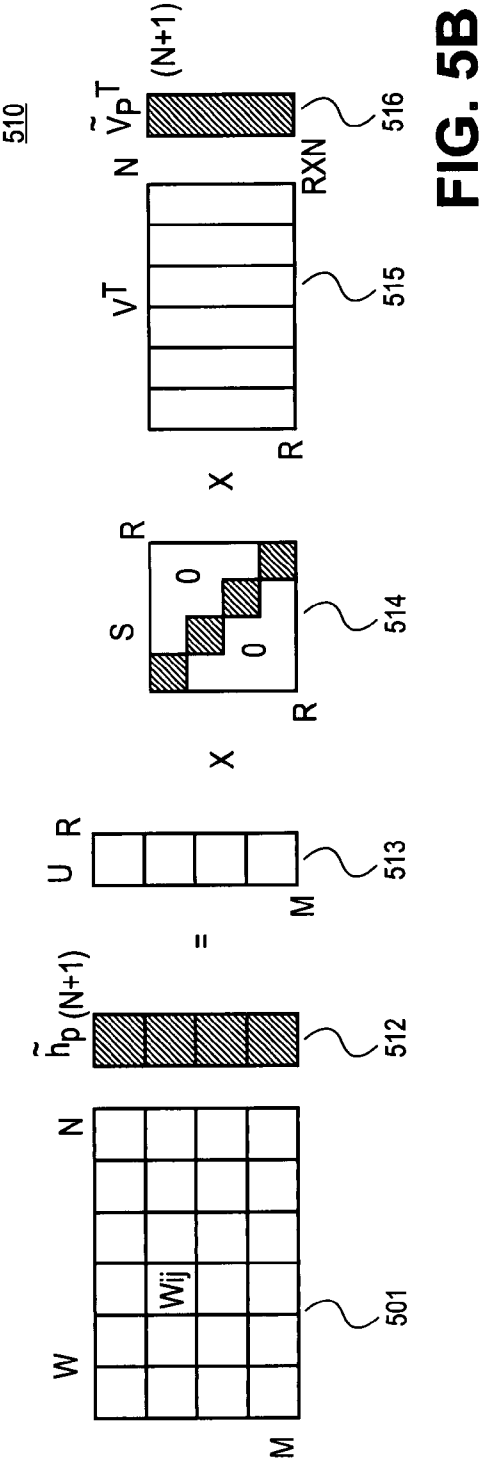

510
W
N
M
Wij
501
$\tilde{h}_p(N+1)$
512
=
U
R
M
513
X
S
R
R
0
0
514
X
V^T
N
R
RXN
515
$\tilde{V}_P^T(N+1)$
516

FIG. 5B

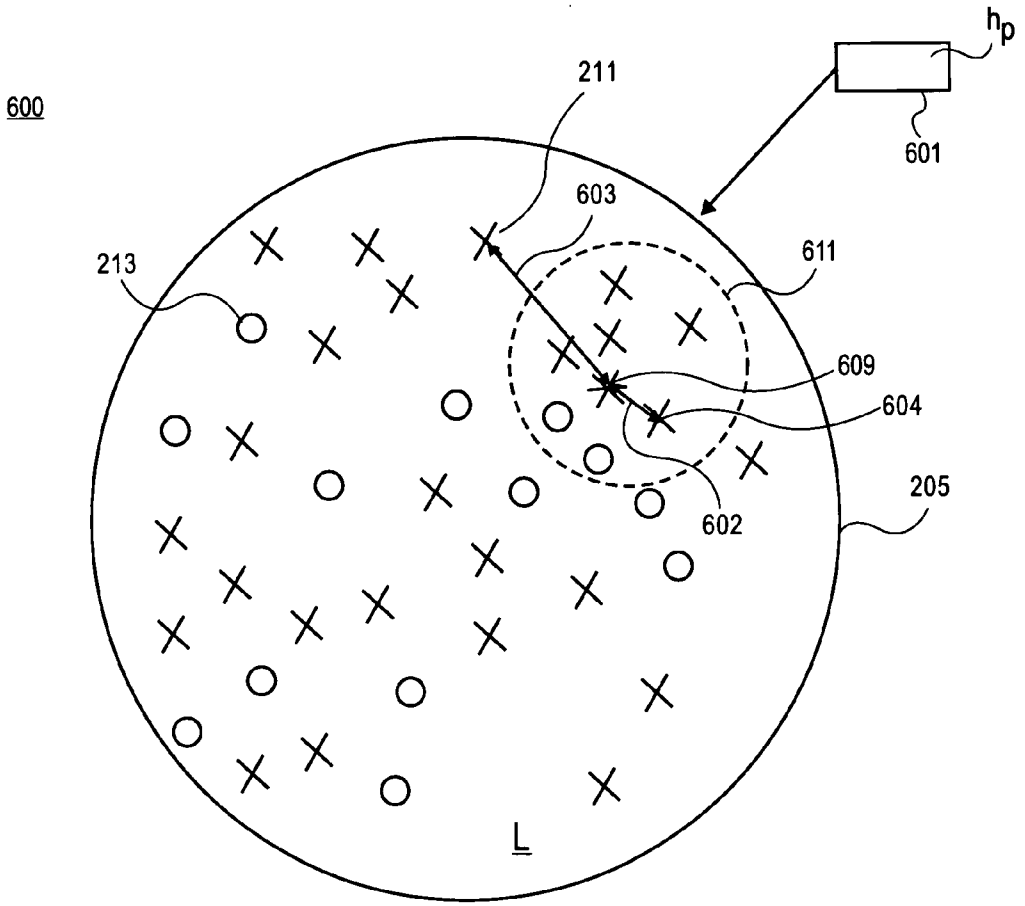


FIG. 6

700

Table I.

701	**jet**/NN propulsion/NN also/RB makes/VBZ flight/NN possible/JJ at/IN extremely/RB high/JJ altitudes/NNS ,/, and/CC even/RB in/IN outer/JJ space/NN these/DT superalloys/NNS are/VBP important/JJ components/NNS of/IN jet/NN engines/NNS and/CC spacecraft/NN
702	high-speed/JJ **streams**/NNS of/IN the/DT solar/JJ wind/NN appear/VBP as/IN the/DT sun/NN 's/POS activity/NN increases/NNS this/DT device/NN sprays/VBZ **streams**/NNS of/IN vapor/NN that/WDT sweep/VBP gas/NN molecules/NNS out/IN of/IN the/DT enclosed/VBN space/NN grade/NN separations/NNS are/VBP often/RB used/VBN to/TO separate/VB crossing/BVG **streams**/NNS of/IN traffic/NN
703	extremely/RB strong/JJ winds/NNS **blow**/VBP in/IN this/DT layer/NN waterline/NNP and/OC trade/NN winds/NNS **blow**/VBP away/RB from/IN the/DT thirty/CD degrees/NNS latitude/VBP belt/NN similar/JJ winds/NNS that/WDT **blow**/VBP in/IN other/JJ parts/NNS of/IN the/DT world/NN are/VBP called/VBN foehns/NNS
704	the/DT temperature/NN in/IN a/DT thin/JJ layer/NN of/IN the/DT troposphere/NN then/RB increases/VBE with/IN altitude/NN other/JJ parts/NNS of/IN the/DT atmosphere/NN are/VBP above/IN the/DT **troposphere**/NN most/JJS clouds/NNS occur/VBP within/IN the/DT **troposphere**/NN

FIG. 7

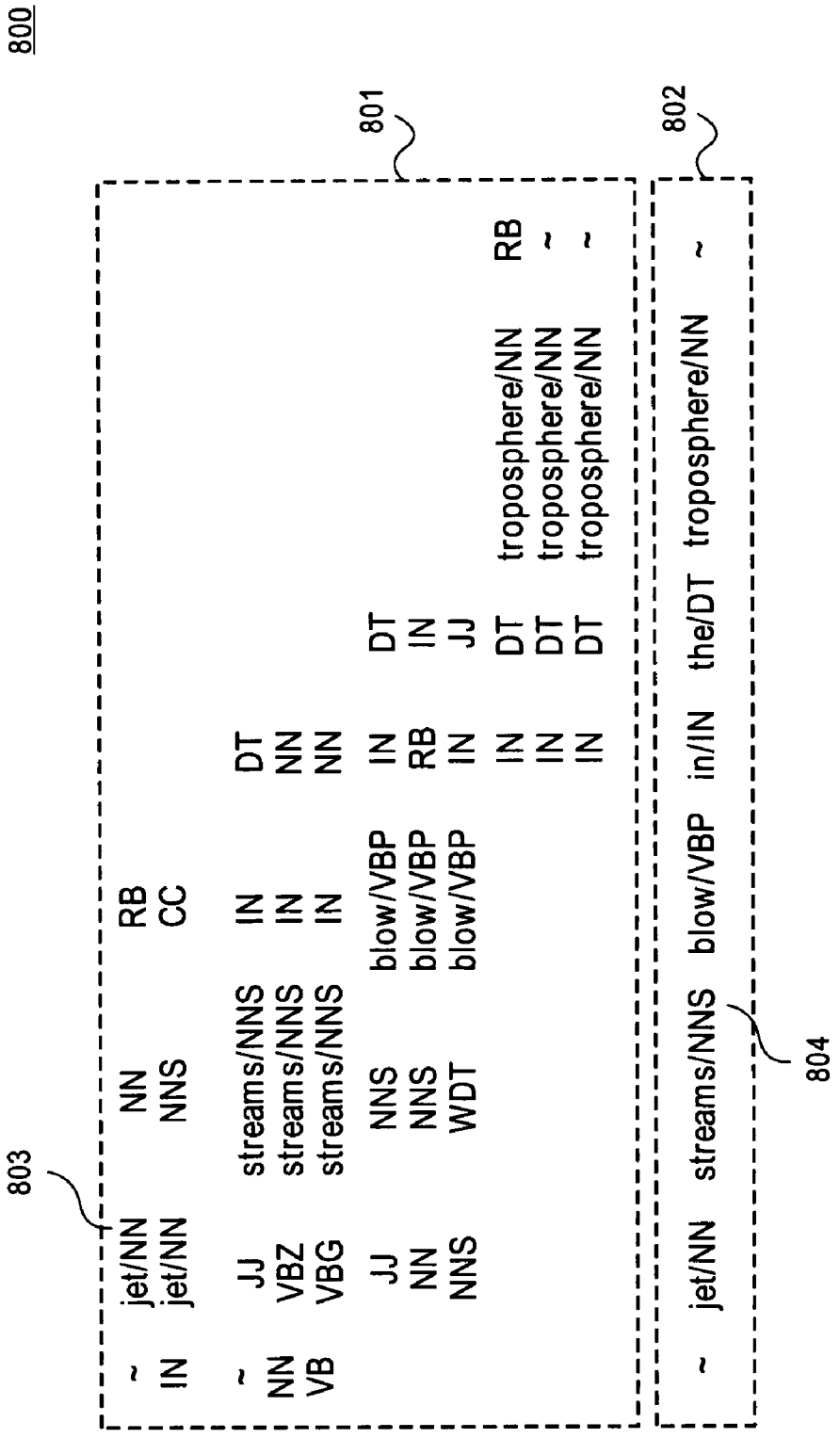

800
801
802
803
804
~ jet/NN NN RB
IN jet/NN NNS CC
~ JJ streams/NNS IN DT
NN VBZ streams/NNS IN NN
VB VBG streams/NNS IN NN
JJ NNS blow/VBP IN DT
NN NNS blow/VBP RB IN
NNS WDT blow/VBP IN JJ
IN DT troposphere/NN RB
IN DT troposphere/NN ~
IN DT troposphere/NN ~
~ jet/NN streams/NNS blow/VBP in/IN the/DT troposphere/NN ~

FIG. 8

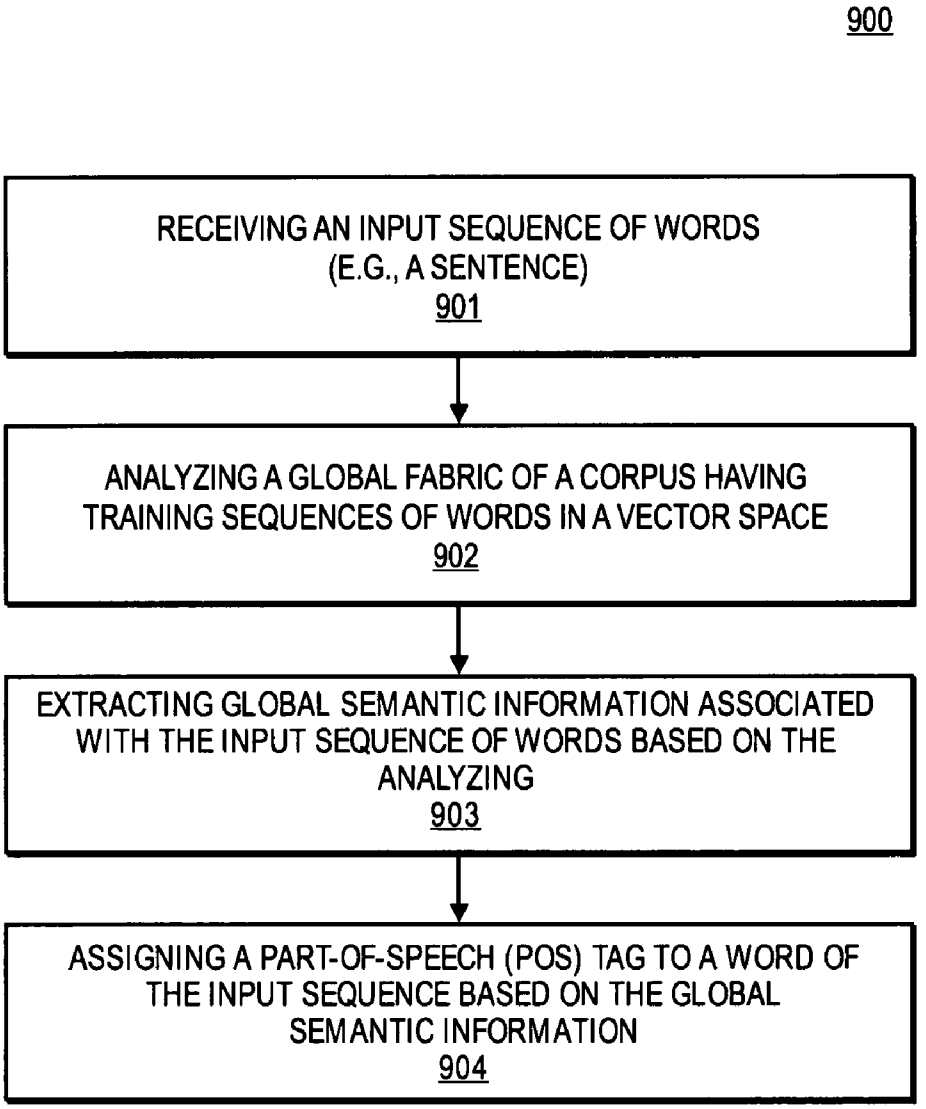

900
RECEIVING AN INPUT SEQUENCE OF WORDS (E.G., A SENTENCE)
901
ANALYZING A GLOBAL FABRIC OF A CORPUS HAVING TRAINING SEQUENCES OF WORDS IN A VECTOR SPACE
902
EXTRACTING GLOBAL SEMANTIC INFORMATION ASSOCIATED WITH THE INPUT SEQUENCE OF WORDS BASED ON THE ANALYZING
903
ASSIGNING A PART-OF-SPEECH (POS) TAG TO A WORD OF THE INPUT SEQUENCE BASED ON THE GLOBAL SEMANTIC INFORMATION
904

FIG. 9

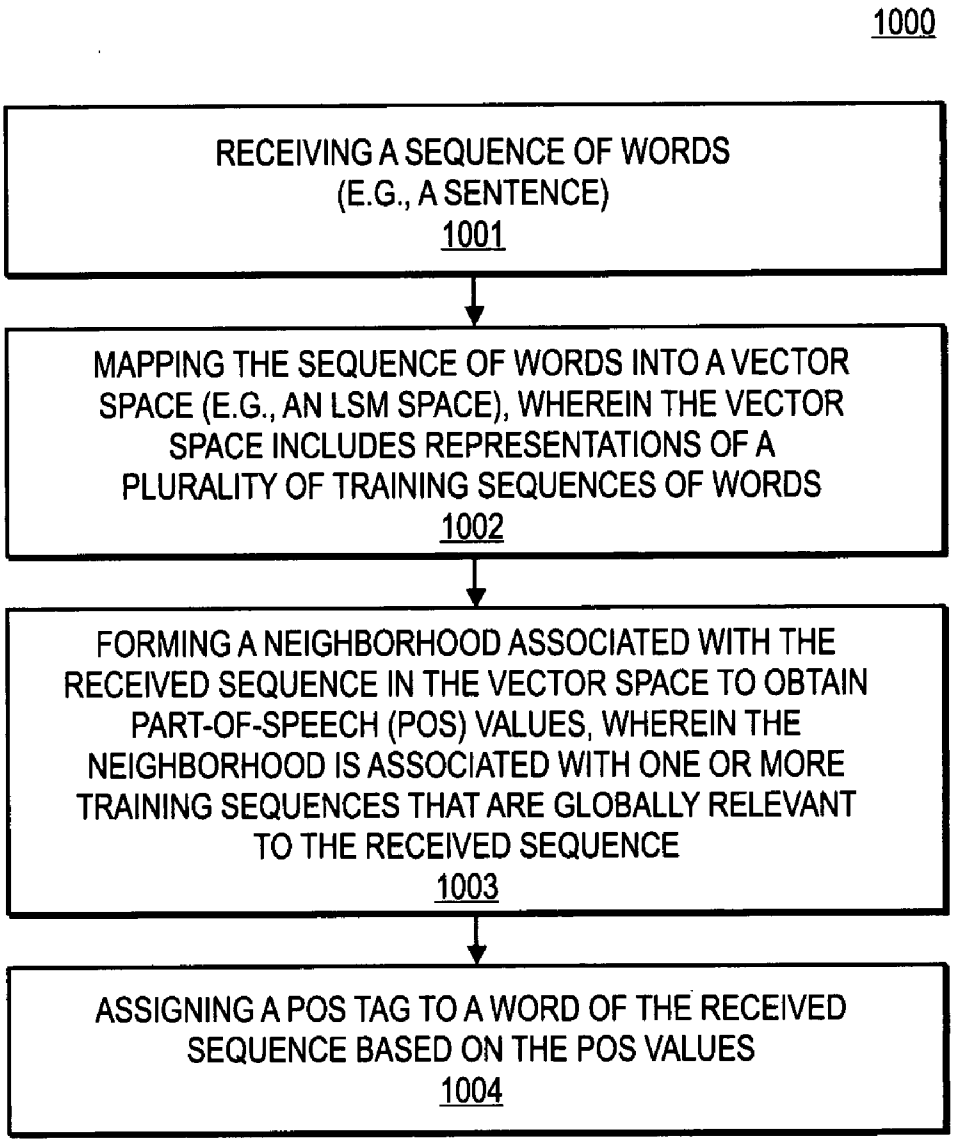

1000
RECEIVING A SEQUENCE OF WORDS (E.G., A SENTENCE)
1001
MAPPING THE SEQUENCE OF WORDS INTO A VECTOR SPACE (E.G., AN LSM SPACE), WHEREIN THE VECTOR SPACE INCLUDES REPRESENTATIONS OF A PLURALITY OF TRAINING SEQUENCES OF WORDS
1002
FORMING A NEIGHBORHOOD ASSOCIATED WITH THE RECEIVED SEQUENCE IN THE VECTOR SPACE TO OBTAIN PART-OF-SPEECH (POS) VALUES, WHEREIN THE NEIGHBORHOOD IS ASSOCIATED WITH ONE OR MORE TRAINING SEQUENCES THAT ARE GLOBALLY RELEVANT TO THE RECEIVED SEQUENCE
1003
ASSIGNING A POS TAG TO A WORD OF THE RECEIVED SEQUENCE BASED ON THE POS VALUES
1004

FIG. 10

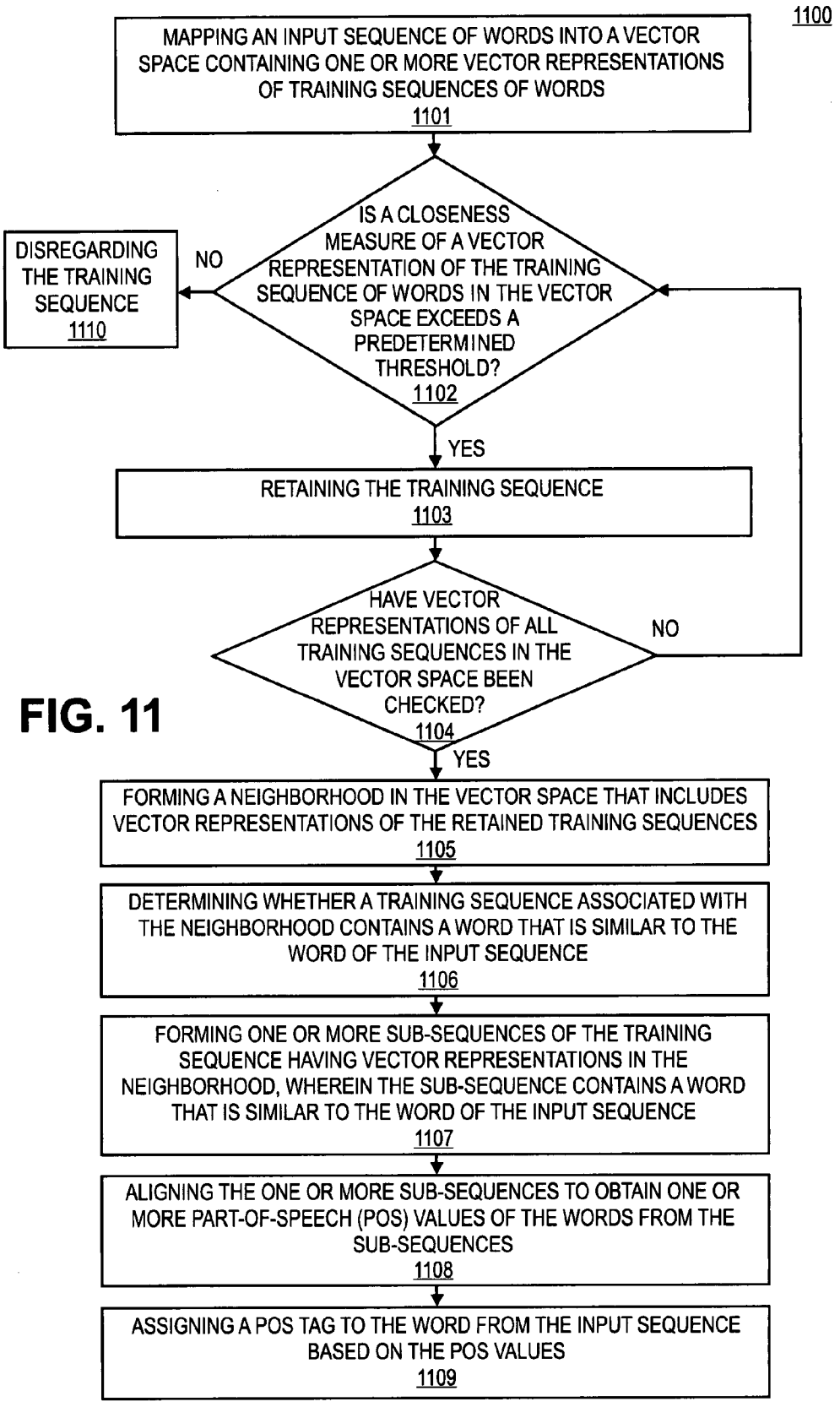

1100
MAPPING AN INPUT SEQUENCE OF WORDS INTO A VECTOR SPACE CONTAINING ONE OR MORE VECTOR REPRESENTATIONS OF TRAINING SEQUENCES OF WORDS
1101
IS A CLOSENESS MEASURE OF A VECTOR REPRESENTATION OF THE TRAINING SEQUENCE OF WORDS IN THE VECTOR SPACE EXCEEDS A PREDETERMINED THRESHOLD?
1102
NO
DISREGARDING THE TRAINING SEQUENCE
1110
YES
RETAINING THE TRAINING SEQUENCE
1103
HAVE VECTOR REPRESENTATIONS OF ALL TRAINING SEQUENCES IN THE VECTOR SPACE BEEN CHECKED?
1104
NO
YES
FORMING A NEIGHBORHOOD IN THE VECTOR SPACE THAT INCLUDES VECTOR REPRESENTATIONS OF THE RETAINED TRAINING SEQUENCES
1105
DETERMINING WHETHER A TRAINING SEQUENCE ASSOCIATED WITH THE NEIGHBORHOOD CONTAINS A WORD THAT IS SIMILAR TO THE WORD OF THE INPUT SEQUENCE
1106
FORMING ONE OR MORE SUB-SEQUENCES OF THE TRAINING SEQUENCE HAVING VECTOR REPRESENTATIONS IN THE NEIGHBORHOOD, WHEREIN THE SUB-SEQUENCE CONTAINS A WORD THAT IS SIMILAR TO THE WORD OF THE INPUT SEQUENCE
1107
ALIGNING THE ONE OR MORE SUB-SEQUENCES TO OBTAIN ONE OR MORE PART-OF-SPEECH (POS) VALUES OF THE WORDS FROM THE SUB-SEQUENCES
1108
ASSIGNING A POS TAG TO THE WORD FROM THE INPUT SEQUENCE BASED ON THE POS VALUES
1109

FIG. 11

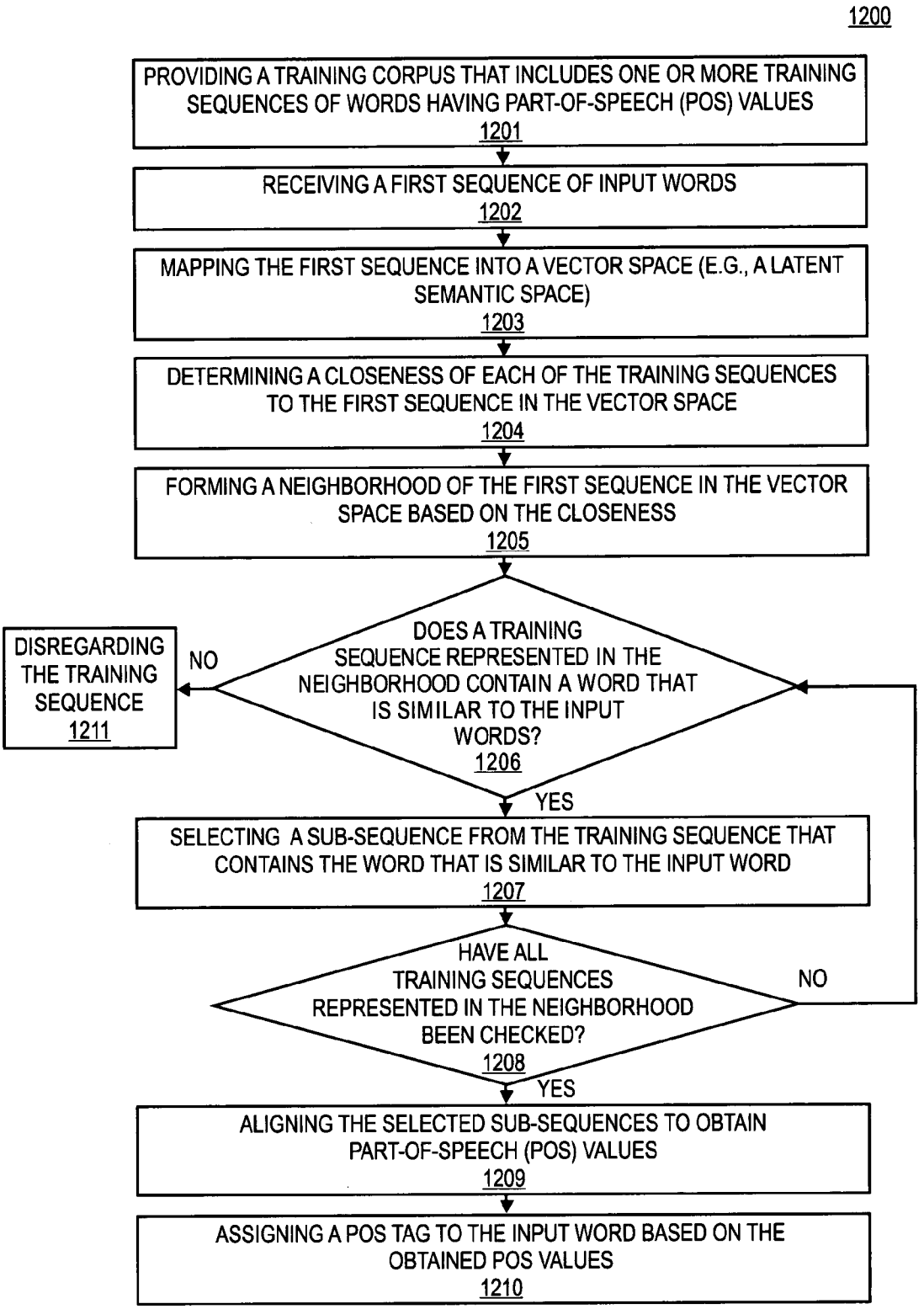

1200
PROVIDING A TRAINING CORPUS THAT INCLUDES ONE OR MORE TRAINING SEQUENCES OF WORDS HAVING PART-OF-SPEECH (POS) VALUES
1201
RECEIVING A FIRST SEQUENCE OF INPUT WORDS
1202
MAPPING THE FIRST SEQUENCE INTO A VECTOR SPACE (E.G., A LATENT SEMANTIC SPACE)
1203
DETERMINING A CLOSENESS OF EACH OF THE TRAINING SEQUENCES TO THE FIRST SEQUENCE IN THE VECTOR SPACE
1204
FORMING A NEIGHBORHOOD OF THE FIRST SEQUENCE IN THE VECTOR SPACE BASED ON THE CLOSENESS
1205
DOES A TRAINING SEQUENCE REPRESENTED IN THE NEIGHBORHOOD CONTAIN A WORD THAT IS SIMILAR TO THE INPUT WORDS?
1206
NO
DISREGARDING THE TRAINING SEQUENCE
1211
YES
SELECTING A SUB-SEQUENCE FROM THE TRAINING SEQUENCE THAT CONTAINS THE WORD THAT IS SIMILAR TO THE INPUT WORD
1207
HAVE ALL TRAINING SEQUENCES REPRESENTED IN THE NEIGHBORHOOD BEEN CHECKED?
1208
NO
YES
ALIGNING THE SELECTED SUB-SEQUENCES TO OBTAIN PART-OF-SPEECH (POS) VALUES
1209
ASSIGNING A POS TAG TO THE INPUT WORD BASED ON THE OBTAINED POS VALUES
1210

FIG. 12

PART-OF-SPEECH TAGGING USING LATENT ANALOGY

COPYRIGHT NOTICES

FIELD OF THE INVENTION

[0002] The present invention relates generally to language processing. More particularly, this invention relates to automatic discovery of the syntactic structure in language.

BACKGROUND

[0003] Part-of-speech ("POS") tagging is used in many natural language processing ("NLP") tasks. As POS tags augment the information contained within words by indicating some of the structure inherent in language, their accuracy is often critical to NLP applications. In text-to-speech (TTS) synthesis POS information is often relied upon to determine how to pronounce a word properly. A word may be pronounced differently depending on a part of speech and/or a tense. For example, a word "read" may be pronounced differently depending on a tense. A word "advocate" may be pronounced differently depending on whether the word "advocate" is a noun or verb.

[0004] POS tags may help to decide whether the synthesized word should be accented or not. For example, a noun may be accented more than a verb. Accordingly, POS tags may greatly influence how natural synthetic speech sounds. Typically, a POS tag is assigned to a word based on the local information contained in a text. For example, to assign a POS tag to a word in the text, adjacent words are typically considered.

[0005] Conceptually, the POS tags may be assigned to words in a text according to predetermined rules. For example, if a determiner, such as "the" or "a", precedes a word in the text, than the word may be assigned an adjective or a noun tag. In another example, if word "to" precedes a word in the text, than the word may be assigned a verb tag.

[0006] In the past, numerous rules were manually generated for the POS tagging. An answer to one rule, however, may conflict with the answer to another rule. Accordingly, the POS tagging may strongly depend on how the rules are ordered. Accordingly, the accuracy of the POS tagging by rules may be poor.

[0007] Current methods of POS tagging involve sophisticated statistical models, such as maximum entropy Markov models ("MEMMs") and conditional random fields ("CRFs"). Both types of modeling rely on a set of feature functions to ensure that important characteristics of the empirical training distribution are reflected in the trained model. These types of modeling, however, may suffer directly or indirectly from the so-called "label bias problem", whereby certain characteristics are unduly favored over other characteristics.

[0008] Hence, the tagging accuracy of both MEMMs and CRFs may depend on how many feature functions are selected and how relevant they are to the task at hand. Such selection may require application-specific linguistic knowledge, complicating deployment across different applications. Moreover, it is basically impossible to specify a set of feature functions that will work well in every environment. For example, a set of feature functions that is selected for the POS tagging of the text from the Wall Street Journal may not be appropriate for the POS tagging of the text from the Word Book Encyclopedia, or from a web blog. Typically, the accuracy of both MEMMs and CRFs may increase as the number of feature functions increases. Increasing the number of feature functions to assign POS tags to words in the text dramatically increases the processing time and/or work load on the processing resources and may be very expensive.

SUMMARY OF THE DESCRIPTION

[0009] Methods and apparatuses to assign part-of-speech tags to words are described. An input sequence of words, for example, a sentence, is received. A global fabric of a training corpus containing training sequences of words is analyzed in a vector space. The vector space may include a latent semantic ("LS") space. Global semantic information associated with the input sequence of words is extracted based on the analyzing. A part-of-speech ("POS") tag is assigned to a word of the input sequence based on POS tags from words in training sequences that are identified using the global semantic information. In one embodiment, analyzing of the global fabric of the training corpus is performed using a latent semantic mapping. In one embodiment, the global semantic information is used to identify which training sequences from the training corpus are globally relevant. In one embodiment, the characteristics of the words of the identified training sequences that are globally relevant to the input sequence are obtained. In one embodiment, the characteristics of the words of the training sequences that are globally relevant to the input sequence include part-of-speech characteristics.

[0010] In one embodiment, an input sequence of words is mapped into a vector space. The vector space may include representations of a plurality of training sequences of words. A neighborhood associated with the input sequence may be formed in the vector space. The neighborhood may represent one or more training sequences of the corpus that are globally relevant to the input sequence. A part-of-speech tag to assign to a word of the input sequence may be determined based on characteristics of the words of the training sequences from the neighborhood.

[0011] In one embodiment, an input sequence is mapped into a vector space, for example, a LS space. The vector space may include representations of a plurality of training sequences of words. A closeness measure between each of the training sequences and the input sequence may be determined in the vector space. One or more training sequences may be selected out of the plurality of the training sequences based on the closeness measure, to form the neighborhood of the input sequence in the vector space. The neighborhood may represent one or more training sequences of a training corpus that are globally relevant to the input sequence. A part-of-speech tag to assign to the word of the input sequence may be determined based on one or more part-of-speech characteristics of words from the training sequences represented in the neighborhood.

[0012] In one embodiment, an input sequence is mapped into a vector space, for example, a LS space. The vector space may include representations of a plurality of training

sequences of words. A neighborhood of the input sequence in the vector space is formed. The neighborhood of the input sequence in the vector space may contain representations of the training sequences that are globally relevant to the input sentence. In one embodiment, determination is made whether a training sequence in the neighborhood contains a word that is similar to an input word of the input sequence. One or more sub-sequences of the training sequence that contain one or more words that are similar to the input words of the input sequence are determined. The one or more sub-sequences that contain the words that are similar to the input words may be aligned to obtain one or more part-of-speech characteristics. One or more part-of-speech tags to assign to one or more words of the input sequence may be determined based on the one or more part-of-speech characteristics.

[0013] Other features will be apparent from the accompanying drawings and from the detailed description which follows.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings in which like references indicate similar elements.

[0015] FIG. **1**A shows a block diagram of a data processing system to assign part-of-speech ("POS") tags to words to perform natural language processing according to one embodiment of invention.

[0016] FIG. **1**B shows a block diagram illustrating a data processing system to assign POS tags to words to perform natural language processing according to another embodiment of the invention.

[0017] FIG. **2** shows an overview of one embodiment of a vector space.

[0018] FIG. **3** shows a schematic that illustrates one embodiment of forming a matrix W using a training corpus and a set of n-grams.

[0019] FIG. **4** illustrates one embodiment of a matrix W that has entries that reflect the extent to which each n-gram appears in the training corpus.

[0020] FIG. **5**A shows a diagram that illustrates a singular value decomposition ("SVD") of a matrix W to construct a vector space according to one embodiment of invention.

[0021] FIG. **6** shows a schematic that illustrates mapping of an input sequence of words into a vector space according to one embodiment of the invention.

[0022] FIG. **7** shows an example of sentence neighborhood according to one embodiment of the invention.

[0023] FIG. **8** shows an example of one embodiment of sequence alignment.

[0024] FIG. **9** shows a flowchart of a method to assign POS tags to words of an input text using latent analogy according to one embodiment of the invention.

[0025] FIG. **10** shows a flowchart of one embodiment of a method to assign POS tags to words.

[0026] FIG. **11** shows a flowchart of one embodiment of a method to form a neighborhood to assign POS tags to words.

[0027] FIG. **12** shows a flowchart of one embodiment of a method to align sub-sequences to assign POS tags to words.

DETAILED DESCRIPTION

[0028] The subject invention will be described with references to numerous details set forth below, and the accompanying drawings will illustrate the invention. The following description and drawings are illustrative of the invention and are not to be construed as limiting the invention. Numerous specific details are described to provide a thorough understanding of the present invention. However, in certain instances, well known or conventional details are not described in order to not unnecessarily obscure the present invention in detail.

[0029] Reference throughout the specification to "one embodiment", "another embodiment", or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the present invention. Thus, the appearance of the phrases "in one embodiment" or "in an embodiment" in various places throughout the specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more embodiments.

[0030] Methods and apparatuses to assign part-of-speech ("POS") tags to words using a latent analogy and a system having a computer readable medium containing executable program code to assign part-of-speech tags to words using a latent analogy are described below. Other methods and other features are also described. A machine-readable medium may include any mechanism for storing information in a form readable by a machine (e.g., a computer). For example, a machine-readable medium includes read only memory ("ROM"); random access memory ("RAM"); magnetic disk storage media; optical storage media; and flash memory devices.

[0031] FIG. **1**A shows a block diagram **100** of a data processing system to assign POS tags to words to perform natural language processing according to one embodiment of invention. Data processing system **113** includes a processing unit **101** that may include a microprocessor, such as an Intel Pentium® microprocessor, Motorola Power PC® microprocessor, Intel Core™ Duo processor, AMD Athlon™ processor, AMD Turion™ processor, AMD Sempron™ processor, and any other microprocessor. Processing unit **101** may include a personal computer (PC), such as a Macintosh® (from Apple Inc. of Cupertino, Calif.), Windows®-based PC (from Microsoft Corporation of Redmond, Wash.), or one of a wide variety of hardware platforms that run the UNIX operating system or other operating systems. For one embodiment, processing unit **101** includes a general purpose data processing system based on the PowerPC®, Intel Core™ Duo, AMD Athlon™, AMD Turion™ processor, AMD Sempron™, HP Pavilion™ PC, HP Compaq™ PC, and any other processor families. Processing unit **101** may be a conventional microprocessor such as an Intel Pentium microprocessor or Motorola Power PC microprocessor.

[0032] As shown in FIG. **1**A, memory **102** is coupled to the processing unit **101** by a bus **103**. Memory **102** can be dynamic random access memory (DRAM) and can also include static random access memory (SRAM). A bus **103** couples processing unit **101** to the memory **102** and also to non-volatile storage **107** and to display controller **104** and to the input/output (I/O) controller **108**. Display controller **104** controls in the conventional manner a display on a display device **105** which can be a cathode ray tube (CRT) or liquid crystal display (LCD). The input/output devices **110** can include a keyboard, disk drives, printers, a scanner, and other input and output devices, including a mouse or other pointing device. One or more input devices **110**, such as a scanner,

keyboard, mouse or other pointing device can be used to input a text for speech synthesis. The display controller **104** and the I/O controller **108** can be implemented with conventional well known technology. An audio output **109**, for example, one or more speakers may be coupled to an I/O controller **108** to produce speech. The non-volatile storage **107** is often a magnetic hard disk, an optical disk, or another form of storage for large amounts of data. Some of this data is often written, by a direct memory access process, into memory **102** during execution of software in the data processing system **113**. One of skill in the art will immediately recognize that the terms "computer-readable medium" and "machine-readable medium" include any type of storage device that is accessible by the processing unit **101**. A data processing system **113** can interface to external systems through a modem or network interface **112**. It will be appreciated that the modem or network interface **112** can be considered to be part of the data processing system **113**. This interface **112** can be an analog modem, ISDN modem, cable modem, token ring interface, satellite transmission interface, or other interfaces for coupling a data processing system to other data processing systems.

[0033] It will be appreciated that data processing system **113** is one example of many possible data processing systems which have different architectures. For example, personal computers based on an Intel microprocessor often have multiple buses, one of which can be an input/output (I/O) bus for the peripherals and one that directly connects the processing unit **101** and the memory **102** (often referred to as a memory bus). The buses are connected together through bridge components that perform any necessary translation due to differing bus protocols.

[0034] Network computers are another type of data processing system that can be used with the embodiments of the present invention. Network computers do not usually include a hard disk or other mass storage, and the executable programs are loaded from a network connection into the memory **102** for execution by the processing unit **101**. A Web TV system, which is known in the art, is also considered to be a data processing system according to the embodiments of the present invention, but it may lack some of the features shown in FIG. **1**A, such as certain input or output devices. A typical data processing system will usually include at least a processor, memory, and a bus coupling the memory to the processor.

[0035] It will also be appreciated that the data processing system **113** is controlled by operating system software which includes a file management system, such as a disk operating system, which is part of the operating system software. One example of operating system software is the family of operating systems known as Macintosh® Operating System (Mac OS®) or Mac OS X® from Apple Inc. of Cupertino, Calif. Another example of operating system software is the family of operating systems known as Windows® from Microsoft Corporation of Redmond, Wash., and their associated file management systems. The file management system is typically stored in the non-volatile storage **107** and causes the processing unit **101** to execute the various acts required by the operating system to input and output data and to store data in memory, including storing files on the non-volatile storage **107**.

[0036] FIG. **1**B shows a block diagram illustrating a data processing system **120** to assign POS tags to words to perform natural language processing according to another embodiment of the invention. As shown in FIG. **1**B, the POS tags are assigned to words to perform a concatenative text-to-speech ("TTS") synthesis. A text analyzing unit **122** receives a text input **121**, for example, one or more sentences, paragraphs, and the like, and analyzes the text to extract words according to one embodiment of the invention. Analyzing unit **122** determines characteristics of a word, for example a pitch, duration, accent, and part-of-speech characteristic according to one embodiment of the invention. The part-of-speech characteristic typically defines whether a word in a sentence is, for example, a noun, verb, adjective, preposition, and/or the like. The POS characteristics may be very informative, and some times are the only way to distinguish a word from the word candidates for speech synthesis. In one embodiment, analyzing unit **122** determines input word's characteristics, such as a pitch, duration, and/or accent based on the POS characteristic of the input word. In one embodiment, analyzing unit **122** analyzes text input **121** to determine a POS characteristic of a word of input text **121** using a latent semantic analogy, as described in further details below with respect to FIGS. **2**-**12**.

[0037] As shown in FIG. **1**B, system **120** includes a training corpus **123** that contains a pool of training words and training word sequences. Training corpus **123** may be stored in a memory incorporated into text analyzing unit **122**, and/or be stored in a separate entity coupled to text analyzing unit **122**. In one embodiment, text analyzing unit **122** determines a POS characteristic of a word from input text **121** by selecting one or more word sequences from the training corpus **123** using latent semantic analogy, as described below. In one embodiment, text analyzing unit **122** assigns POS tags to input words of input text **121** as described in further details below. Generally, the text analyzing unit, such as text analyzing unit **122**, may assign POS tags to input words of the input text, such as input text **121**, for many natural language processing ("NLP") applications, for example, from low-level applications, such as grammar checking and text chunking, to high-level applications, such as text-to-speech synthesis ("TTS") (as shown in FIG. **1**B), speech recognition and machine translation applications.

[0038] As shown in FIG. **1**B, text analyzing unit **122** passes extracted words having assigned POS tags to processing unit **124**. In one embodiment, processing unit **124** concatenates extracted words together, smoothes the transitions between the concatenated words, and passes the concatenated words to a speech generating unit **125** to enable the generation of a naturalized audio output **126**, for example, an utterance, spoken paragraph, and the like.

[0039] Given a natural language sentence comprising z words, POS tagging aims at annotating each observed word w_i with some suitable part-of-speech p_i, (each typically associated with a particular state s_i, $1 \leqq i \leqq z$). Representing the overall sequence of words by W and the corresponding sequence of POS by P, typical statistical models try to maximize the conditional probability Pr (P/W) over all possible POS sequences P.

[0040] Maximum entropy models such as MEMMs and CRFs approach this problem by considering state-observation transition distributions expressed as log-linear models of the form:

$$Prs_i(s_{i+1} \mid w_i) = \frac{1}{Z(S, W)} \exp\left[\sum_k \lambda_k f_k(C_i, w_i)\right] \quad (1)$$

which represent the probability of moving from state s_i to state s_{i+1} conditioned upon observation w_i. In expression (1), $f_k(C_i, w_i)$ is a feature function of the current observation and any entity belonging to the appropriate clique C_i of the underlying undirected graph, λ_k is a parameter to be estimated, and Z(S, W) is a normalization factor. Each feature function expresses some characteristic of the empirical training distribution, which is deemed important to require of the trained model as well. For natural language, however, it is essentially impossible to achieve a systematic and exhaustive determination of empirical importance. In practice, this means that the accuracy of models like expression (1) is largely contingent on the pertinence of the feature functions selected for the particular task at hand.

[0041] The usual way out of this dilemma is to throw in as many feature functions as computational resources will allow. Even so, due to the intrinsic lopsided sparsity of language, many distributional aspects will still be missing. And therefore, in those specific contexts where they happen to matter, this may result in erroneous tagging. Consider, for example, the sentence:

Jet streams blow in the troposphere. (2)

The correct tagging for sentence (2) would read as follows:

jet/NN streams/NNS blow/VBP in/IN the/DT tropo-sphere/NN (3)

[0042] In expression (3) POS tag "NN" may indicate a noun singular, POS tag "NNS" may indicate a noun with a plural, POS tag "VBP" may indicate a verb in present tense, and POS tag "IN" may indicate a preposition. POS tags are known to one of ordinary skill in the art of natural language processing.

[0043] The CRF model provides, however, the following POS tagging:

jet/NN streams/VBZ blow/NN in/IN the/DT tropo-sphere/NN (4)

[0044] As expression (4) indicates, CRF model incorrectly resolves the inherent POS ambiguity in the sub-sequence "streams blow." As shown in (4), word "streams" is assigned a tag VBZ that is third person verb instead of tag NN, as shown in (3) Word "blow" is assigned a tag NN instead of tag VBP, as shown in (3). The problem is that from purely a syntactic viewpoint both interpretations are perfectly acceptable (a frequent situation due to the many dual noun-verb possibilities in English).

[0045] What would clearly help in this case is taking into account the semantic information available. Indeed the word "troposphere," for example, would seem to make the verbal usage of "blow" substantially more likely. That is, the semantic of the sentence (2) can be used to disambiguate between two sequences of words, such as sequence (3) and sequence (4). The semantic information may include a general topic of the sentence and meaning of the words in the sentence.

[0046] For example, the semantic information may be obtained from determination whether words "jet" and "streams" mostly co-occur in a database, such that word "jet" is in most of the times accompanied by word "streams" and vice versa. If the words "jet" and "streams" mostly co-occur, then it means that "jet stream" is a compound. That is, the meaning of the words "jet" and "streams" in the input sentence (2) can be determined. The POS tags may be assigned to words "jet" and "streams" based on the determined meaning of the words, and/or the general topic of the sentence. Tagging using latent analogy may be an attempt to systematically generalize this observation, as described in further detail below.

[0047] Semantic information can be extracted from an analysis of the global fabric of the training corpus of word sequences. In one embodiment, the analysis of the global fabric of the training corpus is performed using latent semantic mapping. For each sequence of words under consideration, a neighborhood of globally relevant training word sequences may be generated. For example, the neighborhood of the globally relevant training word sequences may be the training sequences that belong to the same general topic, as the sequence of words under consideration.

[0048] The POS characteristics of the words of the globally relevant training word sequences from the neighborhood may be extracted. The POS characteristics of the globally relevant training sequences may be used to assign POS tags to the words of the sequence under consideration. The POS disambiguation may emerge automatically as a by-product of latent semantic mapping ("LSM")-based semantic consistency, which practically bypasses the need for an explicit linguistic knowledge. Additionally, POS tagging using latent analogy takes substantially less time than currently available methods, such as MEMMs and CRFs, described above.

[0049] FIG. **9** shows a flowchart of a method to assign POS tags to words of an input text using latent analogy according to one embodiment of the invention. Method **900** begins with operation **901** that includes receiving an input sequence of words, as described above with respect to FIG. **1**B. In one embodiment, an input sequence of words may be a sentence, paragraph, or any other sequence of words. Method **900** continues with operation **902** that involves analyzing a global fabric of a training corpus having training sequences of words in a vector space, for example, a latent semantic ("LS") space.

[0050] In one embodiment, the analysis of the global fabric of the training corpus is performed using latent semantic mapping. In one embodiment, the analyzing of the global fabric of the training corpus in the vector space comprises mapping the input sequence into the vector space, and forming a neighborhood associated with the input sequence in the vector space. In one embodiment, the neighborhood associated with the input sequence of words in the vector space represents one or more training sequences that are globally relevant to the input sequence, as described in further detail below. In one embodiment, the one or more training sequences that are globally relevant to the input sequence are the training sentences that have the substantially the same general topic, as the input sequence of words. In one embodiment, the analyzing of the global fabric of the training corpus in the vector space comprises determining a closeness measure between the training sequences and the input sequence in the vector space, as described in further detail below.

[0051] FIG. **2** shows an overview **200** of one embodiment of a vector space. As shown in FIG. **2**, a training corpus **201** includes a collection T of N training sequences of words (for example, sentences) hj **207**, where N may be any number. In one embodiment, N ranges from about 100 to about 50,000. As shown in FIG. **2**, a set V **203** associated with training corpus **201** includes M n-grams gi **209** observed in the collection T including proper markers for punctuation, etc, where M may be any number. In one embodiment, M ranges from about 1,000 to about 1,000,000. Typically, n-grams gi **209** are words, and strings of words, such as bigrams, trigrams, and the like. N-grams are known to one of ordinary

skill in the art of language processing. In one embodiment, the set V **203** includes the underlying vocabulary (e.g., words) if n=1. In one embodiment, each word in the N training sequences of words hj **207** has been annotated with a POS tag. As shown in FIG. **2**, the training corpus **201** and an associated set V **203** of M n-grams gi observed in the training corpus T **201** are mapped into a vector space L **205**, whereby each sequence hj in a collection T and each n-gram gi in set V **203** is represented by a vector.

[0052] As shown in FIG. **2**, vector space **205** includes vector representations of training sequences of words hj **207**, such as a vector representation **211** illustrated by a cross, and vector representations of n-grams gi **209**, such as a vector representation **213** illustrated by a circle. The continuous vector space L **205** is semantic in nature, because the "closeness" of vectors in the space L **205** is determined by the global pattern of the language used in the training corpus **201**, as opposed to local specific constructs. For example, two words whose representations are "close" (in some suitable metric) tend to appear in the same kind of sentences, whether or not they actually occur within identical word contexts in those sentences. Two word sequences (e.g., sentences) whose representations are "close" tend to convey the same semantic meaning, whether or not they contain substantially the same word constructs. More generally, word and sentence vectors **213** and **211** associated with words **209** and sentences **207** that are semantically linked are also "close" in the space L **205**. In one embodiment, vector space L **205** is a latent semantic ("LS") space.

[0053] FIG. **3** illustrates one embodiment of forming a matrix W using a training corpus and a set of n-grams. For an example shown in FIG. **3** n-grams are words (n=1). In one embodiment, matrix W is formed to contain elements that reflect how many times each n-gram from set **203** appears in the training corpus T **201**. As shown in FIG. **3**, matrix W may be constructed such that each unit of training data, for example, the words of sentence "The cat ate the cheese" may be arranged in a column **301**. As shown in FIG. **3**, counts **305**, **307**, **309**, and **311** reflect the extent to which each word appears in the sentence "The cat ate the cheese". As shown in FIG. **3**, count **311** reflects the fact that word "the" appears in the sentence twice, and counts **305**, **307**, and **309** reflect the fact that corresponding words "cat", "ate" and "cheese" appear in the sentence once.

[0054] FIG. **4** illustrates one embodiment of a matrix W that has entries that reflect the extent to which each n-gram from set **203** appears in the training corpus T **201**. As shown in FIG. **4**, matrix W contains (M×N) entries wij that may reflect the extent to which each n-gram gi **207** ∈ V **203** appeared in each sentence hj **207**∈ T **201**. As shown in FIG. **4**, 1 to N columns of matrix W, such as column **401**, correspond to sequences of words hj **207**, for example, sentences. As shown in FIG. **4**, 1 to M rows of matrix W, such as row **402**, correspond to n-grams gi **207**, for example, words, bigrams, such as "Hong Kong" and trigrams, such as "New York City".

[0055] Each entry w_{ij} of matrix W may be expressed as follows:

$$w_{ij} = (1 - \varepsilon_i)\frac{c_{ij}}{n_j}, \tag{5}$$

where c_{ij} is the number of times g_i occurs in sentence h_j, n_j is the total number of n-grams present in this sentence, and ϵ_i, is the normalized entropy of g_i, in V **203**. The global weighting implied by $1-\epsilon_i$, reflects the fact that two n-grams appearing with the same count in a particular sentence do not necessarily convey the same amount of information; this is subordinated to the distribution of the n-grams in the entire set V **203**. That means, for example, that for a word like 'the', which occurs in almost every sentence, normalized entropy ϵ_i would be very close to 1, which means that global weighting implied by $1-\epsilon_i$ would be very close to zero, and therefore may be not informative. For a word that has normalized entropy ϵ_i close to zero, global weighting implied by $1-\epsilon_i$ would be close to one, meaning that this word may be informative.

[0056] FIG. **5**A shows a diagram that illustrates a singular value decomposition ("SVD") of a matrix W, as shown in FIG. **4** to construct a vector space **205**, as shown in FIG. **2** according to one embodiment of invention. A singular value decomposition ("SVD") of (M×N) matrix W reads as follows:

$$W=U\,S\,V^T, \tag{6}$$

where U is the (M×R) left singular matrix **503** with row vectors u_i ($1 \leqq i \leqq M$), S is the (R×R) diagonal matrix **505** of singular values $s_1 \geqq s_2 \geqq \ldots \geqq s_R \geqq 0$, V is the (N×R) right singular matrix **507** with row vectors v_J ($1 \leqq j \leqq N$), wherein R<<M, N is the order of the decomposition, and ^{T} denotes matrix transposition. Both left and right singular matrices U **503** and V **507** are column-orthonormal, i.e., $U^TU=V^TV=I_R$ (the identity matrix of order R). Thus, the column vectors of matrices U and V each define an orthomormal basis for the space of dimension R spanned by the (R-dimensional) u_i's and v_j's. This space may be referred as a vector space, such as vector space **205** of FIG. **2**. In one embodiment, vector space L **205** is a latent semantic space.

[0057] The basic idea behind (6) is that the rank-R decomposition captures the major structural associations in W and ignores higher order effects. Hence, the relative positions of the sentence vector representations (anchors) in the vector space reflect a parsimonious encoding of the semantic concepts used in the training data. This means that any input sequence of words; e.g., a sentence, mapped onto a vector space "close" (in some suitable metric) to a particular sentence anchor would be expected to be closely related to the corresponding training sentence, and any training sequence of words (e.g., sentence) whose representation ("anchor") is "close" to a vector representation of input sequence of words in the space L would tend to be related to this input sentence. This offers a basis for determining sentence neighborhoods.

[0058] Referring back to FIG. **9**, method **900** continues with operation **903** that involves extracting global semantic information associated with the input sequence of words based on the analyzing. The global semantic information may be, for example, at least a surface meaning of the input sentence, such as a topic of the input sentence, and meanings of the words in the input sentence. In one embodiment, the global semantic information associated with the input sequence of words is used to identify which one or more words of the training sequences from the training corpus are globally relevant to the input sequence. Method **900** continues with operation **904** that involves identifying one or more words in the training sequences of words in the vector space that are associated with the global semantic information. In one embodiment, the identified training sequences of words are globally semantically relevant to the input sequence of words. Method **900** continues with operation **905** that

involves assigning a part-of-speech tag to a word of the input sequence based on POS tags from the identified one or more words in the training sequences.

[0059] FIG. **6** illustrates mapping of an input sequence of words into a vector space according to one embodiment of the invention. As shown in FIG. **6**, vector space L **205** includes vector representations of training sequences of words hj, such as vector representation **211** illustrated by a cross, and vector representations of n-grams gi of the set V, such as a vector representation **213** illustrated by a circle. Two word sequences (e.g., sentences) whose representations are "close" in space **603** tend to convey the substantially the same semantic meaning. As shown in FIG. **6**, an input sequence of words hp **601** is mapped into vector space **205**. The mapping of the input sequence **601** into vector space **205** encodes the semantic information. That is, the position of the input sequence **609** in vector space **205** is driven by the meaning of the input sentence, and therefore may fall into a cluster (not shown) in the vector space **205** that defines the topic of the input sentence **609**. As shown in FIG. **6**, a neighborhood **611** associated with the input sequence **601** in the vector space **205** is formed. Neighborhood **611** represents one or more training sequences of words that are globally relevant, for example, have the substantially similar topic as the input sequence **601**. As shown in FIG. **6**, neighborhood **611** includes vector representations such as a vector representation **604**, of training sequences that are globally relevant to the input sequence, as described in further detail below. That is, mapping of the input sequence **601** to the LS space **205** is performed to evaluate which training sequences from the training corpus are globally relevant to the input sentence **601**.

[0060] FIG. **5**B is a diagram similar to the diagram of FIG. **5**A that illustrates mapping of an input sequence of words (e.g., a sentence) into a latent semantic mapping ("LSM") vector space according to one embodiment of the invention. An input sequence not seen in the training corpus, for example sentence hp (where p>N) may be mapped into a vector space **603** of FIG. **6** as follows. For each n-gram in training corpus $^{\tau}$ **201**, the weighted counts w_{ip} with j=p are computed according to expression (5) for sentence hp. The resulting feature vector, a column vector of dimension M, can be thought of as an additional (N+1) column **512** of the matrix W **511**.

[0061] In one embodiment, if the input sequence of words is globally relevant to training sequences of words, for example, the input sentence has substantially the same style, general topic, matrices U **513** and S **514** will be substantially similar to matrices U **503** and S **505**. Therefore, assuming the matrices U and S do not change appreciably, so that matrix U **513** is substantially similar to matrix U **503**, and matrix S **514** is substantially similar to matrix S **505**, the SVD expansion (6) will read as follows:

$$\hbar_p = US\tilde{v}_p^T \tag{7}$$

where the R-dimensional vector $\tilde{v}_p^T$ acts as an additional (N+1) column **516** of the matrix V^T. This in turn leads to the definition:

$$\overline{\nabla}_p = \tilde{v}_p S = \hbar_p^T U \tag{8}$$

[0062] FIG. **10** shows a flowchart of one embodiment of a method to assign POS tags to words. Method **1000** begins with operation **1001** that involves receiving a sequence of words (e.g., a sentence), as described above. At operation **1002** the received sequence is mapped into a vector space, as described above. In one embodiment, the vector space is an LSM space. In one embodiment, the vector space includes representations of a plurality of training sequences of words from a training corpus, as described above. Next, method **1000** continues with operation **1003** that involves forming a neighborhood associated with the received sequence of words in the vector space to obtain POS characteristics, for example, POS values. In one embodiment, the neighborhood is associated with one or more training sequences that are globally relevant to the received sequence.

[0063] In one embodiment, one or more training sequences that are globally relevant to the received sequence of words are selected from the plurality of training sequences selected to form the neighborhood, and the training sequences that are not globally relevant to the received sequence of words are rejected. In one embodiment, a closeness measure, for example, a distance, between representations of a training sequence of the plurality of the training sequences and the input sequence in the vector space is determined to form the neighborhood. A training sequence may be selected out of the plurality of the training sequences based on the closeness measure, as described in further detail below. Next, at operation **1004**, a POS tag is assigned to a word of the received sequence based on the POS characteristics (e.g., POS values) obtained from the neighborhood.

[0064] Referring back to FIG. **6**, neighborhood **611** is formed based on a closeness measure between vector representations of the input sequence **609** and training sequence in vector space **205**. In one embodiment, the closeness measure is associated with a distance between vector representations of the input sequence **609** and each of the training sequences **211** in vector space **205**. As shown in FIG. **6**, closeness measures **602** and **603** between vector representations of each of the training sequence and input sequence **609** in vector space **205** are determined. As shown in FIG. **6**, training sequence **604** is selected for neighborhood **611**, and training sequence **211** is not selected based on the closeness measure. The closeness measure determines global relevance of the each of the training sequences to the input sequence.

[0065] In one embodiment, each of the closeness measures **602** and **603** are compared to a predetermined threshold. The training sequence may be selected if the closeness measure **602** exceeds the predetermined threshold. The training sequence **211** may be rejected if closeness measure **603** is less or equal to the predetermined threshold. The predetermined threshold may be chosen depending on a particular application or task at hand. In another embodiment, to form neighborhood **611**, the training sequences are ranked according to their closeness measures to the input sequence in vector space **205**. The training sequence that has a rank equal or higher than a predetermined rank may be selected to form neighborhood **611**, and the training sequence that has the rank lower than the predetermined rank may be rejected. The predetermined rank may be any number 2, 3, . . . N and may be chosen according to a particular application or task at hand.

[0066] Referring back to expression (8), it remains to specify a suitable closeness measure to compare $\overline{\nabla}_p$ to each of the $\overline{v}_j$'s. In one embodiment, the closeness measure is a Euclidian distance between vector representation $\overline{\nabla}_p$ of the input sequence and each of the vector representations $\overline{v}_j$ of the training sequences. In another embodiment, the closeness measure is the cosine of the angle between them ("cosine distance"). For example, for each of the training sequences the closeness measure to the vector representation of the input sequence **609** may be calculated as follows:

$$K(\tilde{v}_p, \tilde{v}_j) = \cos(v_p S, v_j S) = \frac{\tilde{v}_p S^2 v_j^T}{\|\tilde{v}_p S\| \|v_j S\|}, \quad (9)$$

for any $1 \leqq j \leqq N$. Using (9), all training sentences can be ranked in decreasing order of closeness to the representation of the input sentence **609**. The associated sentence neighborhood **609** may be formed by retaining only those training sequences whose closeness measure is higher than a predetermined threshold.

[0067] FIG. **7** shows an example of sentence neighborhood according to one embodiment of the invention. As shown in FIG. **7**, a Table I (**701**) contains an actual sentence neighborhood for an example input sentence (2). As shown in Table I, a sentence neighborhood, such as neighborhood **611**, includes training sentences that are globally relevant to an input sentence, such as input sentence "Jet streams blow in the troposphere". As shown in FIG. **7**, training sentences are grouped according to reference words that are substantially the same as the words from the input sentence. Group **701** includes training sentences having word "jet" from input sequence (2), group **702** includes training sentences having word "streams" from input sequence (2), group **703** includes training sentences having word "blow" from input sequence (2), and group **704** includes training sentences having word "troposphere" from input sequence (2).

[0068] FIG. **11** shows a flowchart of one embodiment of a method to form a neighborhood to assign POS tags to words. Method starts with operation **1101** that involves mapping an input sequence of words into a vector space containing one or more vector representations of training sequences of words. Next, at operation **1102** a determination is made whether a closeness measure of a vector representation of the training sequence of words in the vector space exceeds a predetermined threshold. If the closeness measure of the vector representation of the training sequence of words in the vector space exceeds the predetermined threshold, the training sequence is retained at operation **11103**. If the closeness measure of the vector representation of the training sequence of words in the vector space is less than the predetermined threshold, the training sequence is disregarded at operation **1110**. That is, the training sequence of words that is globally not relevant to the input sequence of words is disregarded.

[0069] Next, a determination is made whether closeness measures of vector representations of all training sequences have been checked at operation **1104**. If not all training sequences have been checked, the operation **1102** method **1100** returns to operation **1102**. If closeness measures of vector representations of all training sequences have been checked, method **1100** continues with operation **1105** that involves forming a neighborhood in the vector space that includes representations of the retained training sequences. Next, at operation **1106**, a determination is made whether a training sequence represented in the neighborhood contains a word that is substantially similar to (e.g., the same as) the word of the input sequence of words. Next, operation **1107** is performed that includes forming one or more sub-sequences of the training sequence having vector representation in the neighborhood. The sub-sequences contain the words that are substantially similar to the words of the input sequence. Next, one or more sub-sequences are aligned at operation **1108** to obtain one or more POS characteristics (e.g., values) of the words from the sub-sequences. Method **1100** continues with operation **1109** that involves determining a POS tag for the word from the input sequence based on the obtained POS characteristics (e.g., POS values).

[0070] Referring back to FIG. **6**, as set forth above, neighborhood **611** represents labeled training sequences of words having POS tags. Therefore, associated POS sequences are readily available from the labeled training corpus. In principle, each of these POS sequences contains at least one sub-sequence which is germane to the input sentence. Thus, the final POS sequence can be assembled by judicious alignment of appropriate POS sub-sequences from the sentence neighborhood.

[0071] FIG. **8** illustrates an example of one embodiment of sequence alignment. Referring to example input sentence (2), and proceeding word by word, the POS sub-sequences from entries in the sentence neighborhood, for example, as shown in FIG. **7**, are collected in table **801**. As shown in table **801**, the POS sub-sequences contain the relevant reference words, such as "jet", "streams", "blow", "in", and "troposphere" from the input sequence of words (2). It may be necessary to retain only (2K+1) POS in each sub-sequence, centered around that of the current input word. That is, around each word of the input sentence the sub-sentences may be selected out of globally relevant training sentences, which contain that word of the input sentence. K is referred as the size of the local scope. For example shown in FIG. **8**, the local scope is set to K=2. Proceeding left-to-right along the input sentence, such as sentence (2), we thus obtain a set of POS characteristics, such as a POS value **803**, for each word, where each POS value is substantially consistent with global semantic information extracted from the training corpus and germane to the input sentence, such as sentence (2). A POS tag for each of the words of the input sentence is determined based on POS characteristics of the words from the sub-sequences contained in the neighborhood, such as neighborhood **611**. In one embodiment, a POS tag for each of the words of the input sentence is determined by computing the maximum likelihood estimate for every word of the input sentence using the obtained POS value counts from, for example, table **801**. The resulting POS tags, such as POS tag **804**, for each of the words from the input sequence are shown in table **802**.

[0072] The final POS sequence may read as follows:

Jet/NN streams/NNS blow/VBP in/IN the/DT troposphere/NN (10)

[0073] In one embodiment, when the number of one POS values and the number of another POS values that label the words from the sub-sequences are substantially equal, for example, when 50% of the POS values represent nouns and 50% of POS values represent verbs, then the statistics from the whole training corpus can be used to determine the proper POS tag for the input word. A comparison with (3) and (4) shows that POS tagging using latent analogy is able to satisfactorily resolve the inherent POS ambiguity discussed previously. This bodes well for its general deployability across a wide range of applications.

[0074] FIG. **12** shows a flowchart of one embodiment of a method **1200** to align sub-sequences to assign POS tags to words. At operation **1201** a training corpus that includes one or more training sequences of words having POS characteristics (e.g., POS values) is provided, as described above. Method **1200** continues with operation **1202** that involves receiving a sequence of input words, as described above. At operation **1203** the sequence of input words is mapped in a vector space, for example, a LS space, as described above. The closeness of each of the training sequences to the

sequence of input words is determined at operation **1204**, as described above. Next, operation **1205** that involves forming a neighborhood of the sequence of input words in the vector space is formed based on the closeness. The neighborhood represents one or more training sequences, as described above.

[0075] At operation **1206** determination is made whether the training sequence represented in the neighborhood contains a word that is substantially similar to the input word. If the training sequence does not contain the word that is substantially similar to the input word, the training sentence is disregarded at operation **1211**. If the training sequence contains the word that is substantially similar to the input word, the training sentence is retained, and a sub-sequence of such training sequence is selected that contains the word that is substantially similar to the input word at operation **1207**.

[0076] Next, at operation **1208** determination is made have all or a predetermined number of training sequences of words represented in the vector neighborhood been checked. If not all or predetermined number of training sequences represented in the neighborhood have been checked, method **1200** returns to operation **1206**. If all or a predetermined number of training sequences represented in the neighborhood have been checked, method **1200** continues at operation **1209** that involves aligning the selected sub-sequences to obtain POS characteristics (e.g., POS values) of the words from the sub-sequences. Next, assigning a POS tag to the input word is performed based on the obtained one or more POS characteristics (e.g., POS values) of the word from the sub-sequences that is substantially similar to the input word.

[0077] Some portions of the preceding detailed descriptions have been presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the ways used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of operations leading to a desired result. The operations are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

[0078] It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the above discussion, it is appreciated that throughout the description, discussions utilizing terms such as "processing", "computing", "calculating", "determining" and the like, refer to the action and processes of a data processing system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the data processing system's registers and memories into other data similarly represented as physical quantities within the data processing system memories or registers or other such information storage, transmission or display devices.

[0079] The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various general-purpose systems may be used with programs in accordance with the teachings herein, or it may prove convenient to construct more specialized apparatus to perform the required method operations. The required structure for a variety of these systems will appear from the description below. In addition, embodiments of the present invention are not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of embodiments of the invention as described herein.

[0080] In the foregoing specification, embodiments of the invention have been described with reference to specific exemplary embodiments thereof. It will be evident that various modifications may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative sense rather than a restrictive sense.

What is claimed is:

1. A method, comprising:
- analyzing a corpus having training sequences of words in a vector space;
- extracting a global semantic information associated with an input sequence of words based on the analyzing;
- identifying the training sequences of words in the vector space associated with the global semantic information; and
- assigning a part-of-speech tag to a word of the input sequence based on the identified training sequences.

2. The method of claim **1**, wherein the vector space includes a latent semantic space.

3. The method of claim **1**, wherein the analyzing comprises
- mapping the input sequence into the vector space; and
- forming a neighborhood associated with the input sequence in the vector space, wherein the neighborhood represents one or more training sequences that are globally relevant to the input sequence.

4. The method of claim **1**, wherein the analyzing comprises determining a closeness measure between the training sequences and the input sequence in the vector space.

5. The method of claim **1**, wherein the global semantic information is used to identify the training sequences that are globally relevant to the input sequence.

6. A method to assign part-of-speech tags to words, comprising:
- receiving an input sequence of words;
- mapping the input sequence into a vector space, wherein the vector space includes representations of a plurality of training sequences of words; and
- forming a neighborhood associated with the input sequence in the vector space to obtain part-of-speech characteristics, wherein the neighborhood represents one or more training sequences that are globally relevant to the input sequence.

7. The method of claim **6**, further comprising
- assigning a part-of-speech tag to a word of the input sequence based on the part-of-speech characteristics.

8. The method of claim **6**, wherein the vector space includes a latent semantic space.

9. The method of claim **6**, wherein the forming the neighborhood comprises
- determining a closeness measure between representations of a training sequence of the plurality of the training sequences and the input sequence in the vector space; and

selecting the training sequence out of the plurality of the training sequences based on the closeness measure.

10. The method of claim **9**, further comprising

determining whether the closeness measure exceeds a predetermined threshold, and

selecting the training sequence if the closeness measure exceeds the predetermined threshold.

11. The method of claim **9**, further comprising

ranking the training sequences according to the closeness measure; and

selecting the training sequence that has rank higher than a predetermined rank.

12. The method of claim **6**, further comprising

determining whether a training sequence in the neighborhood contains a first word that is similar to an input word of the input sequence;

forming one or more sub-sequences of the training sequence that contain one or more first words that are similar to the input words;

aligning the one or more sub-sequences to obtain one or more part-of-speech characteristics of the first words; and

determining a part-of-speech tag for the input word based on the one or more part-of-speech characteristics of the first word.

13. An article of manufacture comprising:

a machine-accessible medium including data that, when accessed by a machine, cause the machine to perform operations comprising,

analyzing a corpus having training sequences of words in a vector space;

extracting a global semantic information associated with an input sequence of words based on the analyzing;

identifying the training sequences of words in the vector space associated with the global semantic information; and

assigning a part-of-speech tag to a word of the input sequence based on the identified training sequences.

14. The article of manufacture of claim **13**, wherein the vector space includes a latent semantic space.

15. The article of manufacture of claim **13**, wherein the analyzing comprises

mapping the input sequence into the vector space; and

forming a neighborhood associated with the input sequence in the vector space, wherein the neighborhood represents one or more training sequences that are globally relevant to the input sequence.

16. The article of manufacture of claim **13**, wherein the analyzing comprises determining a closeness measure between the training sequences and the input sequence in the vector space.

17. The article of manufacture of claim **13**, wherein the global semantic information is used to identify the training sequences that are globally relevant to the input sequence.

18. An article of manufacture comprising:

a machine-accessible medium including data that, when accessed by a machine, cause the machine to perform operations to assign part-of-speech tags to words, comprising:

receiving an input sequence of words;

mapping the input sequence into a vector space, wherein the vector space includes representations of a plurality of training sequences of words; and

forming a neighborhood associated with the input sequence in the vector space to obtain part-of-speech characteristics, wherein the neighborhood represents one or more training sequences that are globally relevant to the input sequence.

19. The article of manufacture of claim **18**, wherein the machine-accessible medium further includes data that causes the machine to perform operations comprising,

assigning a part-of-speech tag to a word of the input sequence based on the part-of-speech characteristics.

20. The article of manufacture of claim **18**, wherein the vector space includes a latent semantic space.

21. The article of manufacture of claim **18**, wherein the forming the neighborhood comprises

determining a closeness measure between representations of a training sequence of the plurality of the training sequences and the input sequence in the vector space; and

selecting the training sequence out of the plurality of the training sequences based on the closeness measure.

22. The article of manufacture of claim **21**, wherein the machine-accessible medium further includes data that causes the machine to perform operations comprising,

determining whether the closeness measure exceeds a predetermined threshold, and

selecting the training sequence if the closeness measure exceeds the predetermined threshold.

23. The article of manufacture of claim **21**, wherein the machine-accessible medium further includes data that causes the machine to perform operations comprising,

ranking the training sequences according to the closeness measure; and

selecting the training sequence that has rank higher than a predetermined rank.

24. The article of manufacture of claim **18**, wherein the machine-accessible medium further includes data that causes the machine to perform operations comprising,

determining whether a training sequence in the neighborhood contains a first word that is similar to an input word of the input sequence;

forming one or more sub-sequences of the training sequence that contain one or more first words that are similar to the input words;

aligning the one or more sub-sequences to obtain one or more part-of-speech characteristics of the first words; and

determining a part-of-speech tag for the input word based on the one or more part-of-speech characteristics of the first word.

25. A data processing system, comprising:

means for analyzing a corpus having training sequences of words in a vector space;

means for extracting a global semantic information associated with an input sequence of words based on the analyzing;

means for identifying the training sequences of words in the vector space associated with the global semantic information; and

means for assigning a part-of-speech tag to a word of the input sequence based on the identified training sequences.

* * * * *