



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2020년07월03일

(11) 등록번호 10-2129832

(24) 등록일자 2020년06월29일

(51) 국제특허분류(Int. Cl.)  
**G06F 40/20** (2020.01) **G06F 3/023** (2006.01)  
**G06F 40/12** (2020.01)

(21) 출원번호 10-2014-7035005

(22) 출원일자(국제) 2013년05월14일

심사청구일자 2018년05월14일

(85) 번역문제출일자 2014년12월12일

(65) 공개번호 10-2015-0013294

(43) 공개일자 2015년02월04일

(86) 국제출원번호 PCT/GB2013/051244

(87) 국제공개번호 WO 2013/171481

국제공개일자 2013년11월21일

(30) 우선권주장

1208373.9 2012년05월14일 영국(GB)

(56) 선행기술조사문헌

KR1020140119763 A

(뒷면에 계속)

전체 청구항 수 : 총 26 항

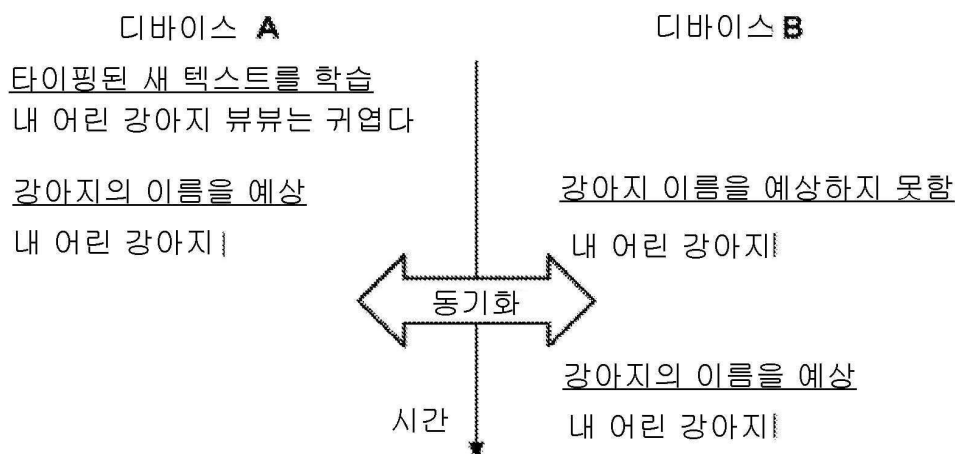
심사관 : 경연정

(54) 발명의 명칭 디바이스를 동기화하는 메카니즘, 시스템 및 방법

## (57) 요약

한 명의 사용자와 관련된, 다이나믹 언어모델을 각각 포함한 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 메카니즘이 제공된다. 상기 메카니즘은 하나 이상의 복수의 디바이스들에 사용자가 입력한 텍스트를 나타내는 텍스트 데이터를 수신하고; 텍스트 데이터에 대해 적어도 하나의 언어모델을 훈련시키며; 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하도록 구성된다. 상기 메카니즘과 복수의 디바이스들을 포함한 시스템 및 한 명의 사용자와 관련된 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 방법이 또한 제공된다.

## 대표도 - 도2



(72) 발명자

**헤이트건 엠마누엘 조지**

영국 캄프릿지 씨비1 9엑스유 켈시 크레슨트 49

**메드록 벤저민**

영국 런던 에스이3 9엘엘 블랙히스 론 테러스 21

(56) 선행기술조사문헌

KR1020150074075 A

JP2012507809 A\*

KR1020050082249 A\*

US7035866 A

\*는 심사관에 의하여 인용된 문헌

---

## 명세서

### 청구범위

#### 청구항 1

시스템에 있어서,

프로세서; 및

명령어들을 저장하는 메모리

를 포함하며,

상기 명령어들은, 상기 프로세서에 의해 실행될 때:

사용자에 의해 상기 사용자와 연관된 복수의 디바이스들 중 하나 이상의 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 수신하고;

상기 텍스트 데이터의 시퀀스들에 대한 발생 빈도를 누적(accumulating)함으로써, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키며 - 상기 빈도는 상기 하나 이상의 언어 모델 중 각각의 언어 모델에서의 발생에 대한 것임 - ;

상기 하나 이상의 언어 모델을 상기 복수의 디바이스들 중 하나 이상의 디바이스에 제공하도록 - 상기 하나 이상의 언어 모델은 상기 디바이스들 각각에서 상기 하나 이상의 언어 모델 중 각각의 언어 모델에서의 발생 빈도를 병합(merge)시키기 위하여 결합됨 -

상기 프로세서를 구성하며,

상기 복수의 디바이스들 사이에서의 상기 하나 이상의 언어 모델의 전개(deployment)에 응답하여, 상기 하나 이상의 언어 모델은 상기 병합된 발생 빈도에 기반하여 상기 복수의 디바이스들 간에 텍스트 입력들에 대한 동기화된 텍스트 예측을 가능하게 하고,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 델타 언어 모델과 연관된 디바이스를 제외한 상기 복수의 디바이스들로부터 수신된 텍스트 데이터를 사용하여, 상기 복수의 디바이스들 각각에 대해 상기 델타 언어 모델을 발생시키는 것을 포함하는 것인, 시스템.

#### 청구항 2

제1항에 있어서,

상기 텍스트 데이터는 텍스트를 포함하고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 상기 텍스트에 대해 적어도 하나의 언어 모델을 훈련시키는(training) 것을 포함하는 것인, 시스템.

#### 청구항 3

제1항에 있어서,

상기 텍스트 데이터는 상기 사용자에 의해 입력된 텍스트에 대해 훈련된 디바이스 델타 언어 모델을 포함하고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 상기 디바이스 델타 언어 모델을 상기 하나 이상의 언어 모델에 병합하는 것을 포함하는 것인, 시스템.

#### 청구항 4

제1항에 있어서,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 상기 텍스트 데이터로부터 적어도 하나의 언어 모델을 발생시키는 것, 또는 상기 텍스트 데이터를 적어도 하나의 기존 언어 모델에 통합시키는 것을 포함하는 것인, 시스템.

#### 청구항 5

제1항에 있어서,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 누적(cumulative) 언어 모델을 발생시키기 위해 상기 텍스트 데이터를 단일 언어 모델에 통합시키는 것을 포함하는 것인, 시스템.

#### 청구항 6

제5항에 있어서,

상기 텍스트 데이터는 상기 디바이스들의 이전 동기화 이후에 상기 복수의 디바이스들 중 임의의 디바이스에 입력된 텍스트를 나타내고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 것은, 상기 텍스트 데이터를 상기 누적 언어 모델에 통합시키는 것을 포함하는 것인, 시스템.

#### 청구항 7

삭제

#### 청구항 8

제1항에 있어서,

상기 시스템은, 주어진 시간에 상기 복수의 디바이스들 중 단일 디바이스와 동기화하도록 구성되는 것인, 시스템.

#### 청구항 9

제1항에 있어서,

상기 시스템은 복수의 디바이스들을 더 포함하며, 각각의 디바이스는 다이나믹 언어 모델을 포함하고,

상기 복수의 디바이스들 각각은 해당 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 상기 프로세서에 송신하도록 구성되는 것인, 시스템.

#### 청구항 10

제5항에 있어서,

상기 시스템은 복수의 디바이스들을 더 포함하며, 각각의 디바이스는 다이나믹 언어 모델을 포함하고,

상기 복수의 디바이스들 각각은 해당 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 상기 프로세서에 송신하도록 구성되고,

각각의 디바이스는, 상기 누적 언어 모델을 수신하고, 해당 디바이스에 입력된 상기 텍스트 데이터를 상기 프로세서에 송신하기 이전에, 상기 누적 언어 모델을 자신의 다이나믹 언어 모델에 병합시키도록 구성되는 것인, 시스템.

#### 청구항 11

제1항에 있어서,

상기 시스템은 복수의 디바이스들을 더 포함하며, 각각의 디바이스는 다이나믹 언어 모델을 포함하고,

상기 시스템은, 주어진 시간에 상기 복수의 디바이스들 중 단일 디바이스와 동기화하도록 구성되고,

상기 복수의 디바이스들 각각은 해당 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 상기 프로세서에 송신하도록 구성되고,

각각의 디바이스는 해당 디바이스와 연관된 상기 델타 언어 모델을 수신하고, 상기 델타 언어 모델을 자신의 다이나믹 언어 모델에 병합시키도록 구성되는 것인, 시스템.

#### 청구항 12

제9항에 있어서,

각각의 디바이스의 각각의 다이나믹 언어 모델은 상기 디바이스에 입력된 텍스트에 기반하여 적어도 하나의 텍스트 예측을 발생시키도록 구성되고, 동기화되면, 상기 복수의 디바이스들의 다이나믹 언어 모델들은 동일한 텍스트 입력이 제공될 때, 동일한 상기 적어도 하나의 텍스트 예측을 발생시킬 수 있는 것인, 시스템.

#### 청구항 13

제9항에 있어서,

상기 시스템은 서버를 포함하고, 상기 복수의 디바이스들 각각은 상기 서버로부터 상기 하나 이상의 언어 모델을 다운로드하고, 상기 서버에 상기 텍스트 데이터를 업로드하도록 구성되는 것인, 시스템.

#### 청구항 14

제1항에 있어서,

상기 하나 이상의 언어 모델은, 용어들의 시퀀스들을 각각의 시퀀스에 대한 발생 빈도와 연관시키는 데이터 구조를 포함하는 것인, 시스템.

#### 청구항 15

제10항에 있어서,

상기 하나 이상의 언어 모델은, 용어들의 시퀀스들을 각각의 시퀀스에 대한 발생 빈도와 연관시키는 데이터 구조를 포함하고,

상기 디바이스는:

제2 언어 모델의 데이터 구조 내의 시퀀스들에 대한 발생 빈도를 제1 언어 모델의 데이터 구조 내의 대응 시퀀스들에 대한 발생 빈도에 추가하는 것; 및

새로운 시퀀스 및 그것의 대응 발생 빈도를, 해당 시퀀스가 상기 제1 언어 모델의 데이터 구조 내에 있지 않고 상기 제2 언어 모델의 데이터 구조 내에 있는 경우, 상기 제1 언어 모델의 데이터 구조에 삽입하는 것에 의해, 상기 제1 언어 모델을 상기 제2 언어 모델과 병합하도록 구성되는 것인, 시스템.

#### 청구항 16

단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법에 있어서,

각각의 디바이스는 다이나믹 언어 모델을 포함하고, 상기 방법은:

동기화시키기 위한 시스템에서, 사용자에 의해 상기 복수의 디바이스들 중 하나 이상의 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 수신하는 단계;

상기 시스템으로, 상기 텍스트 데이터의 시퀀스들에 대한 발생 빈도를 누적함으로써, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계 - 상기 빈도는 상기 하나 이상의 언어 모델 각각으로부터의 발생에 대한 것임 - ; 및

상기 시스템에 의해, 상기 하나 이상의 언어 모델을 제공하는 단계 - 상기 하나 이상의 언어 모델은 상기 디바이스들 각각에서 상기 하나 이상의 언어 모델 중 각각의 언어 모델에서의 발생 빈도를 병합시키기 위해 결합됨 -

를 포함하며,

상기 복수의 디바이스들 사이에서의 상기 하나 이상의 언어 모델의 전개에 응답하여, 상기 하나 이상의 언어 모델은 상기 병합된 발생 빈도에 기반하여 상기 복수의 디바이스들 간에 텍스트 입력들에 대한 동기화된 텍스트 예측을 가능하게 하고,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 델타 언어 모델과 연관된 디바이스를 제외한 상기 복수의 디바이스들로부터 상기 시스템에 의해 수신된 텍스트 데이터를 사용하여, 상기 복수의 디바이스

들 각각에 대해 상기 델타 언어 모델을 발생시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 17

제16항에 있어서,

상기 텍스트 데이터는 텍스트를 포함하고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 상기 텍스트에 대해 적어도 하나의 언어 모델을 훈련시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 18

제16항에 있어서,

상기 텍스트 데이터는 상기 사용자에 의해 입력된 텍스트에 대해 훈련된 디바이스 델타 언어 모델을 포함하고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 상기 디바이스 델타 언어 모델을 상기 하나 이상의 언어 모델에 병합시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 19

제16항에 있어서,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 상기 텍스트 데이터로부터 하나 이상의 언어 모델을 발생시키는 단계, 또는 상기 텍스트 데이터를 적어도 하나의 기존 언어 모델에 통합시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 20

제16항에 있어서,

상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 누적 언어 모델을 발생시키기 위하여 상기 텍스트 데이터를 단일 언어 모델에 통합시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 21

제20항에 있어서,

상기 텍스트 데이터는, 상기 복수의 디바이스들의 이전 동기화 이후에 상기 복수의 디바이스들에 입력된 텍스트를 나타내고, 상기 텍스트 데이터를 하나 이상의 언어 모델에 통합시키는 단계는, 상기 텍스트 데이터를 상기 누적 언어 모델에 통합시키는 단계를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 22

제16항에 있어서,

상기 복수의 디바이스들 각각은 개별적으로 상기 시스템과 동기화되는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 23

삭제

#### 청구항 24

제16항에 있어서,

상기 복수의 디바이스들 각각은 개별적으로 상기 시스템과 동기화되고, 델타 언어 모델들에 통합된 상기 텍스트

데이터는, 상기 델타 언어 모델과 연관된 디바이스를 제외한 상기 복수의 디바이스들의 동기화 및 후속 동기화 중 적어도 하나 동안에 상기 시스템에 의해 수신된 텍스트 데이터를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 25

제20항에 있어서,

각각의 디바이스가 상기 누적 언어 모델을 수신하고, 해당 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 상기 시스템에 송신하기 이전에, 상기 누적 언어 모델을 해당 디바이스와 연관된 다이나믹 언어 모델에 병합시키는 단계

를 더 포함하는, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 26

제16항에 있어서,

각각의 디바이스가 해당 디바이스와 연관된 델타 언어 모델을 수신하고, 상기 델타 언어 모델을 해당 디바이스의 다이나믹 언어 모델에 병합시키는 단계

를 더 포함하는, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 27

제18항에 있어서,

상기 하나 이상의 언어 모델은, 용어의 시퀀스들을 각각의 시퀀스에 대한 발생 빈도와 연관시키는 데이터 구조를 포함하고,

제1 언어 모델을 제2 언어 모델과 병합시키는 것은:

상기 제2 언어 모델의 데이터 구조 내의 시퀀스들에 대한 발생 빈도를 상기 제1 언어 모델의 데이터 구조 내의 대응 시퀀스들에 대한 발생 빈도에 추가하는 단계; 및

새로운 시퀀스 및 그것의 대응 발생 빈도를, 해당 시퀀스가 상기 제1 언어 모델의 데이터 구조 내에 있지 않고 상기 제2 언어 모델의 데이터 구조 내에 있는 경우, 상기 제1 언어 모델의 데이터 구조에 삽입하는 단계

를 포함하는 것인, 단일 사용자와 연관된 복수의 디바이스들에 상주하는 복수의 다이나믹 언어 모델들을 동기화시키기 위한 방법.

#### 청구항 28

프로세서에 의해 실행될 때, 제16항의 방법을 실행하도록 상기 프로세서를 구성하는 프로그램 명령어들을 포함한 비-일시적 컴퓨터 판독가능 저장 매체.

#### 청구항 29

삭제

#### 청구항 30

삭제

#### 청구항 31

삭제

#### 청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48



삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

청구항 58

삭제

청구항 59

삭제

청구항 60

삭제

청구항 61

삭제

청구항 62

삭제

청구항 63

삭제

청구항 64

삭제

청구항 65

삭제

청구항 66

삭제

청구항 67

삭제

청구항 68

삭제

청구항 69

삭제

청구항 70

삭제

청구항 71

삭제

청구항 72

삭제

## 발명의 설명

### 기술 분야

[0001] 본 발명은 디바이스를 동기화하기 위한 메카니즘과 그렇게 하는 방법에 관한 것이다.

### 배경 기술

[0002] 많은 사용자들은 복수의 디바이스들에 문자를 입력한다. 예컨대, 사용자는 휴대폰에 문자메시지(SMS/MMS) 또는 이메일 타이핑하는 것 이외에 태블릿 또는 PC에 이메일이나 문서들을 작성할 수 있다. 각각의 디바이스는 사용자들의 문자 입력을 돕기 위한 텍스트 입력 시스템을 포함한다.

[0003] 텍스트 입력 시스템은 소정의 언어 스타일, 가령, 사용자의 쓰기 스타일의 확률적 인캡슐레이션(probabilistic encapsulation)일 수 있는 언어모델을 포함할 수 있다. 텍스트 입력 시스템을 기반으로 한 언어 모델은 기능 범위, 가령, 언어를 기초로 한 오타/오자 입력 정정 및/또는 차례로 가능한 장래 용어들의 예상을 통해 전자 디바이스에 대한 타이핑 경험을 강화할 수 있다.

[0004] 언어모델은 사용자가 텍스트를 디바이스에 입력할 때 사용자 입력에 대해 점차 훈련되며, 이에 따라 텍스트 입력 시스템이 오타/오자 입력을 정정하거나 사용자가 이전에 입력한 텍스트를 기반으로 차례대로 가능성 있는 장래의 용어들을 예상하게 할 수 있는 다이나믹 언어모델일 수 있다.

[0005] 본 출원의 발명자는 복수의 디바이스들을 소유한 사용자가 체험한 문제를 알았고, 각각의 디바이스는 시간에 걸쳐 사용자의 언어 스타일을 학습한다: 디바이스가 발생한 예상은 사용에 따라 벗어날 수 있다. 예컨대, 사용자는 다른 디바이스보다 한 디바이스를 훨씬 더 많이 사용할 수 있다. 자주 사용되는 디바이스는 자주 사용되지 않는 디바이스보다 사용자에게 대해 더 정확한 예상을 발생할 것이며, 이는 사용자에게 성가시고 문제가 될 것이다.

### 발명의 내용

## 해결하려는 과제

[0006] 본 발명의 목적은 이런 문제를 극복하는 것이다.

## 과제의 해결 수단

[0007] 본 발명의 일태양으로, 한 명의 사용자와 관련된, 다이나믹 언어모델을 각각 포함한 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 메카니즘이 제공된다. 상기 메카니즘은 하나 이상의 복수의 디바이스들에 사용자가 입력한 텍스트를 수신하고; 텍스트에 대해 적어도 하나의 언어모델을 훈련시키며; 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하도록 구성된다.

[0008] 텍스트 데이터는 사용자가 입력한 텍스트를 나타내는 임의의 데이터일 수 있다. 제 1 실시예에서, 텍스트 데이터는 사용자가 입력한 실제 텍스트를 포함할 수 있다. 이 경우, 텍스트 데이터를 언어모델에 포함시키는 것은 사용자가 입력한 텍스트에 대한 언어모델을 훈련시키는 것을 포함할 수 있다. 제 2 실시예에서, 텍스트 데이터는 상기 입력된 텍스트에 대해 훈련된 디바이스 델타언어모델일 수 있다. 이 경우, 텍스트 데이터를 언어모델에 포함시키는 것은 디바이스 델타언어모델을 적어도 하나의 언어모델에 병합시키는 것을 포함할 수 있다.

[0009] 따라서, 본 발명의 메카니즘의 제 1 실시예에서, 한 명의 사용자와 관련된, 다이나믹 언어모델을 각각 포함한 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 메카니즘이 제공된다. 메카니즘은 하나 이상의 복수의 디바이스들에 사용자가 입력한 텍스트를 수신하고; 텍스트에 대해 적어도 하나의 언어모델을 훈련시키며; 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하도록 구성된다.

[0010] 바람직하기로, 메카니즘은 텍스트로부터 적어도 하나의 언어모델을 발생하거나 상기 텍스트로 적어도 하나의 언어모델을 훈련시킴으로써 텍스트에 대한 적어도 하나의 언어모델을 훈련시키도록 구성된다.

[0011] 텍스트는 복수의 디바이스들에 입력된 임의의 텍스트를 포함할 수 있고, 적어도 하나의 언어모델을 훈련시키는 것은 바람직하게는 누적 언어모델을 발생시키기 위해 텍스트에 대한 하나의 언어모델을 훈련시키는 것을 포함한다. 텍스트는 복수의 다이나믹 언어모델의 이전 동기화 이후 복수의 디바이스들 중 어느 하나에 입력된 텍스트를 포함하고 적어도 하나의 언어모델을 훈련시키는 것은 텍스트에 대한 누적 언어모델을 더 훈련시키는 것을 포함한다.

[0012] 적어도 하나의 언어모델을 훈련시키는 것은 특정 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들로부터 수신된 텍스트를 이용해 복수의 디바이스들 각각에 대한 델타언어모델을 발생하는 것을 포함할 수 있다.

[0013] 메카니즘은 바람직하게는 주어진 시간에 복수의 디바이스들 중 하나의 디바이스를 동기화하도록 구성된다.

[0014] 메카니즘은 사용자를 인증하도록 구성될 수 있다.

[0015] 본 발명의 메카니즘의 제 2 실시예로, 한 명의 사용자와 관련되고, 다이나믹 언어모델 및 사용자가 디바이스에 입력한 텍스트에 대해 훈련된 디바이스 델타언어모델을 각각 포함한 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 메카니즘이 제공된다. 상기 메카니즘은 디바이스 델타언어모델을 수신하고, 상기 디바이스 델타언어모델을 적어도 하나의 언어모델에 병합시키며, 상기 디바이스들을 동기화하기 위해 적어도 하나의 언어모델을 제공하도록 구성된다.

[0016] 바람직하기로, 상기 메카니즘은 디바이스 델타언어모델로부터 적어도 하나의 언어모델을 발생하거나 디바이스 델타언어모델을 적어도 하나의 기존 언어모델과 병합함으로써 디바이스 델타언어모델을 적어도 하나의 언어모델에 병합하도록 구성된다.

[0017] 디바이스 델타언어모델을 적어도 하나의 기존 언어모델과 병합시키는 것은 누적언어모델을 발생시키기 위해 상기 디바이스 델타언어모델을 하나의 언어모델에 병합시키는 것을 포함한다. 디바이스 델타언어모델은 복수의 다이나믹 언어모델의 이전 동기화 이후로 복수의 디바이스들 중 하나에 입력된 텍스트에 대해 훈련될 수 있다. 디바이스 델타언어모델을 적어도 하나의 기존 언어모델과 병합시키는 것은 특정 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들로부터 수신된 디바이스 델타언어모델을 이용해 복수의 디바이스들 각각에 대한 델타언어모델을 발생하는 것을 포함할 수 있다.

[0018] 메카니즘은 바람직하게는 주어진 시간에 복수의 디바이스들 중 한 디바이스와 동기화하도록 구성된다.

- [0019] 메카니즘은 사용자를 인증하도록 구성될 수 있다.
- [0020] 본 발명의 제 2 태양으로, 텍스트 입력 시스템이 제공된다. 시스템은 다이나믹 언어모델을 각각 포함한 복수의 디바이스들; 및 상기 실시예들 중 어느 하나에 따른 메카니즘을 포함한다. 복수의 디바이스들 각각은 상기 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 메카니즘으로 전송하도록 구성된다.
- [0021] 본 발명의 시스템의 제 1 실시예에서, 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터는 디바이스에 입력된 실제 텍스트이다.
- [0022] 시스템의 각 디바이스는 누적언어모델을 수신하고, 상기 디바이스에 입력된 텍스트를 메카니즘으로 전송하기 전에 상기 누적언어모델을 다이나믹 언어모델에 병합시키도록 구성될 수 있다.
- [0023] 각 디바이스는 상기 디바이스와 관련된 특정 델타언어모델을 수신하고 상기 델타언어모델을 다이나믹 언어모델에 병합하도록 구성될 수 있다.
- [0024] 각 디바이스의 각 다이나믹 언어모델은 디바이스에 입력된 텍스트를 기반으로 적어도 하나의 텍스트 예측을 발생하도록 구성되고, 일단 동기화된 후, 복수의 다이나믹 언어모델은 동일한 텍스트 입력이 제공될 경우 동일한 적어도 하나의 텍스트 예측을 발생할 수 있다.
- [0025] 본 시스템의 일실시예에서, 메카니즘은 서버를 포함하고, 복수의 디바이스들 각각은 서버로부터 누적언어모델 또는 델타언어모델을 다운로드하고 서버에 텍스트를 업로드하도록 구성된다. 시스템은 서버와 복수의 디바이스들 각각 사이에 보안연결수단을 포함할 수 있다.
- [0026] 상술한 메카니즘 또는 시스템에 대해, 언어모델은 용어의 시퀀스를 각 시퀀스에 대한 발생 빈도와 연관시키는 데이터 구조를 포함할 수 있다. 디바이스는 제 2 언어모델의 데이터 구조에서 시퀀스에 대한 발생 빈도를 제 1 언어모델의 데이터 구조에서 해당 시퀀스에 대한 발생빈도에 추가하고, 시퀀스가 제 2 언어모델의 데이터 구조에 있으나 제 1 언어모델의 데이터 구조에 없다면, 새 시퀀스와 해당 발생빈도를 제 1 언어모델의 데이터 구조에 삽입함으로써, 제 1 언어모델을 제 2 언어모델과 병합하도록 구성될 수 있다. 하나 이상의 시퀀스들은 발생 빈도가 문턱값 이하로 떨어지면, 각 디바이스는 병합된 데이터 구조로부터 하나 이상의 시퀀스를 제거하도록 구성될 수 있다.
- [0027] 본 발명의 시스템의 제 2 실시예에서, 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터는 상기 디바이스에 입력된 텍스트에 대해 훈련된 델타언어모델이다. 이 실시예에서, 시스템은 다이나믹 언어모델과 상기 디바이스에 사용자가 입력한 텍스트에 대해 훈련된 디바이스 델타언어모델을 각각 포함하는 복수의 디바이스들과 상기 실시예들 중 어느 하나에 따른 메카니즘을 포함한다. 복수의 디바이스들 각각은 디바이스 델타언어모델을 메카니즘으로 전송하도록 구성된다.
- [0028] 시스템의 각 디바이스는 누적언어모델을 수신하고 델타언어모델을 메카니즘으로 전송하기 전에 상기 누적언어모델을 다이나믹 언어모델에 병합시키도록 구성될 수 있다.
- [0029] 각 디바이스는 디바이스와 관련된 특정 델타언어모델을 수신하고 상기 델타언어모델을 다이나믹 언어모델에 병합시키도록 구성될 수 있다.
- [0030] 각 디바이스의 각 다이나믹 언어모델은 디바이스에 입력된 텍스트를 기반으로 적어도 하나의 텍스트 예측을 발생하도록 구성될 수 있고, 일단 동기화된 후, 복수의 다이나믹 언어모델들이 동일한 텍스트 입력이 제공될 경우 동일한 적어도 하나의 텍스트 예측을 발생할 수 있다.
- [0031] 본 시스템의 일실시예에서, 메카니즘은 서버를 포함하고 복수의 디바이스들 각각은 서버로부터 누적언어모델 또는 델타언어모델을 다운로드하고 서버에 디바이스 델타언어모델을 업로드하도록 구성된다. 시스템은 서버와 복수의 디바이스들 각각 간에 보안연결수단을 포함할 수 있다.
- [0032] 상술한 메카니즘 또는 시스템에 대해, 언어모델은 용어의 시퀀스를 각 시퀀스에 대한 발생빈도와 연관시키는 데이터 구조를 포함할 수 있다. 메카니즘 및/또는 디바이스는 제 2 언어모델의 데이터 구조에서 시퀀스에 대한 발생빈도를 제 1 언어모델의 데이터 구조에서 해당 시퀀스에 대한 발생빈도에 추가하고; 시퀀스가 제 2 언어모델의 데이터 구조에 있으나 제 1 언어모델의 데이터 구조에 없다면, 새 시퀀스와 해당 발생빈도를 제 1 언어모델의 데이터 구조에 삽입함으로써 제 1 언어모델을 제 2 언어모델과 병합하도록 구성될 수 있다. 하나 이상의 시퀀스들은 발생 빈도가 기준치 아래로 떨어지면, 하나 이상의 시퀀스들을 병합된 데이터 구조에서 제거하도록 메카니즘 및/또는 디바이스가 구성될 수 있다.

- [0033] 본 발명의 제 3 태양으로, 한 명의 사용자와 관련된, 다이나믹 언어모델을 각각 포함한 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 방법이 제공된다. 상기 방법은 동기화 메카니즘에, 하나 이상의 복수의 디바이스들에 사용자가 입력한 텍스트를 나타내는 텍스트 데이터를 수신하는 단계; 메카니즘으로, 텍스트에 대해 적어도 하나의 언어모델을 훈련시키는 단계; 및 메카니즘으로, 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하는 단계를 포함한다.
- [0034] 본 발명의 방법의 제 1 실시예로, 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터는 디바이스에 입력된 실제 텍스트이다.
- [0035] 텍스트에 대한 적어도 하나의 언어모델을 훈련시키는 단계는 텍스트로부터 적어도 하나의 언어모델을 발생하거나 텍스트로 적어도 하나의 기존 언어모델을 훈련시키는 단계를 포함한다.
- [0036] 텍스트는 복수의 디바이스들에 입력된 임의의 텍스트를 포함하고, 적어도 하나의 언어모델을 훈련시키는 단계는 누적언어모델을 발생하기 위해 텍스트에 대한 적어도 하나의 언어모델을 훈련시키는 단계를 포함할 수 있다. 텍스트는 복수의 디바이스들의 이전의 동기화 이후 상기 복수의 디바이스들에 입력된 텍스트를 포함할 수 있고, 적어도 하나의 언어모델을 훈련시키는 단계는 상기 텍스트에 대한 누적언어모델을 더 훈련시키는 단계를 포함한다.
- [0037] 바람직하기로, 복수의 디바이스들 각각은 메카니즘으로 개별적으로 동기화된다.
- [0038] 누적언어모델은 메카니즘으로 초기에 동기화된 임의의 디바이스로부터 메카니즘에 의해 수신된 텍스트에 대해 훈련될 수 있다. 누적언어모델은 메카니즘과 연이어 동기화된 임의의 디바이스로부터 메카니즘에 의해 수신된 텍스트에 대해 더 훈련될 수 있다.
- [0039] 적어도 하나의 언어모델을 훈련시키는 단계는 특정 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들로부터 메카니즘에 의해 수신된 텍스트를 이용해 복수의 디바이스들 각각에 대한 델타언어모델을 발생하는 단계를 포함할 수 있다.
- [0040] 델타언어모델을 훈련시키는데 이용된 텍스트는 특정한 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들의 동기화 및/또는 연속 동기화 동안 메카니즘에 의해 수신된 텍스트를 포함할 수 있다.
- [0041] 상기 방법은 사용자를 인증하는 메카니즘을 더 포함할 수 있다.
- [0042] 바람직하기로, 상기 방법은 누적언어모델을 수신하고 디바이스에 입력된 텍스트를 메카니즘으로 전송하기 전에 상기 디바이스와 관련된 다이나믹 언어모델에 누적언어모델을 병합시키는 각 디바이스를 더 포함한다.
- [0043] 바람직하기로, 상기 방법은 상기 디바이스와 관련된 특정 델타언어모델을 수신하고, 상기 디바이스의 다이나믹 언어모델에 델타언어모델을 병합시키는 각 디바이스를 포함한다.
- [0044] 메카니즘은 바람직하게는 서버를 포함한다. 상기 방법은 서버와 복수의 디바이스들 각각 간에 보안연결을 확립하는 단계를 더 포함한다.
- [0045] 언어모델은 용어의 발생 빈도와 각 시퀀스에 대한 발생 빈도를 연관시키는 데이터 구조를 포함할 수 있다. 상기 방법에서, 제 1 언어모델을 제 2 언어모델과 병합시키는 단계는: 제 2 언어모델의 데이터 구조에서 시퀀스에 대한 발생빈도를 제 1 언어모델의 데이터 구조에서 해당 시퀀스에 대한 발생빈도에 추가하는 단계; 및 시퀀스가 제 2 언어모델의 데이터 구조에 있으나 제 1 언어모델의 데이터 구조에 없다면, 새 시퀀스와 해당 발생빈도를 제 1 언어모델의 데이터 구조에 삽입하는 단계를 포함한다. 상기 방법은 하나 이상의 시퀀스들이 발생 빈도가 문턱값 이하로 떨어지면, 병합된 데이터구조로부터 하나 이상의 시퀀스들을 제거하는 복수의 디바이스들 각각을 더 포함할 수 있다.
- [0046] 본 발명의 제 2 실시예에서, 텍스트 데이터는 입력된 텍스트에 대해 훈련된 디바이스 델타언어모델이다. 이 경우, 각 디바이스는 다이나믹 언어모델과 상기 디바이스에 사용자가 입력한 텍스트에 대해 훈련된 디바이스 델타언어모델을 포함한다. 상기 방법은, 동기화 메카니즘에, 디바이스 델타언어모델을 수신하는 단계; 메카니즘으로, 디바이스 델타언어모델을 적어도 하나의 언어모델과 병합시키는 단계; 및 메카니즘으로 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하는 단계를 포함한다.
- [0047] 디바이스 델타언어모델을 적어도 하나의 언어모델에 병합시키는 단계는 디바이스 델타언어모델로부터 적어도 하나의 언어모델을 발생하거나 디바이스 델타언어모델을 적어도 하나의 기존 언어모델과 병합시키는 단계를 포함

한다.

- [0048] 디바이스 델타언어모델을 적어도 하나의 언어모델에 병합시키는 단계는 누적언어모델을 발생하기 위해 디바이스 델타언어모델을 하나의 언어모델에 병합시키는 단계를 포함할 수 있다. 디바이스 델타언어모델은 복수의 다이나믹 언어모델의 이전 동기화 이후 복수의 디바이스들 중 하나에 입력된 텍스트에 대해 훈련될 수 있다.
- [0049] 바람직하기로, 복수의 디바이스들 각각은 메카니즘으로 개별적으로 동기화된다.
- [0050] 메카니즘으로 초기에 동기화된 임의의 디바이스로부터 수신된 델타언어모델은 누적언어모델과 병합될 수 있다. 메카니즘과 연속으로 동기화된 임의의 디바이스로부터 수신된 임의의 디바이스 델타언어모델은 누적언어모델과 병합될 수 있다.
- [0051] 디바이스 델타언어모델을 적어도 하나의 언어모델과 병합하는 단계는 특정 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들로부터 메카니즘에 의해 수신된 디바이스 델타언어모델을 이용해 복수의 디바이스들 각각에 대해 델타언어모델을 발생하는 단계를 포함할 수 있다.
- [0052] 델타언어모델과 병합된 디바이스 델타언어모델은 특정 델타언어모델과 관련된 디바이스를 제외하고 복수의 디바이스들의 동기화 및/또는 연속 동기화 동안 메카니즘에 의해 수신된 디바이스 델타언어모델을 포함할 수 있다.
- [0053] 상기 방법은 사용자를 인증하는 메카니즘을 포함할 수 있다.
- [0054] 바람직하기로, 상기 방법은 누적언어모델을 수신하고 디바이스 델타언어모델을 메카니즘으로 전송하기 전에 디바이스와 관련된 다이나믹 언어모델에 상기 누적언어모델을 병합하는 각 디바이스를 포함한다.
- [0055] 바람직하기로, 상기 방법은 디바이스와 관련된 특정 델타언어모델을 수신하고 상기 델타언어모델을 상기 디바이스의 다이나믹 언어모델에 병합하는 각 디바이스를 포함한다.
- [0056] 메카니즘은 바람직하게는 서버를 포함한다. 상기 방법은 서버와 복수의 디바이스들 각각 간에 보안 연결을 확립하는 단계를 더 포함한다.
- [0057] 언어모델은 용어 시퀀스를 각 시퀀스에 대한 발생 빈도와 연관시키는 데이터 구조를 포함할 수 있다. 상기 방법에서, 제 1 언어모델을 제 2 언어모델과 병합시키는 단계는: 제 2 언어모델의 데이터 구조에서 시퀀스에 대한 발생빈도를 제 1 언어모델의 데이터 구조에서 해당 시퀀스에 대한 발생빈도에 추가하는 단계; 및 시퀀스가 제 2 언어모델의 데이터 구조에 있으나 제 1 언어모델의 데이터 구조에 없다면, 새 시퀀스와 해당 발생빈도를 제 1 언어모델의 데이터 구조에 삽입하는 단계를 포함한다. 상기 방법은 하나 이상의 시퀀스들이 발생 빈도가 문턱값 아래로 떨어지면 하나 이상의 시퀀스들을 병합된 데이터 구조에서 제거하는 복수의 디바이스들 각각을 더 포함할 수 있다.
- [0058] 본 발명의 제 4 태양으로, 프로세서가 상술한 바와 같은 방법들 중 어느 하나를 실행하도록 하는 컴퓨터 프로그램 수단에 저장된 컴퓨터 판독가능 매체를 구비한 컴퓨터 프로그램 제품이 제공된다.
- [0059] 본 발명의 제 5 태양으로, 메카니즘과 디바이스 간에 주기적으로 동기화하는 방법이 제공된다. 상기 방법은, 메카니즘에, 디바이스의 마지막 동기화 동안 상기 디바이스로 전송된 데이터 번들을 식별하는 파라미터를 저장하는 단계; 메카니즘에, 디바이스의 마지막 동기화 이후로 데이터에 대한 업데이트를 포함하는 1차 데이터 번들을 저장하는 단계; 메카니즘에, 디바이스의 마지막 동기화 동안 디바이스로 전송된 데이터 번들과 디바이스의 마지막 동기화 이후로 데이터에 대한 업데이트를 포함하는 백업 데이터 번들을 저장하는 단계; 메카니즘에, 디바이스에 의해 마지막에 수신된 데이터 번들을 식별하고 이를 저장된 파라미터와 비교하는 디바이스로부터 파라미터를 수신하는 단계; 및 저장되고 수신된 파라미터들이 동일하다면 1차 데이터 번들을 전송하거나, 저장되고 수신된 파라미터들이 동일하지 않다면 백업 데이터 번들을 전송하는 단계를 포함한다.

## 발명의 효과

- [0060] 본 발명의 내용에 포함됨.

## 도면의 간단한 설명

- [0061] 첨부도면을 참조로 본 발명을 더 상세히 설명한다:

도 1은 본 발명에 따른 복수의 디바이스들을 도시한 것이다.



도 2는 본 발명에 따른 동기화 전후 2개 디바이스들에 의해 발생된 예상의 일예를 도시한 것이다.

도 3a는 본 발명의 제 1 실시예에 따른 디바이스들(A,B,C)의 초기 동기화 동안 데이터 전송을 도시한 것이다.

도 3b는 본 발명의 제 1 실시예에 따른 디바이스들(A,B,C)의 연속 동기화 동안 데이터 전송을 도시한 것이다.

도 3c는 도 3a 및 도 3b에 따른 디바이스(C)의 초기 및 연이은 동기화 동안 데이터 전송을 도시한 것이다.

도 4a는 본 발명의 제 2 실시예에 따른 디바이스들(A,B,C)의 초기 동기화 동안 데이터 전송을 도시한 것이다.

도 4b는 본 발명의 제 2 실시예에 따른 디바이스들(A,B,C)의 연속 동기화 동안 데이터 전송을 도시한 것이다.

도 4c는 도 4a 및 도 4b에 따른 디바이스(C)의 초기 및 연이은 동기화 동안 데이터 전송을 도시한 것이다.

도 5는 복수의 언어모델들을 동기화하는 메카니즘이 서버 기반인 본 발명의 바람직한 실시예를 도시한 것이다.

도 6은 디바이스의 관점에서 서버와 디바이스의 초기 동기화를 도시한 것이다.

도 7은 서버의 관점에서 서버와 디바이스의 초기 동기화를 도시한 것이다.

도 8은 디바이스의 관점에서 서버와 디바이스의 연속 동기화를 도시한 것이다.

도 9는 서버의 관점에서 서버와 디바이스의 연속 동기화를 도시한 것이다.

도 10은 서버의 관점에서 초기화 프로세스를 도시한 것이다.

도 11은 본 발명의 태양에 따른 예러완화를 이용한 시스템에서 동기화 동안 데이터 전송을 도시한 것이다.

도 12는 예러완화가 이용되는 시스템에서 서버의 관점에서 서버와 디바이스의 동기화 동안 디바이스에 델타 언어모델의 전달을 도시한 것이다.

### 발명을 실시하기 위한 구체적인 내용

- [0062] 본 발명은 복수의 디바이스들에 있는 복수의 다이나믹 언어모델들을 동기화하는 메카니즘을 제공한다. 각 디바이스는 다이나믹 언어모델을 포함하며, 이 다이나믹 언어모델은 사용자가 작성하고자 하는 말을 더 정확히 예상하기 위해 사용자 입력 텍스트와 더불어 진화한다.
- [0063] 다이나믹 언어모델을 동기화함으로써, 시스템은 상기 시스템의 디바이스들을 가로질러 발생한 텍스트 예상의 일치를 개선할 수 있어, 다수의 디바이스들이 비슷한 단어 예상 및 교정을 제공하게 한다. 따라서, 본 발명은 각 디바이스는 동일한 사용자가 사용한/동일한 사용자에게 의해 사용된 다른 디바이스들로부터 학습하기 때문에 정확한 예상을 제공하는데 있어 거의 사용되지 않는 디바이스가 기능이 뒤떨어지는 문제를 해결한다.
- [0064] 도 1은 본 발명에 따른 복수의 디바이스들(10,10')을 도시한 것이다. 각 디바이스(10,10')는 다이나믹 언어모델(1,1')을 포함한다. 바람직하기로, 각 디바이스(10,10')는 다이나믹 언어모델(1,1') 및 선택적으로 하나 이상의 스태틱 언어모델(2,2')을 포함하는 텍스트 입력 시스템(5,5')을 포함한다. 스태틱 언어모델(2)은 일단 훈련되고 나면 사용자 입력과 더불어 진화하지 않는 언어모델이다.
- [0065] 이러한 텍스트 입력 시스템(5,5')의 예가 국제특허출원 W02010/112841 "System and method for inputting text into electronic devices"에 기술되어 있으며, 상기 문헌은 전체적으로 본 명세서에 참조로 합체되어 있다. 이 참고문헌은 또한 예상이 추후 더 상세히 설명된 2 이상의 언어모델들, 가령, 다이나믹 언어모델(1,1') 및 하나 이상의 스태틱 언어모델(2,2')로부터 발생되거나 조합될 수 있는 방식을 언급한다.
- [0066] 각 다이나믹 언어모델(1,1')은 n-그램 맵에서 사용자가 입력한 n-그램 경로들의 발생 빈도를 기록하기 위해 업데이트되는 n-그램 언어모델을 기초로 한다. n-그램 맵은 연상 맵 구조이다. n-그램 맵에서, 어휘의 용어들은 맵에 저장되고 빈도 또는 확률값과 관련된 수치 식별자(쇼트 정수)와 관련된다.
- [0067] 디바이스의 다이나믹 언어모델(1,1')은 증거의 2가지 추가 타입들을 고려하기 위해 시간에 걸쳐 진화할 능력을 갖는다. 먼저, 다이나믹 언어모델(1,1')은 사용자가 입력한 문자 시퀀스로부터 발생된 통계적 언어 데이터를 동화시킨다. 두번째로, 다이나믹 언어모델(1,1')은 또 다른 언어모델의 통계적 언어 데이터를 동화시킨다. 이 적용에서, 한 언어모델을 또 다른 언어모델의 통계적 언어 데이터로의 동화를 "병합(merging)"이라 한다.
- [0068] 다이나믹 언어모델은 새 경로를 n-그램 데이터 삽입함으로써 다이나믹 언어모델 어휘에 이전에 없는 용어를 포함하기 위해, 그리고 특히 n-그램과 관련해 기존 용어의 빈도를 업데이트하기 위해 2가지 방식들 중 하나로 사

용자가 입력한 문자 시퀀스로부터 발생한 통계적 언어 데이터를 동화시킨다. 다이나믹 n-그램 맵은 n-그램 경로들이 사용자에게 의해 입력되는 빈도를 저장하며, 'n-그램 경로'는 특정 용어 및 선행 문맥의 n-1 용어들까지를 말한다.

[0069] 제 1 언어모델은 제 2 언어모델의 데이터 구조에 있어 시퀀스들에 대한 발생 빈도들을 제 1 언어모델의 데이터 구조에 있어 해당 시퀀스들에 대한 발생 빈도들에 추가함으로써 제 2 언어모델과 병합된다. 더욱이, 상기 시퀀스가 제 2 언어모델의 데이터 구조에 있으나 제 1 언어모델의 데이터 구조에 없다면, 새 시퀀스와 그 시퀀스에 해당하는 발생 빈도가 제 1 언어모델의 데이터 구조에 삽입된다.

[0070] 다이나믹 언어모델들의 크기를 제한하기 위해, 낮은 확률 이벤트들이 다이나믹 언어모델들로부터 가지쳐질 수 있다. 가령, 그 빈도가 소정 문턱값 미만이라면 다이나믹 언어모델을 통한 경로가 가지쳐질 수 있다. 가지치기(pruning)은 하기의 방식에 따라 발생한다:

[0071] 1) 모든 이벤트 빈도들은 하기의 함수에 따라 저하된다:

[0072]  $f' = \text{int}(d*/f)$

[0073] 2) 빈도가 기정의된 문턱값 미만인 모든 이벤트들이 제거된다. 즉:

[0074]  $f' < t$

[0075] f는 본래 빈도이고, f'는 저하된 빈도이며,  $\text{int}(x)$ 는  $i < x$ 이도록 가장 큰 정수(i)이고, d는 (가령, 0.9로 설정된) 소멸인자, 및 t는 (가령 1로 설정된) 문턱값이다.

[0076] 언어모델(1,1',2,2')은 바람직하게는 예상 언어모델 추론의 원리로 구성되며,  $P(s|c,m)$  형태의 확률평가를 제공하기 위해 히스토리컬 연속 증거에 훈련된 문맥상 시퀀스(c), 및 모델(m)이 주어진다면 특정 문자 시퀀스(s)의 확률이 평가된다. 다이나믹 언어모델(1,1')에 대해, 모델(m)은 (다이나믹 언어모델(1,1')은 사용자에게 의해 디바이스에 입력된 임의의 텍스트에 대해 디바이스(10,10')에 의해 점차 훈련되기 때문에) 사용자가 텍스트를 디바이스(10,10')에 입력하는 시간에 따라 진화한다.

[0077] 따라서, 시간에 따라, 사용자가 디바이스(10,10')에 텍스트를 입력하면, 내재한 다이나믹 언어모델(1,1')이 사용자 언어 스타일에 대해 진화하기 때문에 텍스트 입력 시스템(5,5')에 의해 발생한 교정 및 예측의 정확도가 향상될 것이다.

[0078] 도 2는 동기화 후 일치하게 된 텍스트 입력 경험의 일예를 제공한다. 동기화 전에, 디바이스(B)는 (디바이스(A)에 이전에 텍스트 입력을 통해) 디바이스(A)가 학습한, 이 예에서 강아지 이름 "Beau-Beau"라는 말을 예상할 수 없다. 동기화 후, 디바이스(B)도 또한 이름 "Beau-Beau"를 예상할 수 있다.

[0079] 동기화 후, 동기화된 디바이스들(10,10') 모두가 이들의 동기화된 다이나믹 언어모델(1,1')에 의해 동일한 사용자 입력에 응답해 동일한 예측을 발생할 수 있다. 그러나, 디바이스(10,10')는 동일한 텍스트 예측을 실제로 발생할 수 없는데, 이는 각 디바이스(10,10')가 다른 텍스트들에 대해 초기에 훈련된 다이나믹 언어모델들을 가질 수 있고, 또한 사용자 입력과 더불어 진화하지 않는 하나 이상의 스태틱 언어모델들(2)을 선택적으로 포함할 수 있고 다른 디바이스들에서 다를 수 있기 때문이다.

[0080] 텍스트 예측은 스태틱 언어모델(들)(2,2')뿐만 아니라 다이나믹 언어모델(1,1')을 포함할 수 있는 텍스트 엔트리 시스템(5,5')으로부터 발생되기 때문에, 제 1 디바이스(10)의 텍스트 엔트리 시스템(5)은 제 1 디바이스(10)와 동기화된 다이나믹 언어모델(1')을 가진 제 2 디바이스(10')의 텍스트 엔트리 시스템(5')에 대해 다른 예측을 발생할 수 있다. 다이나믹 언어모델(1')과 하나 이상의 스태틱 언어모델(들)(2,2')로부터의 예측을 조합한 텍스트 엔트리 시스템(5,5')이 국제특허출원 W02010/112841, "System and method for inputting text into electronic devices"에 기술되어 있으며, 상기 참조문헌은 전체적으로 본 명세서에 참조로 합체되어 있다. 이 참조문헌에 기술된 바와 같이, 텍스트 예측 엔진(가령, 텍스트 엔트리 시스템(5,5'))은 다중언어모델로부터 동시에 텍스트 예측을 발생하도록 구성될 수 있다. 다중언어모델(Multi-LM)을 이용해 디스플레이 및 사용자 선택을 위해 사용자 인터페이스에 제공되는 최종 예측을 발생하기 위해 다중언어모델들 각각에서 나온 예측들을 조합함으로써 이렇게 한다. 최종 예측은 전반적으로 가장 가능성 있는 예측들의 세트(즉, 특정 번호)이다. Multi-LM은 각 언어모델로부터의 예측을 순위 매긴 STL '멀티맵' 구조일 수 있는 순위 매긴 연대 구조에 삽입함으로써 최종 예측을 발생한다.

[0081] 예로써, 제 1 언어모델에서 예측이 "a"→0.2 및 "the"→0.3이고 제 2 언어모델, 가령, 텍스트 입력 시스템(5)의



스태틱 언어모델(2)에서 예상이 "a"→0.1 및 "these"→0.2이면, Multi-LM은 엔트리들이 이들의 확률((0.1→"an"), (0.2→"a"), (0.2→"these"), (0.3→"the"))로 순위 매겨지도록 이들 예상들을 순위 매긴 연대구조에 삽입한다. 이 구조는 최종 '가장 가능성 있는' 예상들의 세트를 얻기 위해 상한치로부터 순차적으로 읽어들 수 있다.

- [0082] 도 3a-9를 참조로 중앙 메카니즘을 갖는 복수의 디바이스들의 동기화를 논의할 것이다.
- [0083] 도 3a 및 도 3b는 시스템(1000)의 메카니즘(500)이 본 발명에 따라 디바이스(A,B,C)와 동기화하는 동안 시스템(1000)에서 데이터의 전송을 도시한 것이다. 데이터 전송은 전송되고 있는 객체(원으로 표시됨)에서 시작하고 전송된 객체(실선 화살표로 표시됨)에서 끝나는 대시선으로 도시되어 있다. 도 3a는 메카니즘(500)으로 디바이스(A,B,C)의 초기 동기화 동안 데이터 전송을 도시한 것이다. 도 3b는 메카니즘(500)으로 디바이스(A,B,C)의 연속 동기화 동안 데이터 전송을 도시한 것이다. 메카니즘(500)으로 시스템의 모든 디바이스들(A,B,C)의 동기화는 상기 디바이스들에 있는 복수의 언어모델들의 동기화를 제공한다.
- [0084] 도 3a 및 도 3b에 도시된 시스템(1000)은 다이나믹 언어모델(101,102,103) 및 사용자에게 의해 상기 디바이스들(A,B,C)에 입력된 텍스트( $T_A, T_B, T_C$ )를 각각 포함한 3개 디바이스들(A,B,C)을 포함한다. 시스템은 3개 디바이스들에 국한되지 않는다. 단일 디바이스, 2개 디바이스 또는 임의 개수의 추가 디바이스들을 포함할 수 있다.
- [0085] 시스템(1000)은 디바이스들(A,B,C)을 동기화하기 위한 메카니즘(500)을 포함한다. 메카니즘(500)은 각 디바이스들(A,B,C)과 관련된 누적언어모델(LM)(300)과 델타언어모델(201,202,203)을 포함하거나 생성한다. 누적언어모델(300)은 메카니즘(500)에 의해 디바이스들(A,B,C)에 입력된 임의의 텍스트에 대해 훈련된다. 각 델타언어모델(201,202,203)은, 아래에 더 상세히 기술된 바와 같이, 델타언어모델에 한정된 디바이스와는 다른 디바이스들(A,B,C)에 입력된 텍스트( $T_A, T_B, T_C$ )를 이용해 메카니즘에 의해 훈련된다.
- [0086] 디바이스들(A,B,C)은 임의의 시간에 그리고 종종 요망/필요에 따라 메카니즘(500)으로 동기화하는데 자유롭다. 각 디바이스(A,B,C)는 동기화를 위해 메카니즘(500)에 요청을 보낸다. 가령, 디바이스는 메카니즘으로 동기화를 위해 푸시/풀 신호를 제어한다. 동기화 요청에 응답해, 동기화 메카니즘(500)은 한번에 하나의 디바이스(A,B,C)와 동기화한다. 따라서, 동기화 메카니즘(500)은 동기화 요청이 어떤 순서로 수신되는지 간에 디바이스들(A,B,C)로부터 동기화 요청을 차례로 처리하도록 구성된다.
- [0087] 도 3a에 도시된 시스템(1000)은 각 디바이스(A,B,C)의 초기 동기화 동안 3개 디바이스들(A,B,C)과 동기화를 위한 메카니즘(500) 간에 데이터 전송을 도시한 것이다. 디바이스들은 임의 순서로 동기화 요청을 할 수 있고, 따라서, 메카니즘(500)은 임의 순서로 디바이스들의 동기화를 처리할 수 있다. 예컨대, 제 1 디바이스(A)는 메카니즘(500)으로 동기화된 다음(디바이스(A)의 초기 동기화), 디바이스(B)가 메카니즘(500)으로 동기화되고(디바이스(B)의 초기 동기화), 그런 후 디바이스(C)가 메카니즘(500)으로 동기화된다(디바이스(C)의 초기 동기화).
- [0088] 도 3b는 디바이스(A,B,C)의 연속 동기화 동안 데이터 전송을 도시한 것이다. '초기' 동기화라는 용어는 단일 디바이스의 초기 동기화를 말하며 복수의 디바이스들의 초기 동기화를 말하는 것이 아닌데, 복수의 디바이스들은 하나 이상의 디바이스들의 연속 동기화가 발생하기 전에 초기에 동기화될 필요가 없기 때문이다. 마찬가지로, '연속' 동기화는 단일 디바이스의 연속 동기화를 말한다. 즉, 디바이스가 초기 동기화를 수행한 후 발생하는 동기화를 말한다.
- [0089] 도 3a에 도시된 바와 같이, 디바이스의 초기 동기화 동안, 디바이스는 메카니즘(500)으로 이전에 동기화된 임의의 디바이스에 입력된 텍스트를 이용해 메카니즘(500)에 의해 발생된 누적언어모델(300)을 메카니즘(500)으로부터 수신한다. 디바이스는 누적언어모델(300)을 자신의 다이나믹 언어모델에 병합하도록 구성된다. 디바이스는 상기 디바이스에 입력된 임의의 텍스트를 메카니즘(500)으로 전송하도록 더 구성된다.
- [0090] 초기 동기화 프로세스는 도 3a에 도시된 특정 시스템에 대해 설명되어 있으며, 비제한적인 예로서, 디바이스들(A,B,C)은 차례로 메카니즘(500)과 동기화된다.
- [0091] 메카니즘(500)으로 디바이스(A)의 초기 동기화 동안, 디바이스(A)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드한다. 디바이스(A)는 처음으로 동기화되기 때문에, (누적언어모델(300)은 메카니즘(500)으로 이전에 동기화된 임의의 디바이스에 입력된 임의의 텍스트로 훈련되기 때문에) 누적언어모델(300)은 비어있다. 누적언어모델(300)을 다운로드한 후, 디바이스(A)는 자신의 다이나믹 언어모델(101)에 이를 병합시킨다. 이 상황에서, 병합으로 인해 다이나믹 언어모델(101)에는 전혀 변화가 일어나지 않는데, 이는 누적언어모델(300)이 비어있기 때문이다. 그런 후, 디바이스(A)는 사용자에게 의해 상기 디바이스(A)에 입력된 텍스트( $T_A$ )를 전송한다. 텍스트

( $T_A$ )는 디바이스에 입력된 임의의 이용가능한 텍스트, 가령, 디바이스의 메모리에 있으며 소거되지 않은 텍스트를 포함한다.

[0092] 이 초기 동기화 동안, 처음으로 사용자가 동기화를 수행했기 때문에, 메카니즘(500)은 바람직하게는 빈 누적언어모델(300)을 발생한다. 대안으로, 메카니즘(500)에 빈 누적언어모델(300)이 제공될 수 있다. 메카니즘(500)은 디바이스에 입력된 텍스트( $T_A$ )를 수신하고 상기 텍스트( $T_A$ )에 대해 빈 누적언어모델(300)을 훈련시킨다. 더욱이, 메카니즘(500)은 바람직하게는 디바이스(A)에 대해 빈 델타언어모델(201)을 생성한다. 명확히 하기 위해, 도 3a는 메카니즘(500)내 텍스트의 전송을 나타내지 않는다. 그러나, 메카니즘(500)내 텍스트의 전송의 일례가 도 3c의 디바이스(C)에 대해 도시되어 있으며, 아래에 설명되어 있다.

[0093] 메카니즘(500)으로 디바이스(B)의 초기 동기화 동안, 디바이스(B)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드한다. 누적언어모델(300)은 디바이스(A)에 입력된 텍스트( $T_A$ )를 나타낸다. 누적언어모델(300)을 다운로드한 후, 디바이스(B)는 이를 자신의 다이내믹 언어모델(102)에 병합한다. 디바이스(B)는 그런 후 사용자에게 의해 상기 디바이스(B)에 입력된 텍스트( $T_B$ )를 메카니즘(500)으로 전송한다.

[0094] 메카니즘(500)은 디바이스들에 텍스트( $T_A, T_B, T_C$ )를 수신하기 전에 누적언어모델(300)을 디바이스들(A,B,C)에 전송한다. 디바이스들(A,B,C)의 다이내믹 언어모델(101,102,103)은 사용자 텍스트 입력에 따라 진화한다. 누적언어모델(300)을 먼저 전송함으로써, 메카니즘은 디바이스들(A,B,C)이 다이내믹 언어모델(101,102,103)에 이미 있는 텍스트( $T_A, T_B, T_C$ )를 포함하지 않는 것을 확실히 한다. 이 순서의 중요성이 추후 더 상세히 언급된다.

[0095] 디바이스(B)의 초기 동기화 동안, 메카니즘(500)은 디바이스(B)에 입력된 텍스트( $T_B$ )를 수신하고 상기 텍스트( $T_B$ )에 대해 누적언어모델(300)을 더 훈련시킨다. 또한, 메카니즘(500)은 디바이스(B)로부터 수신한 텍스트( $T_B$ )에 대해 디바이스(A)와 관련된 델타언어모델(201)을 훈련시킨다. 게다가, 메카니즘(500)은 디바이스(B)에 대해 빈 델타언어모델(202)을 생성한다.

[0096] 메카니즘(500)으로 디바이스(C)의 초기 동기화 동안, 디바이스(C)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드한다. 누적언어모델(300)은 디바이스(A 및 B)에 입력된 텍스트( $T_A, T_B$ )를 나타낸다. 누적언어모델(300)을 다운로드한 후, 디바이스(C)는 이를 자신의 다이내믹 언어모델(203)에 병합한다. 디바이스(C)는 그런 후 상기 디바이스(C)에 입력된 텍스트( $T_C$ )를 메카니즘(500)으로 전송한다.

[0097] 디바이스(C)의 초기 동기화 동안, 메카니즘(500)은 디바이스(C)에 입력된 텍스트( $T_C$ )를 수신하고 상기 텍스트( $T_C$ )에 대해 누적언어모델(300)을 더 훈련시킨다. 또한, 메카니즘(500)은 디바이스(C)로부터 수신한 텍스트( $T_C$ )에 대해 디바이스(A)와 관련된 델타언어모델(201)을 더 훈련시킨다. 메카니즘(500)은 디바이스(C)로부터 수신한 텍스트( $T_C$ )에 대해 디바이스(B)와 관련된 빈 델타언어모델(202)을 훈련시킨다. 게다가, 메카니즘(500)은 디바이스(C)에 대해 빈 델타언어모델(203)을 생성한다.

[0098] 각 디바이스(A,B,C)는 바람직하게는 메카니즘(500)에 전송한 후 디바이스로부터 텍스트( $T_A, T_B, T_C$ )를 제거한다. 그런 후 디바이스(A,B,C)는 초기 동기화 이후 디바이스에 입력된 임의의 텍스트( $T_A, T_B, T_C$ )를 저장하며, 그런 후 이 텍스트는 연이은 동기화 등등 동안 전송된 후에 제거된다.

[0099] 시스템의 디바이스(A,B,C)의 연속 동기화가 도 3b에 도시되어 있다. 상술한 바와 같이, 디바이스들은 필요에 따라 동기화될 수 있다. 디바이스들의 동기화에 어떤 순서가 있을 필요는 없다. 가령, 디바이스(B)는 디바이스(C)가 초기 동기화를 요청하기 전에 연속 동기화를 요청할 수 있다. 따라서, 이 예에서, 디바이스(C)가 동기화 동안 수신한 델타언어모델(203)이 디바이스(B)로부터 수신된, 즉 디바이스(B)의 초기 동기화 동안 수신되고 디바이스(B)의 연속 동기화 동안 수신된 텍스트( $T_B$ )를 2개 로트들에 대해 훈련될 것이다.

[0100] 메카니즘(500)과 디바이스(A,B,C) 간에 연속 동기화가 수행되면, 디바이스(A,B,C)는 이전 동기화 이후 상기 디바이스에 입력된 임의의 텍스트( $T_A, T_B, T_C$ )를 메카니즘(500)으로 전송하며, 이전 동기화는 초기 동기화 또는 상기 초기 동기화에 연이은 동기화일 수 있다.

[0101] 각 디바이스(A,B,C)는 상기 디바이스에 한정된 델타언어모델(201,202,203)을 다운로드한다. 디바이스는 텍스트를 업로딩하는 것과 동시에 델타언어모델을 다운로드하는데, 이는 단계들이 독립적이기 때문이다(디바이스에 대

한 텔타언어모델은 다른 디바이스들에 입력된 텍스트에 대해 훈련되기 때문이다).

- [0102] 연속 동기화 동안, 메카니즘(500)은 이전 동기화 이후 입력된 텍스트( $T_A, T_B, T_C$ )를 수신한다. 이는 또한 텔타언어 모델(201, 202, 203)을 관련된 디바이스(A, B, C)에 전송한다. 텔타언어모델의 전송 및 텍스트의 수신은 주어진 디바이스에 대해 나란히 실행될 수 있다. 텔타언어모델이 디바이스에 전송된 후, 텔타언어모델(201, 202, 203)은 빈 텔타언어모델로 다시 소거될 수 있다.
- [0103] 도 3b에 도시된 디바이스의 연속 동기화를 이 도면을 참조로 상세히 설명한다. 작동예를 제공하기 위해, 디바이스들(A, B, C)은 상기 디바이스들이 모두 순서대로(즉, 디바이스 A, 다음에 디바이스 B, 다음에 디바이스 C) 초기 동기화를 진행한 후 상기 순서대로 연속 동기화를 요청하는 간단한 시나리오를 고려하자. 그러나, 상술한 바와 같이, 디바이스들은 임의의 순서로 임의의 시간에 동기화를 요청할 수 있다. 디바이스(A)의 연속 동기화에서, 디바이스(A)는 메카니즘(500)으로부터 디바이스(A)에 한정된 텔타언어모델(201)을 수신한다. 상술한 바와 같이, 이 텔타언어모델(201)은 디바이스(B 및 C)의 초기 동기화 동안 업로드된 디바이스(B 및 C)에 입력된 텍스트( $T_B, T_C$ )에 대해 훈련된다. 디바이스(A)는 그런 후 텔타언어모델(201)을 다이내믹 언어모델(101)에 병합한다.
- [0104] 메카니즘(500)은 바람직하게는 관련된 디바이스로 전송된 다음 각 텔타언어모델을 소거한다. 따라서, 텔타언어 모델(201)을 디바이스(A)로 전송한 후, 메카니즘(500)은 텔타언어모델(201)을 소거해, 디바이스(B 및 C)의 초기 동기화 이후 디바이스(B 및 C)에 입력된 텍스트( $T_B, T_C$ )에 대해 순차적으로 훈련되는 빈 텔타언어모델(201)을 제공한다.
- [0105] 디바이스(A)는, 텔타언어모델을 다운로드함과 동시에 반드시 필요한 것은 아니나, 초기 동기화 이후 디바이스(A)에 입력된 텍스트( $T_A$ )를 메카니즘(500)으로 전송한다. 메카니즘(500)은 이 텍스트( $T_A$ )를 수신하고 이 텍스트( $T_A$ )에 대해 디바이스(B 및 C)에 한정된 텔타언어모델(202, 203)을 더 훈련시킨다.
- [0106] 디바이스(B)의 연속 동기화 동안, 디바이스(B)에 한정된 텔타언어모델(202)이 메카니즘(500)으로부터 다운로드된다. 텔타언어모델(202)은 디바이스(C)의 초기 동기화 동안 메카니즘(500)에 의해 수신된 텍스트( $T_C$ ) 및 디바이스(A)의 초기 동기화 동안 메카니즘(500)에 의해 수신된 텍스트( $T_A$ )에 대해 훈련된다. 디바이스(B)는 텔타언어 모델(202)을 다이내믹 언어모델(102)에 병합시킨다. 메카니즘(500)이 텔타언어모델(202)을 디바이스(B)로 전송한 다음, 언어모델(202)을 소거해 디바이스(A 및 C)에 입력된 텍스트( $T_A, T_C$ )에 대해 순차적으로 훈련되는 빈 언어모델(202)을 제공한다.
- [0107] 디바이스(B)는 이전의 동기화 이후로, 가령, 이 시나리오에서 초기 동기화 이후로 디바이스(B)에 입력된 텍스트( $T_B$ )를 전송한다. 일단 전송된 후, 디바이스(B)는 바람직하게는 메모리로부터 텍스트를 소거한다. 메카니즘(500)은 텍스트( $T_B$ )를 수신하고, 상기 텍스트( $T_B$ )에 대해 디바이스(A)의 소거된 텔타언어모델(201)을 훈련시킨다. 더욱이, 메카니즘(500)은 상기 텍스트( $T_B$ )에 대해 디바이스(B)의 텔타언어모델(203)을 더 훈련시키며, 디바이스(C)에 대한 텔타언어모델은 디바이스(A)에 입력된 텍스트( $T_A$ )에 대해 이미 훈련되었다.
- [0108] 상술한 바와 같이 유사한 동기화 프로세스가 디바이스(C)의 연속 동기화에 대해 발생한다. 디바이스(C)에 대한 텔타언어모델(203)은 디바이스(A 및 C)의 초기 동기화 이후로 디바이스(A 및 B)에 입력된 텍스트( $T_A, T_B$ )에 대해 훈련된다.
- [0109] 상술한 것 이외에, 연속 동기화 동안, 메카니즘(500)은 바람직하게는 디바이스(A, B, C)의 연속 동기화 동안 수신한 텍스트( $T_A, T_B, T_C$ )에 대해 누적언어모델(300)을 훈련시키고 유지한다. 이는 추후 더 상세히 설명되는 바와 같이 메카니즘(500)으로 임의의 다른 디바이스의 빠르고 효율적인 동기화를 가능하게 한다.
- [0110] 특정 디바이스에 대해 초기 및 연속 동기화 동안 데이터의 전송이 디바이스(C)에 대해 도 3c 에 도시되어 있다. 초기 동기화 동안, 디바이스(C)는 누적언어모델(300)을 다운로드하고 이를 다이내믹 언어모델(103)에 병합시킨다. 디바이스(C)는 그런 후 상기 디바이스는 디바이스(A 및 B)와 관련된 텔타언어모델(201, 202)을 훈련시키고 누적언어모델(300)을 훈련시키기 위해 메카니즘(500)에 의해 사용되는 디바이스에 입력된 텍스트( $T_C$ )를 업로드한다. 연속 동기화 동안, 디바이스(C)는 텔타언어모델(203)을 다운로드하고 이를 다이내믹 언어모델(103)에 병합시킨다. 디바이스(C)는 이전 동기화 이후로 디바이스(C)에 입력되고 메카니즘(500)은 누적언어모델(300)과 디바이스(A 및 B)와 관련된 텔타언어모델(201, 202)을 훈련시키는데 이용하는 임의의 텍스트( $T_C$ )를 메카니즘(500)

에 업로드시킨다.

- [0111] 디바이스가 메카니즘으로 동기화되면, 본 발명에 따르면, 디바이스를 사용자가 전에 결코 사용한 적이 없더라도, 시스템의 다른 디바이스들에 입력된 모든 텍스트로부터 학습했기 때문에 사용자의 쓰기 스타일을 정확히 예상할 수 있다.
- [0112] 상기에서 명백한 바와 같이, 새 디바이스, 가령 디바이스(D)가 메카니즘(500)과 동기화되면, 어떤 특별한 초기 동기화를 수행할 필요가 없다. 디바이스(A, B, 및 C)와 이전에 동기화된 메카니즘(500)으로 디바이스(D)의 초기 동기화 동안, 디바이스(D)는 디바이스(A,B,C)에 입력된 모든 텍스트에 대해 훈련된 누적언어모델(300)을 수신한다. 디바이스(D)는 다이나믹 언어모델(103)에 누적언어모델(300)을 병합함으로써, 디바이스(A,B,C)에 입력된 모든 텍스트로부터 학습한다. 모든 다른 단계들은 동기화 동안, 가령, 초기 동기화 동안, 동일하게 유지된다. 메카니즘(500)은 디바이스(D)에 대해 빈 델타언어모델을 생성하고, 연속 동기화 동안, 디바이스(A,B,C)에 입력된 텍스트( $T_A, T_B, T_C$ )에 대해 델타언어모델을 훈련시킨다.
- [0113] 상기에서 명백한 바와 같이, 메카니즘(500)은 바람직하게는 텍스트가 메카니즘(500)에 의해 수신되는 순서대로 텍스트와 더불어 언어모델을 훈련시킨다. 예컨대, 디바이스(A)는 메카니즘(500)으로 동기화한 다음 디바이스(C)가 동기화하면, 디바이스(B)에 대한 델타언어모델은 디바이스(A)로부터 수신된 텍스트( $T_A$ )에 대해 훈련되고 그런 후 디바이스(C)로부터 수신된 텍스트( $T_C$ )에 대해 훈련된다.
- [0114] 몇몇 실시예에서, 관련된 델타언어모델(201,202,203) 및/또는 누적언어모델(300)은 텍스트가 동기화 동안 수신될 경우 텍스트( $T_A, T_B, T_C$ )에 대해 훈련될 수 없다. 이들 실시예에서, 언어모델이 장래의 동기화 동안 디바이스에 전송되는 것이 필요할 때에만, 수신된 텍스트( $T_A, T_B, T_C$ )가 메카니즘(500)에 의해 저장되고 델타훈련모델(201,202,203) 및/또는 누적언어모델(300)을 훈련시키는데 이용된다.
- [0115] 본 발명에 따른 대안적인 데이터 전송 실시예가 도 4a, 4b 및 4c에 도시되어 있다. 이 대안적인 실시예는 실질적으로 도 3a, 3b 및 3c를 참조로 상술한 실시예와 유사하나, 사용자가 디바이스에 입력한 미가공 텍스트( $T_A, T_B, T_C$ )의 전송보다는 디바이스(A,B,C)로부터 메카니즘으로 디바이스 델타언어모델(111,112,113)의 전송에 의해 디바이스(A,B,C)의 동기화가 달성되게 한다. 디바이스 델타언어모델(111,112,113)은 디바이스(A,B,C)에 한정되나 메카니즘(500)에 저장된 델타언어모델(201,202,203)과는 대조적으로 디바이스(A,B,C)에 저장된 다이나믹 언어모델이다.
- [0116] 도 4a 및 도 4b는 시스템(1000)의 메카니즘(500)이 디바이스(A,B,C)와 동기화하는 동안 시스템(1000)에서 데이터의 전송을 도시한 것이다. 도 4a는 메카니즘(500)으로 디바이스(A,B,C)의 초기 동기화 동안 데이터 전송을 도시한 것이다. 도 4b는 메카니즘(500)으로 디바이스(A,B,C)의 연속 동기화 동안 데이터 전송을 도시한 것이다. 메카니즘(500)으로 시스템의 모든 디바이스들(A,B,C)의 동기화는 상기 디바이스에 있는 복수의 언어모델들의 동기화를 제공한다.
- [0117] 도 4a 및 도 4b에 도시된 시스템(1000)은 3개의 디바이스들(A,B,C)을 구비하고, 각각은 다이나믹 언어모델(101,102,103)과 디바이스 델타언어모델(111,112,113)을 포함한다. 시스템은 3개의 디바이스들에 국한되지 않는다; 단일 디바이스, 2개의 디바이스 또는 임의 개수의 추가 디바이스들을 포함할 수 있다.
- [0118] 디바이스 델타언어모델(111,112,113)은 메카니즘(500)으로 디바이스의 마지막 동기화 이후 상관된 디바이스(A,B,C)에 입력된 텍스트에 대해 훈련된 다이나믹 언어모델이다. 즉, 디바이스 델타언어모델(111,112,113)은 메카니즘(500)으로 디바이스의 마지막 동기화 이후 상관된 디바이스(A,B,C)에 입력된 미가공 텍스트( $T_A, T_B, T_C$ )에 대해 훈련된다.
- [0119] 시스템(1000)은 디바이스(A,B,C)를 동기화하기 위한 메카니즘(500)을 포함한다. 메카니즘(500)은 누적언어모델(LM)(300) 및 각 디바이스(A,B,C)와 관련된 델타언어모델(201,202,203)을 포함한다. 메카니즘(500)은 디바이스(A,B,C)에 입력된 임의의 텍스트를 나타내는 디바이스 델타언어모델(111,112,113)을 누적언어모델(300)에 병합시킨다.
- [0120] 동기화 동안, 각 디바이스 델타언어모델(111,112,113)은, 하기에 더 상세히 기술된 바와 같이, 메카니즘(500)에 의해 상기 디바이스 델타언어모델(111,112,113)이 기원한 디바이스(A,B,C)에 한정된 델타언어모델(201,202,203)과는 다른 모든 델타언어모델(201,202,203)에 병합된다.



- [0121] 디바이스(A,B,C)는 임의의 시간에 그리고 종종 요구/필요에 따라 메카니즘(500)으로 동기화하는데 자유롭다. 도 4a에 도시된 시스템(1000)은 각 디바이스(A,B,C)의 초기 동기화 동안 동기화를 위해 3개 디바이스(A,B,C)와 메카니즘(500) 간에 데이터 전달을 도시한 것이다. 디바이스는 임의의 순서로 동기화를 요청할 수 있다.
- [0122] 도 4a에 도시된 바와 같이, 디바이스의 초기 동기화 동안, 디바이스는 메카니즘(500)으로 이전에 동기화된 임의의 디바이스에 입력된 텍스트로부터 도출된 데이터를 이용해 메카니즘(500)에 의해 발생된 누적언어모델(300)을 메카니즘(500)으로부터 수신한다. 디바이스는 누적언어모델(300)을 자신의 다이나믹 언어모델에 병합하도록 구성된다. 디바이스는 상기 디바이스에 입력된 임의의 텍스트로부터 데이터를 포함한 디바이스 텔타언어모델(111,112,113)을 메카니즘(500)으로 전송하도록 더 구성된다.
- [0123] 비제한적인 예로, 디바이스들(A,B,C)이 차례로 메카니즘(500)과 동기화되는 도 4a에 예시된 특정 시스템에 대해 초기 동기화 프로세스를 설명한다.
- [0124] 메카니즘(500)으로 디바이스(A)의 초기 동기화 동안, 디바이스(A)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드하고 이를 자신의 다이나믹 언어모델(101)에 병합하도록 구성된다. 디바이스(A)는 그런 후 디바이스 텔타언어모델(111)을 전송하며, 상기 디바이스 텔타언어모델은 사용자에게 이해 디바이스(A)에 입력된 텍스트에 대해 훈련되어 있다. 디바이스 텔타언어모델(111)은 디바이스에 입력된 임의의 이용가능한 텍스트, 가령, 디바이스의 메모리에 있으며 소거되지 않은 텍스트에 대해 훈련될 수 있다.
- [0125] 이 초기 동기화 동안, 메카니즘(500)은 바람직하게는 빈 누적언어모델(300)을 발생하거나, 메카니즘(500)에 빈 누적언어모델(300)이 제공될 수 있다. 메카니즘(500)은 디바이스 텔타언어모델(111)을 수신하고 상기 디바이스 텔타언어모델(111)을 빈 누적언어모델(300)에 병합시킨다. 더욱이, 메카니즘(500)은 디바이스(A)에 한정된 빈 텔타언어모델(201)을 생성한다.
- [0126] 메카니즘(500)과 디바이스(B)의 초기 동기화 동안, 디바이스(B)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드한다. 누적언어모델(300)은 디바이스(A)에 입력되었던 텍스트에 대해 훈련된 디바이스 텔타언어모델(111)로부터 데이터를 포함한다. 누적언어모델(300)을 다운로드한 후, 디바이스(B)는 자신의 동적언어모델(102)에 이를 병합시킨다. 이후 디바이스(B)는 사용자가 디바이스(B)에 입력한 텍스트에 대해 훈련된 텔타언어모델(112)을 메카니즘(500)으로 전송한다.
- [0127] 각 디바이스의 동기화 동안, 메카니즘(500)은 디바이스에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(111,112,113)을 수신하기 전에 누적언어모델(300)을 디바이스(A,B,C)로 전송한다. 누적언어모델(300)을 먼저 전송함으로써, 메카니즘은 디바이스(A,B,C)가 다이나믹 언어모델(101,102,103)에 이미 있는 디바이스 텔타언어모델(111,112,113)로부터의 데이터를 포함하지 않는 것을 보장한다. 이 순서의 중요성은 추후 더 상세히 언급된다.
- [0128] 디바이스(B)에 대한 초기 동기화 동안, 메카니즘(500)은 디바이스(B)에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(112)을 수신하고 상기 디바이스 텔타언어모델(112)을 누적언어모델(300)과 병합시킨다. 또한, 메카니즘(500)은 디바이스(B)로부터 수신된 디바이스 텔타언어모델(112)을 디바이스(A)와 관련된 텔타언어모델(201)과 병합시킨다. 게다가, 메카니즘(500)은 디바이스(B)용의 빈 텔타언어모델(202)을 생성한다.
- [0129] 메카니즘(500)으로 디바이스(C)의 초기 동기화 동안, 디바이스(C)는 메카니즘(500)으로부터 누적언어모델(300)을 다운로드한다. 누적언어모델(300)은 디바이스(A 및 B)에 입력된 텍스트를 나타낸다. 누적언어모델(300)을 다운로드한 후에, 디바이스(C)는 이를 자신의 다이나믹 언어모델(203)에 병합시킨다. 그런 후 디바이스(C)는 사용자가 디바이스(C)에 입력한 텍스트에 대해 훈련된 텔타언어모델(113)을 메카니즘(500)으로 전송한다.
- [0130] 이 디바이스(C)의 초기 동기화 동안, 메카니즘(500)은 디바이스(C)에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(113)을 수신하고 상기 디바이스 텔타언어모델(113)을 누적언어모델(300)과 병합시킨다. 또한, 메카니즘(500)은 디바이스(C)로부터 수신된 디바이스 텔타언어모델(113)을 디바이스(A)와 관련된 텔타언어모델(201)과 병합시킨다. 메카니즘(500)은 디바이스(C)로부터 수신된 디바이스 텔타언어모델(113)을 디바이스(B)용의 빈 텔타언어모델(202)과 더 병합시킨다. 게다가, 메카니즘(500)은 디바이스(C)와 관련된 빈 텔타언어모델(203)을 생성한다.
- [0131] 각 디바이스(A,B,C)는 바람직하게는 메카니즘(500)으로 전송된 다음 디바이스 텔타언어모델(111,112,113)을 빈 디바이스 텔타언어모델로 비워버린다. 그 다음, 디바이스(A,B,C)는 초기 동기화 이후 디바이스에 입력된 임의의 텍스트에 대해 디바이스 텔타언어모델(111,112,113)을 훈련시킨다. 디바이스 텔타언어모델(111,112,113)은 연속

동기화 등등 동안 전송된 후 다시 비워진다.

- [0132] 시스템의 디바이스(A,B,C)의 연속 동기화가 도 4b에 도시되어 있다. 상술한 바와 같이, 디바이스는 필요할 때마다 동기화될 수 있고, 디바이스의 동기화에 대해 어떤 순서가 있을 필요가 없다.
- [0133] 메카니즘(500)과 디바이스(A,B,C) 간에 연속 동기화가 수행되면, 상기 디바이스(A,B,C)는 이전 동기화 이후 상기 디바이스에 입력된 임의 텍스트에 대해 훈련된 디바이스 텔타언어모델(111,112,113)을 메카니즘(500)으로 보낸다.
- [0134] 각 디바이스(A,B,C)는 상기 디바이스에 한정된 텔타언어모델(201,202,203)을 다운로드한다. 디바이스는 디바이스 텔타언어모델(111,112,113)을 업로딩하는 것과 동시에 텔타언어모델(201,202,203)을 다운로드할 수 있고, 바람직하게는 그렇게 한다.
- [0135] 연속 동기화 동안, 메카니즘(500)은 이전 동기화 이후로 디바이스(A,B,C)에 입력된 텍스트를 나타내는 디바이스 텔타언어모델(111,112,113)을 수신한다. 이는 또한 텔타언어모델(201,202,203)을 관련된 디바이스(A,B,C)로 전송한다. 디바이스 텔타언어모델(111,112,113)의 수신과 텔타언어모델(201,202,203)의 송신은 주어진 장치에 대해 나란히 수행될 수 있다. 텔타언어모델(201,202,203)이 디바이스에 전송된 후, 상기 텔타언어모델(201,202,203)은 빈 텔타언어모델로 다시 비워질 수 있다.
- [0136] 디바이스가 순서대로 모두 초기 동기화된 후, 디바이스(A,B,C)가 순서대로 연속 동기화를 요청하는, 도 3b와 같은 시나리오로 여겨지는 도면을 참조로 도 4b에 도시된 연속 동기화를 상세히 설명한다. 그러나, 상술한 바와 같이, 디바이스는 임의의 시간에 임의의 순서로 동기화를 요청할 수 있다. 디바이스(A)의 연속 동기화에 있어, 디바이스(A)는 (디바이스(A)의 마지막 동기화 이후 디바이스(C)에 입력된 텍스트를 나타내는 디바이스 텔타언어모델(113)과 병합된 디바이스(A)의 마지막 동기화 이후 디바이스(B)에 입력된 텍스트를 나타내는 디바이스 텔타언어모델(112)인) 디바이스(A)에 한정된 텔타언어모델(201)을 메카니즘(500)으로부터 수신하고 텔타언어모델(201)을 다이내믹 언어모델(101)에 병합시킨다.
- [0137] 메카니즘(500)은 바람직하게는 관련된 디바이스에 전송된 이후 각 텔타언어모델(201,202,203)을 삭제한다.
- [0138] 디바이스(A)는 텔타언어모델과 동시에 다운로드하는 것이 바람직하나 반드시 그럴 필요 없이 초기 동기화 이후 디바이스(A)에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(111)을 메카니즘(500)으로 전송한다. 메카니즘(500)은 디바이스 텔타언어모델(111)을 수신하고 상기 디바이스 텔타언어모델(111)을 디바이스(B 및 C)에 한정된 텔타언어모델(202,203)에 병합시킨다.
- [0139] 디바이스(B)의 연속 동기화 동안, 디바이스(B)에 한정된 텔타언어모델(202)이 메카니즘(500)으로부터 다운로드되고 다이내믹 언어모델(102)에 병합시킨다. 메카니즘(500)이 텔타언어모델(202)을 디바이스(B)로 전송한 다음, 언어모델(202)을 삭제한다.
- [0140] 디바이스(B)는 이전 동기화 이후, 가령, 이 시나리오에서 초기 동기화 이후, 상기 디바이스(B)에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(112)을 디바이스(B)로 전송한다. 일단 전송된 다음, 디바이스(B)는 바람직하게는 디바이스 텔타언어모델(112)을 삭제하여 빈 텔타언어모델을 생성한다. 메카니즘(500)은 디바이스 텔타언어모델(112)을 수신하고 상기 디바이스 텔타언어모델(112)을 디바이스(A)용의 삭제된 텔타언어모델(201)에 병합시킨다. 더욱이, 메카니즘(500)은 디바이스 텔타언어모델(112)을 디바이스(C)용의 텔타언어모델(203)에 병합시키며, 디바이스(C)용의 텔타언어모델(203)은 디바이스(A)에 입력된 텍스트로부터 도출된 데이터를 이미 포함한다.
- [0141] 상술한 바와 같이 유사한 동기화 프로세스가 디바이스(C)의 연속 동기화에 대해 발생한다.
- [0142] 상술한 것 이외에, 연속 초기화 동안, 메카니즘(500)은 바람직하게는 디바이스(A,B,C)의 연속 동기화 동안 수신한 디바이스 텔타언어모델(111,112,113)을 병합함으로써 누적언어모델(300)을 유지한다.
- [0143] 특정한 디바이스에 대한 초기화 및 연속 동기화 동안 데이터의 전달이 도 4c의 디바이스(C)에 대해 도시되어 있다. 초기 동기화 동안, 디바이스(C)는 누적언어모델(300)을 다운로드하고 이를 다이내믹 언어모델(103)에 병합시킨다. 그런 후 디바이스(C)는 상기 디바이스에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델(113)을 업로드한다. 메카니즘(500)은 디바이스 텔타언어모델(113)을 디바이스(A 및 B)와 관련된 텔타언어모델(201,202)과 누적언어모델(300)에 병합시킨다. 연속 동기화 동안, 디바이스(C)는 텔타언어모델(203)을 다운로드하고 이를 다이내믹 언어모델(103)에 병합시킨다. 디바이스(C)는 이전의 동기화 이후로 디바이스(C)에 입력된 임의의 텍스트에 대해 훈련된 디바이스 텔타언어모델(113)을 메카니즘(500)으로 업로드시키며, 메카니즘(500)은 누적언어

어모델(300)과 디바이스(A 및 B)와 관련된 텔타언어모델(201,202)에 병합시킨다.

- [0144] 상기로부터 명백한 바와 같이, 새 디바이스, 가령 디바이스(D)가 메카니즘(500)과 동기화되면, 임의의 특수한 초기 동기화를 수행할 필요가 없다.
- [0145] 메카니즘(500)은 바람직하게는 디바이스 텔타언어모델(111,112,113)이 메카니즘에 의해 수신되는 순서대로 상기 디바이스 텔타언어모델(111,112,113)을 언어모델에 병합시킨다. 예컨대, 디바이스(A)가 메카니즘(500)과 동기화한 다음 디바이스(C)가 동기화되면, 디바이스(A)로부터 수신된 디바이스 텔타언어모델(111)은 디바이스(B)용 텔타언어모델(202)에 병합될 것이며, 그런 다음 디바이스(C)로부터 수신된 디바이스 텔타언어모델(113)이 텔타언어모델(202)에 병합될 것이다.
- [0146] 본 발명의 상술한 태양에서, 텔타언어모델이 입력 텍스트 대신에 메카니즘으로 전송되면, 디바이스의 다이나믹 언어모델과 디바이스 텔타언어모델은 사용자가 입력한 텍스트에 대해 훈련될 수 있다. 바람직한 실시예에서, 이 훈련은 동시에 발생한다. 다이나믹 언어모델과 텔타언어모델 모두가 입력 후 바로 입력 텍스트에 훈련되면, 사용자가 입력한 실제 텍스트를 저장할 필요가 없다.
- [0147] 다른 실시예에서, 마지막 동기화 이후 텍스트 입력이 디바이스에 저장될 수 있고, 디바이스 텔타언어모델은 상기 디바이스 텔타언어모델이 전송되는 연속 동기화 전에 훈련될 수 있다.
- [0148] 상기 실시예의 변형으로, 동기화 동안 디바이스 텔타언어모델(111,112,113)이 수신되면, 디바이스 텔타언어모델(111,112,113)은 관련된 텔타언어모델(201,202,203) 및/또는 누적언어모델(300)에 병합될 수 없다. 대신, 수신된 디바이스 텔타언어모델(111,112,113)은 메카니즘(500)에 의해 저장될 수 있고 최종발생한 언어모델이 장래 동기화 동안 디바이스에 전송될 필요가 있을 때에만 텔타언어모델(201,202,203) 및/또는 누적언어모델(300)에 병합될 수 있다.
- [0149] 텍스트 대신 언어모델을 전송하는 가능한 이점은 데이터 전송 및 저장 모두에 있어 난독화(obfuscation)를 통해 증가된 프라이버시와 보안이다. 소정 시나리오에서, 가령, 사용자 입력 텍스트량이 크거나 반복되는 워드들이 많을 경우, 저장 및 전송 데이터 크기는 텍스트 대신 언어모델을 이용함으로써 감소될 수 있다. 그러나, 사용자 입력 텍스트량이 작거나 반복되는 워드들이 거의 없을 경우, 저장 및 전송 데이터 크기는 언어모델 대신에 텍스트를 이용함으로써 감소될 수 있다는 점에서, 반대도 또한 마찬가지이다. 언어모델을 전송하는 가능한 다른 이점은 아래에 더 상세히 설명된 바와 같이 메카니즘에 의해서라기보다는 디바이스에 초기 프로세스가 수행되게 할 수 있다는 것이다.
- [0150] 상술한 어느 한 실시예에서, 각 디바이스는 동기화가 요망되거나 필요하면 메카니즘(500)과 동기화를 요청한다. 동기화를 요청하기 위한 트리거가 디바이스의 동기화를 부추기도록 결정된 사용자에게 있을 수 있다. 대안으로 또는 추가로, 각 디바이스는 주기적으로 동기화를 요청하도록 구성될 수 있어, 최소 수준의 업데이트가 발생하는 것을 보장하게 한다. 디바이스가 업데이트하는 빈도는 같을 필요가 없다. 일 실시예에서, 디바이스는 새 텍스트가 디바이스에 추가될 때마다 동기화를 트리거하도록 구성될 수 있다.
- [0151] 상술한 바람직한 실시예에서, 디바이스가 동기화를 제어한다. 다른 실시예에서, 메카니즘이 동기화를 트리거하도록 구성될 수 있고, 메카니즘은 순차적으로 복수의 디바이스들을 동기화시킨다.
- [0152] 본 발명의 바람직한 실시예에서, 동기화를 위한 메카니즘은 서버(1500)이다. 웹서버/클라우드가 복수의 디바이스들과 서버(1500) 간에 쉽게 액세스가능한 동기화를 제공할 수 있다. 도 5는 본 발명에 따른 시스템을 도시한 것으로, 시스템(2000)은 서버(1500)와 동기화될 2개의 디바이스(1100,1200)(태블릿(1100) 및 전화(1200))를 포함한다. 따라서, 디바이스의 초기 동기화 동안, 디바이스는 서버(1500)로부터 누적언어모델을 다운로드하고 서버에 텍스트를 업로드한다. 디바이스의 연속 동기화 동안, 마지막 동기화 이후로 디바이스(1100,1200)에 입력된 텍스트는 서버로 업로드되고 누적 및 텔타언어모델을 훈련하기 위해 사용된다. 도 4a-4c와 관련해 기술된 다른 실시예에서, 디바이스는 마지막 동기화 이후로 디바이스(1100,1200)에 입력된 텍스트에 대해 훈련된 디바이스 텔타언어모델을 동기화 동안 서버에 업로드시키고, 이 디바이스 텔타언어모델은 누적언어모델과 텔타언어모델에 병합된다. 각 디바이스(1100,1200)는 또한 마지막 동기화 이후 다른 모든 디바이스들에 입력된 텍스트를 기술한 특정 텔타언어모델을 서버(1500)로부터 다운로드한다.
- [0153] 사용자가 디바이스에 입력한 텍스트는 민감한 정보를 나타내며, 정보의 보안이 손상되지 않아야 한다. 본 시스템은 복수의 디바이스들과 메카니즘 간에 로컬 저장 또는 전송 동안 텍스트의 보안이 손상되지 않는 것을 보장하기 위한 많은 방안들을 제공한다.

- [0154] 텍스트가 디바이스로부터 메카니즘으로 전송되는 실시예에서, 사용자가 입력한 바와 같이 실제 텍스트는 바람직하게는 이 텍스트의 보안이 전송동안 손상되지 않는 것을 보장하기 위해 보안 연결을 통해, 가령, 암호화 통신 채널을 통해 보내진다. 마찬가지로, 메카니즘으로부터 복수의 디바이스들 각각으로 델타 및 누적언어모델을 전송하기 위해 메카니즘과 디바이스 간에 보안연결이 확립될 수 있다.
- [0155] 본 발명의 시스템(1000,2000)의 고유 속성은 메카니즘(500,1500)이 실제 텍스트를 포함하지 않는 누적언어모델 또는 델타언어모델의 형태로 업데이트를 저장 및 전송한다는 것이다. 따라서, 보안연결이 손상되더라도, (사용자 언어 스타일의 확률적 인캡슐레이션인) 언어모델만이 노출되지 텍스트 그 자체는 아니다. 그러므로, 바람직하지만, 델타(디바이스 델타언어모델 및 메카니즘의 델타언어모델) 및 누적언어모델을 전송하는데 보안연결이 반드시 필요하지 않다.
- [0156] 디바이스와 서버 모두의 관점에서, 웹기반의 서버를 포함한 시스템에서 복수의 다이나믹 언어모델의 동기화를 논의한다.
- [0157] 도 6 및 도 7은 디바이스와 서버의 관점에서 초기 동기화를 도시한 것이고, 도 8 및 도 9는 디바이스와 서버의 관점에서 연속 동기화를 도시한 것이다.
- [0158] 도 6은 디바이스의 관점에서 초기 동기화를 도시한 것이다. 디바이스와 서버 간에 동기화를 요청하고 보안연결을 구축한 후, 사용자는 바람직하게는 사용자에게 한정된 다이나믹 언어모델을 동기화하도록 (가령, 패스워드 요청과 제공에 의해) 인증한다.
- [0159] 그런 후 디바이스는 서버로부터 누적언어모델을 다운로드하고 누적언어모델을 기존 다이나믹 언어모델에 병합한다. 텍스트가 동기화 전에 디바이스에 입력되면, 디바이스는 누적언어모델을 다운로드한 후 상기 디바이스에 입력되었던 텍스트를 나타내는 텍스트 데이터를 업로드하도록 구성된다. 디바이스에 입력된 텍스트를 나타내는 데이터는 도 3a, 3b, 및 3c를 참조로 상술한 바와 같이, 그리고 도 6에 도시된 바와 같이 실제 텍스트를 포함할 수 있거나, 도 4a, 4b, 및 4c를 참조로 기술된 바와 같이 디바이스 델타언어모델을 포함할 수 있다.
- [0160] 서버 관점에서 초기 동기화가 도 7에 도시되어 있다. 초기 동기화는 디바이스와 연속 동기화를 위해 서버를 준비하는 단계와 서버와 이전에 동기화된 모든 디바이스들에 대해 여태까지 입력되었던 모든 텍스트를 나타내는 누적언어모델을 디바이스에 제공하는 단계로 구성된다.
- [0161] 서버는 사용자 및 디바이스를 인증하고, 해당 사용자에게 대해 먼저 동기화가 있으며, 서버는 빈 누적언어모델을 생성한다. 도 3a 및 3b를 참조로 상기에서 상세히 기술된 바와 같이, 이 빈 누적언어모델은 추후 디바이스에 입력된 모든 텍스트에 대해 훈련된다. 도 4a 및 4b를 참조로 기술된 다른 실시예에서, 디바이스에 입력된 모든 텍스트를 나타내는 디바이스 델타언어모델이 누적언어모델에 병합된다.
- [0162] 디바이스에 대한 초기 동기화 동안, 서버는 또한 디바이스에 대한 빈 델타언어모델을 생성하도록 구성되며, 상기 델타언어모델은 도 3a 및 3b를 참조로 기술된 바와 같이, 다른 디바이스들의 이전 동기화 동안 서버에 의해 수신되고 다른 디바이스들에 입력된 텍스트로 훈련된다. 다른 실시예에서, 도 4a 및 4b를 참조로 상세히 기술된 바와 같이, 다른 디바이스들에 입력된 모든 텍스트를 나타내는 디바이스 델타언어모델들이 델타언어모델에 병합된다.
- [0163] 그런 후 서버는 디바이스에 누적언어모델을 제공한다. 사용자용의 제 1 동기화가 있다면, 서버는 도 3a 또는 도 4a의 디바이스(A)에 대해 상술한 바와 같이 디바이스에 빈 누적언어모델을 제공한다. 대안으로, 다른 디바이스들의 동기화가 이미 발생했다면, 서버는 다른 디바이스들의 동기화 동안 수신된 텍스트 데이터에 대해 훈련된 누적언어모델을 디바이스에 제공한다.
- [0164] 그 후 서버는 (있다면) 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 업로드한다. 누적언어모델이 디바이스에 제공된 후 텍스트 데이터가 업로드되어, 상기 업로드된 텍스트가 디바이스에 제공된 누적언어모델에 포함되지 않는 것을 보장한다.
- [0165] 디바이스 관점에서 (동기화 동안 디바이스를 텍스트에 전송하는 실시예에서) 연속 동기화가 도 8에 도시되어 있다. 초기 동기화의 경우와 같이, 디바이스는 먼저 서버와 보안연결을 확립해야 하고 사용자는 인증을 해야 한다. 서버는 또한 디바이스를 인증하고, (특정 델타언어모델은 서버가 디바이스들과 이들 디바이스들에 입력된 텍스트를 나타내는 텍스트 데이터를 식별하는 것을 필요하며 각 디바이스와 관련되어 있기 때문에) 델타언어모델을 발생하도록 디바이스들이 서로 식별되게 한다. 디바이스는 서버로부터 키에 대한 요청에 응답해 상기 디바이스에 고유한 키를 제공함으로써 서버와 인증할 수 있다.



- [0166] 도 8에 도시된 바와 같이, 각 디바이스는 마지막 동기화 이벤트 이후로 디바이스에 입력된 임의의 텍스트를 서버에 업로드한다. 바람직하기로, 디바이스는 텍스트를 서버에 업로드한 후 텍스트 버퍼를 삭제하도록 구성된다. 대안적인 실시예로, 각 디바이스는 디바이스 델타언어모델을 서버에 업로드하고 업로드된 후에 이 디바이스 델타언어모델을 삭제한다.
- [0167] 디바이스는 또한 서버로부터 특정한 델타언어모델을 다운로드하고 델타언어모델을 다이나믹 언어모델에 병합하도록 구성된다. 따라서, 연속 동기화에서, 각 디바이스는 다른 디바이스들의 이전 동기화 동안 다른 디바이스들로부터 서버에 의해 수신된 입력 텍스트를 나타내는 텍스트 데이터로부터 발생된 (특정 델타언어모델 형태의) 통계 언어데이터를 학습한다.
- [0168] 도 8에 도시된 바와 같이, 디바이스에 한정된 델타언어모델은 다른 디바이스들 모두에 입력된 텍스트를 나타내는 텍스트 데이터로부터 발생되기 때문에, 텍스트 데이터를 업로드하고 델타언어모델을 다운로드하는 2개의 과제들은 완전히 독립적이다. 따라서, 이들 2개의 과제들은 나란히 실행될 수 있다.
- [0169] 서버 관점에서 디바이스의 연속 동기화가 도 9에 도시되어 있다. 동기화를 위한 요청을 수신하고, 보안연결을 확립하며, 사용자와 디바이스를 인증한 후, 서버는 2개의 별개의 과제들을, 바람직하게는 나란히 수행하도록 구성된다. 한가지 과제로, 마지막 동기화 이후 디바이스에 입력된 텍스트를 나타내는 텍스트 데이터를 검색한다. 데이터가 실제 입력 텍스트를 포함하면, 도 9에 도시된 바와 같이, 이 텍스트는 시스템에 있는 다른 디바이스들과 관련된 델타언어모델을 훈련시키는데 이용된다. 바람직하기로, 이 텍스트는 또한 모든 디바이스들에 입력된 모든 텍스트에 대해 훈련된 누적언어모델을 훈련하는데도 이용된다. 업데이트가 발생함에 따라 모든 업데이트들로 누적언어모델을 훈련시킴으로써, 서버는 더 신속하고 효율적으로 임의의 추가 디바이스를 사전 동기화된 디바이스들과 동기화할 수 있다. 서버는 바람직하게는 델타언어모델과 누적언어모델을 나란히 훈련하도록 구성된다. 제 2 과제로, 서버는 디바이스에 한정된 델타언어모델을 상기 디바이스에 제공한다. 바람직하기로, 서버는 일단 델타언어모델이 디바이스에 제공된 후 상기 델타언어모델을 삭제한다.
- [0170] 대안으로, 데이터가 디바이스 델타언어모델을 포함한 실시예에서, 디바이스 델타언어모델은 시스템에 있는 다른 디바이스들과 관련된 델타언어모델에 병합된다. 바람직하기로, 델타언어모델은 또한 모든 디바이스에 입력된 모든 텍스트를 나타내는 누적언어모델에 병합된다. 각 동기화가 발생할 때 디바이스 델타언어모델을 누적언어모델에 병합함으로써, 서버는 임의의 추가 디바이스를 사전 동기화된 디바이스와 더 신속히 효율적으로 동기화할 수 있다. 서버는 바람직하게는 디바이스 델타언어모델을 델타언어모델과 누적언어모델에 나란히 병합시키도록 구성된다. 제 2 과제로, 서버는 디바이스에 한정된 델타언어모델을 상기 디바이스에 제공한다. 바람직하기로, 서버는 일단 델타언어모델이 디바이스에 제공된 후 상기 델타언어모델을 삭제한다.
- [0171] 동기화의 문제들 중 하나는 반복된 병합의 부작용으로 부자연스럽게 어떤 통계언어 데이터의 강도를 높임으로써 다이나믹 언어모델이 바이어스되는 경향이 있다는 것이다. 본 발명의 메카니즘은 디바이스의 동기화 동안, 다이나믹 언어모델이 자신의 다이나믹 언어모델내에 이미 있는 지금까지 통계언어 데이터를 재학습하지 않고도 시스템의 다른 디바이스에 입력된 용어 및 일련의 용어들을 학습하는 것을 보장함으로써 이 문제를 해결한다. 이를 달성하기 위해, 디바이스의 초기 동기화 동안, 메카니즘은 상기 디바이스로부터의 입력 텍스트를 나타내는 임의의 텍스트 데이터를 업로드하기 전에 누적언어모델을 디바이스에 전송하여, 디바이스가 상기 디바이스에 이미 입력된 텍스트를 나타내는 텍스트 데이터를 수신하지 않는 것을 보장한다. 마찬가지로, 연속 동기화 동안, 메카니즘은 델타언어모델이 의도된 디바이스를 제외하고 모든 디바이스들에 입력된 텍스트를 나타내는 텍스트 데이터를 이용해 상기 디바이스에 한정된 델타언어모델을 발생한다. 따라서, 본 발명의 동기화 메카니즘은 다이나믹 언어모델이 상기 다이나믹 언어모델에 이미 있는 통계 데이터를 수신하고 병합하지 않는 것을 보장한다.
- [0172] 다이나믹 언어모델(101,102,103)의 바이어싱이 문제가 되지 않는다면, 제 1 실시예에서, 메카니즘(500)은 초기화 및 연속 동기화 동안 모든 디바이스들로부터 수신된 텍스트( $T_A, T_B, T_C$ )에 대해 누적언어모델을 훈련시키고, 초기화 및 연속 동기화 동안 각 디바이스에 누적언어모델(300)을 제공하도록 구성될 수 있다. 즉, 메카니즘은 전혀 델타언어모델을 포함하지 않으며, 디바이스들은 개별적으로 또는 집합적으로 동기화된다. 마찬가지로, 다른 실시예에서, 메카니즘(500)은 초기화 및 연속 동기화 동안 모든 디바이스들로부터 수신된 디바이스 델타언어모델을 누적언어모델(300)에 병합시키고, 초기화 및 연속 동기화 동안 누적언어모델(300)을 각 디바이스에 제공하도록 구성될 수 있다. 즉, 메카니즘은 전혀 델타언어모델을 포함하지 않으며, 디바이스들은 개별적으로 또는 집합적으로 동기화된다. 또 다른 실시예로, 메카니즘은 디바이스에 대한 델타언어모델을 다운로드함으로써 각 디바이스에 대해 개별적으로 연속 동기화가 실행되게 하고, 이로써 디바이스가 초기 동기화를 수행할 때, 가령, 새 디바이스가 시스템과 동기화될 때, 시점(occasion)에 대한 바이어스를 제한하여 디바이스의 집합적 초기 동

기화를 실행하도록 구성될 수 있다.

- [0173] 상기 설명은 메카니즘이 복수의 델타언어모델들에 누적언어모델을 발생하는 본 발명의 바람직한 실시예들을 상세히 말하고 있다. 그러나, 또 다른 실시예에서, 메카니즘은 복수의 델타언어모델들만을 발생할 수 있다. 이 실시예는 모든 디바이스들이 누적언어모델로부터라기 보다는 이들과 관련된 델타언어모델들을 통해 다른 디바이스들로부터 초기 텍스트를 학습하는 점에서 도 3a, 3b, 4a 및 4b에 대해 상술한 바람직한 실시예와는 다르다. 사전 동기화된 시스템에 도입된 임의의 새 디바이스는 이들 디바이스들에 입력된 임의의 다른 텍스트 이외에 이전 동기화 이후로 사전 동기화된 디바이스들에 입력된 텍스트로부터만 학습할 것이다. 이는 사용자가 입력한 가장 최신의 텍스트가 학습을 위해 상기 디바이스에 대한 가장 관련있는 텍스트일 수 있기 때문에 허용될 수 있다. 더욱이, 이런 시스템에서 통계언어모델의 바이어싱이 전혀 없을 것이다. 그러나, 이는 시스템의 모든 통계언어 데이터를 동화시키지 않기 때문에, 새 디바이스가 사용자의 쓰기 스타일을 학습하는데 시간이 더 걸릴 것임을 의미한다.
- [0174] 상기 설명은 디바이스들이 하나의 다이내믹 언어모델을 포함하는 본 발명의 실시예를 상세히 기술한다. 그러나, 다른 실시예에서, 하나 이상의 디바이스들(몇몇 실시예에서 모든 디바이스들)은 다수의 다이내믹 언어모델들을 이용할 수 있다. 이를 용이하게 하기 위해, 각 다이내믹 언어모델은 고유하게 태그된다. 즉, 디바이스내 각 다이내믹 언어모델은 디바이스내서 이를 식별하도록 태그된다. 일예는 다음과 같다. 디바이스는 이메일 텍스트에 대해 그리고 또 다른 트위터 텍스트에 대해 다이내믹 언어모델을 갖는다. 이들 언어모델들 각각은 서로서로 식별하도록 태그될 수 있다. 태그는 "이메일", "트위터", "페이스북" 등과 같이 텍스트열과 같은 시작부의 형태를 취할 수 있다. 태그는 임의의 적절한 방식으로 디바이스에 저장될 수 있다.
- [0175] 적어도 하나의 디바이스가 다수의 다이내믹 언어모델을 이용하는 경우, 메카니즘은 다른 디바이스들로부터 다이내믹 언어모델을 동일한 태그로 동기화시킨다. 즉, 디바이스(X)가 앞서 지문에서 언급한 바와 같이 이메일 다이내믹 언어모델 및 트위터 다이내믹 언어모델을 갖고, 디바이스(Y)가 이메일 다이내믹 언어모델을 가지면, 양 디바이스에서 이메일 다이내믹 언어모델은 동일한 태그로 연관될 수 있다. 이들 태그를 인식하는 메카니즘은 동일한 태그를 갖는 다이내믹 언어모델을 동기화할 것이다.
- [0176] 이 실시예에서, 메카니즘과 관련된 디스패처 모듈을 또한 제공할 것이며, 메카니즘은 하나가 각 태그와 관련된 많은 인스턴스들을 포함하거나, 많은 별도의 메카니즘들의 형태를 취할 것이다. 관련된 다이내믹 언어모델로부터 다른 데이터와 함께 각 태그는 동기화 동안 각 디바이스에 의해 (메카니즘에 데이터의 제공에 대해 상술한 방식으로) 디스패처 모듈에 제공될 것이다. 이런 데이터의 수신시, 디스패처 모듈은 데이터 내에 있는 태그를 식별하고 분석한다.
- [0177] 태그는 또한 바람직하게는 맵에 또는 디스패처 모듈과 관련해 저장될 것이다. 이 시나리오에서, 태그는 맵의 핵심이다. 따라서, 태그를 식별할 때마다, 디스패처 모듈은 식별된 태그가 맵에 있는지 여부를 확실히 하는 형태를 취하는 분석을 수행한다. 그렇지 않으면, 메카니즘 또는 새 메카니즘의 새 인스턴스가 디스패처 모듈에 의해 생성되고 맵에서 해당 태그와 연관된다. 태그와 함께 수신된 데이터는 그런 후 이전 실시예에서 기술된 바와 같이 처리를 위해 새 메카니즘 인스턴스 또는 새 메카니즘으로 보내진다. 태그가 맵에 있다면, 디스패처 모듈은 태그와 관련된 데이터를 메카니즘 인스턴스 또는 태그와 관련된 메카니즘으로 보내며, 맵으로부터 식별된다. 데이터는 이전 실시예에서 상술한 바와 같이 메카니즘 인스턴스 또는 메카니즘에 의해 처리된다.
- [0178] 이 실시예에 대해 상기 주어진 예에서, 동기화가 누적언어모델(이메일), 디바이스(X)에 대한 델타언어모델(이메일) 및 디바이스(Y)에 대한 델타언어모델(이메일)에 있게 된다. 디바이스(Y)는 트위터 다이내믹 언어모델을 갖지 않으므로 트위터 다이내믹 언어모델에 대해 동기화하지 않는다. 따라서, 동기화가 디바이스(X)에 대해 누적언어모델(트위터) 및 델타언어모델(트위터)에 있게 된다. 쉽게 인식될 수 있는 바와 같이, 따라서, 이 예에는 이메일 메카니즘 인스턴스와 트위터 메카니즘 인스턴스(또는 별개의 이메일 및 트위터 메카니즘)가 있다.
- [0179] 디바이스(Y)가 또한 트위터 다이내믹 언어모델을 가지면, 동기화는 이메일 시나리오에 언급된 바와 같이 전개될 것이므로 디바이스(X)에 대한 상기 누적언어모델(트위터) 및 델타언어모델(트위터), 디바이스(Y)에 대한 델타언어모델(트위터)이 추가로 있게 된다. 이는 디바이스 내에 있을 수 있는 모든 가능한 다이내믹 언어모델들이 다른 디바이스들에 있는 이들의 등가물과 동기화될 수 있도록 확장된다. 쉽게 알 수 있는 바와 같이, 본 발명에 따르면 상술한 바와 같이 동기화는 발생한 디바이스를 이용해 언어모델들과 추가 태그를 식별하고, 다른 디바이스들로부터 하지만 상술한 방식으로 동일한 태그를 갖는 다이내믹 언어모델을 동기화함으로써 디바이스 시나리오 당 다수의 다이내믹 언어모델로 확장된다.

- [0180] 본 발명의 동기화 프로세스는 성능 문제를 막기 위해 온-디바이스 계산 면에서 가볍고 효율적이다. 이는 (다른 운영시스템 또는 소프트웨어 버전을 실행하는) 가능한 아주 많은 다양한 디바이스들이 다른 시간 간격으로 동기화하게 해 매우 확장적이다. 상술한 바와 같이, 동기화 빈도는 디바이스마다 다를 수 있다. 가령, 디바이스(A)는 디바이스(B 또는 C)보다 시스템(1000)과 더 정규적으로 동기화될 수 있다.
- [0181] 앞선 설명은 사용자에게 의한 디바이스의 사용 후 및/또는 사용 동안 복수의 디바이스들의 동기화를 설명한다. 그러나, 사용자가 많은 텍스트를 동기화 시스템의 디바이스들에 입력할 기회를 갖기 전에, 디바이스가 발생한 텍스트 예상은 사용자에게 대해 아주 정확하지 않을 수 있는데, 이는 다이나믹 언어모델이 사용자의 쓰기 스타일을 학습하지 못했기 때문이다.
- [0182] 사용자에게 대한 텍스트 예상 정확도를 높이기 위해, 동기화 시스템의 디바이스들의 다이나믹 언어모델은, 바람직하게는, 동기화 시스템의 디바이스들 밖에 있는 임의의 디바이스에 사용자가 입력했던 임의의 텍스트, 가령, 사용자의 PC에 저장된 이메일로 훈련될 수 있으며, 여기서 PC는 시스템의 디바이스들 중 하나가 아니다. 이 학습 프로세스를 '초기화'라 한다.
- [0183] 동기화 시스템의 외부에 있는 외부 메카니즘은 사용자가 동기화 시스템의 외부에 있는 디바이스들에 입력한 텍스트에 대해 하나 이상의 언어모델들을 훈련시킴으로써 하나 이상의 초기화 언어모델을 발생하도록 구성된다.
- [0184] 초기화 프로세스가 도 10에 도시되어 있다. 초기화 동안, 동기화 메카니즘/서버는 초기화 소스, 즉, 초기화 언어모델(들)을 생성한 외부 메카니즘과 보안 연결을 구축하도록 구성된다. 서버는 사용자와 초기화 소스를 인증하고, 인증된 후, 서버는 초기화 소스로부터 하나 이상의 초기화 언어모델을 다운로드한다.
- [0185] 그런 후 서버는 초기화 언어모델(들)을 누적언어모델에 병합하도록 구성된다. 직접적인 결과는 동기화 디바이스는 명백한 초기화를 수행하도록 요구되지 않으나 초기 동기화 동안 다이나믹 언어모델에 병합된 누적언어모델로부터 물려받을 것이라는 점이다.
- [0186] 초기화 프로세스가 수행되면, 서버는 가령, 도 3a 및 4a에서 디바이스(A)의 초기 동기화 후, 시스템의 제 1 디바이스가 동기화될 때 초기화 프로세스를 수행하는 것이 바람직하다. 바람직한 초기화 프로세스가 도 10의 연속선으로 도시되어 있다.
- [0187] 그러나, 초기화 프로세스는 대신 상술한 동기화 프로세스내 임의의 시기에 수행될 수 있다. 시스템의 제 1 동기화 후 (가령, 디바이스(A)의 초기 동기화 후) 초기화가 수행되면, 초기화 언어모델은 서버에 의해 누적언어모델 및 모든 델타언어모델에 병합될 것이다. 그 결과, 디바이스가 델타언어모델을 다이나믹 언어모델에 다운로드하고 병합할 경우, 시스템의 디바이스들은 디바이스의 연속 동기화 동안 초기화 데이터를 수신할 것이다. 추후에 메카니즘/서버와 동기화되는 임의의 디바이스(가령, 상술한 바와 같이 디바이스(D))는, 디바이스가 누적언어모델을 다운로드하고 이를 다이나믹 언어모델에 병합시키기 때문에, 디바이스의 초기 동기화 동안 초기화 데이터를 수신할 것이다.
- [0188] 제 1 동기화 이후에 수행되는 초기화 프로세스가 도 10에 도시되고, 여기서 프로세스는 이제 또한 (초기화 언어모델을 델타언어모델에 병합시키는) 대시선으로 표시된 프로세스의 일부를 포함한다. 델타언어모델에 초기화 모델의 병합은 초기화 모델을 누적언어모델에 병합하는 것과 동시에 수행될 수 있다.
- [0189] 다른 실시예에서, 초기화는 메카니즘 또는 동기화 서버 대신 디바이스들 중 하나에서 발생할 수 있다. 이들 실시예에서, 초기화 소스와 연결되고 상술한 바와 같이 초기화 언어모델을 다운로드하는 디바이스가 있다. 초기화 언어모델이 다운로드된 후, 디바이스의 기존 다이나믹 언어모델과 병합되고, 초기화 언어모델에 포함된 텍스트 데이터는 앞서 상세히 기술된 바와 같이 메카니즘과 동기화를 위한 디바이스의 텍스트 데이터 포함된다. 대표적으로 디바이스 초기화로, 디바이스 상의 텍스트 데이터는 디바이스 델타 언어모델의 형태일 것이고, 초기화 언어모델을 텍스트 데이터에 포함하는 것은 초기화 언어모델을 디바이스 델타언어모델에 병합하는 것을 포함할 것이다.
- [0190] 디바이스 초기화는 바람직하게는 초기화 프로세스가 사용자로부터 인터랙션을 필요로 하는 상황(가령, 초기화 소스 선택 및 제 3 자 서비스 인증)에 바람직할 것이다. 초기화가 메카니즘에 의해 수행되었던, 메카니즘은 사용자와 외부 초기화 메카니즘 간에 프록시로서 행동해야 할 것이다. 따라서, 디바이스로부터 초기화를 수행하는 것은 동기화 서버에 대한 프록시 메카니즘을 실행할 필요가 없어 실행 복잡도를 줄인다. 또한, 디바이스 초기화로, 초기화 서비스를 위한 동기화 서비스에 대한 의존성이 없어, 동기화가 이용될 수 없거나 사용되지 않는 상황에서도 초기화가 여전히 가능하다.

- [0191] 반복된 병합으로 인한 어떠한 바이어스도 도입되지 않는 것을 보장하기 위해, 초기화 프로세스는 단지 한번만 수행된다.
- [0192] 본 발명의 2가지 다른 태양들은 에러 완화를 위한 메카니즘과 방법이다. 에러 완화는 델타언어모델이 메카니즘에서 디바이스로 전송될 때 야기된 문제를 해결하나, 에러로 인해 델타언어모델을 디바이스에 수신하거나 성공적으로 이용할 수 없게 된다. 이런 상황을 야기할 수 있는 에러의 예로는 네트워크 연결 문제 및 데이터 손상을 포함한다; 이들은 특히 모바일 환경에서 통상적인 문제이다. 이런 에러 시나리오에서, 메카니즘은 델타언어모델을 디바이스로 전송하고 이를 다시 빈 델타언어모델로 삭제하는 것으로 생각된다. 연속 동기화 동안, 디바이스는 이전에 수신되지 못한 델타언어모델의 콘텐츠를 포함하지 않는 델타언어모델을 수신한다. 따라서, 이 디바이스에 의해 제공된 예상은 동일한 사용자가 사용하는 다른 디바이스들에 의해 제공된 것들에서 벗어날 것이다.
- [0193] 하기에 상세히 기술된 바와 같은 에러완화 프로세스 없이, 이 문제를 해결하기 위한 유일한 방법은 메카니즘에서 디바이스로 그리고 디바이스에 대해 다이나믹 언어모델을 이 누적모델로 대체하기 위해 누적언어모델을 전송하는 것이다. 그러나, 이 에러상황을 감지하는 수단이 있을 필요가 있다. 게다가, 누적모델의 전송은 디바이스의 다이나믹 언어모델에 이미 포함된 데이터가 누적언어모델의 일부로서 메카니즘에서 디바이스로 전송될 것이므로 비효율적이게 된다.
- [0194] 에러완화를 이용하는 시스템(1000)의 실시예가 도 11에 도시되어 있다. 간략히 하기 위해, 하나의 디바이스만이 이 도면에 도시되어 있으나, 복수의 디바이스들이 동일한 메카니즘(500)으로 동기화될 수 있음이 이해될 것이다. 전과 같이, 디바이스(Z)는 (가령, 사용자가 입력한 실제 텍스트 또는 텍스트에 훈련된 언어모델을 포함할 수 있는) 다이나믹 언어모델(104)과 텍스트 데이터(114)를 포함한다. 디바이스(Z)는 또한 파라미터( $R_z$ )를 포함한다. 상술한 시스템에 대해, 메카니즘(500)은 바람직하게는 누적언어모델(300)을 포함한다. 메카니즘은 또한 1차 델타언어모델(204), 백업언어모델(214), 및 디바이스(Z)에 한정된 파라미터( $S_z$ )를 포함한다. 이 데이터 전송은 모든 동기화 동안 발생하므로,  $R_z$  및 텍스트 데이터의 데이터 전송은 실선으로 도시되어 있다. 1차 델타언어모델 또는 백업 델타언어모델이 (하기에 더 상세히 기술된 바와 같이) 동기화 동안 디바이스에 전송되므로, 이들 가능한 데이터 전송은 파선으로 도시되어 있다.
- [0195] 에러완화가 사용될 경우 디바이스(Z)의 동기화 동안 메카니즘에 의해 수행된 프로세스가 도 12에 도시되어 있다. 도 12에 도시된 단계들은 도 8에서 "델타(LM) 제공" 및 "델타 LM 삭제"에 해당하고 이를 대체한다. 도 11 및 도 12에 의해 도시된 에러완화 프로세스가 하기에 더 상세히 기술되어 있다.
- [0196] 도 11 및 도 12에 도시된 에러완화 프로세스의 실시예에서, 메카니즘은 메카니즘에서 디바이스로 전송된 데이터를 나타내는 각 디바이스와 관련된 파라미터(S)를 더 포함한다. 델타언어모델이 디바이스로 전송되면, 파라미터(S)는 델타언어모델과 관련된 식별자(I)와 함께 설정된다. 식별자(I)는 또한 델타언어모델을 갖는 디바이스로 전송된다. 식별자는 HTTP가, 가령, 통신 프로토콜이면 HTTP 파라미터 또는 고객 헤더로 전송될 수 있다.
- [0197] 디바이스가 에러없이 델타언어모델과 식별자를 수신하면, 디바이스는 식별자(I)와 같은 수신된 데이터를 나타내는 파라미터(R)를 설정한다. 데이터 전송시 에러가 있으면, 파라미터(R)는 업데이트되지 않고, 마지막 성공한 동기화의 식별자와 함께 남아 있다.
- [0198] 디바이스가 다음 동기화될 때, 디바이스는 파라미터(R)를 메카니즘으로 보낸다. 메카니즘은 수신된 파라미터(R)를 저장된 파라미터(S)와 비교한다. 마지막 동기화 동안 데이터 전송이 성공되었다면, 파라미터(R 및 S)는 동일하다. 이들이 같지 않다면, 마지막 동기화의 데이터 전송은 성공적이지 못하고 에러가 발생된다.
- [0199] 에러를 완화하기 위해, 메카니즘(500)은 각 디바이스와 관련해 하나는 1차 델타언어모델이고 하나는 백업 델타언어모델의 2개의 델타언어모델을 포함한다. 1차 델타언어모델은 가령 도 3a, 3b, 4a, 및 4b에 대해 상술된 델타언어모델과 같다. 1차 델타언어모델은 상기에서 상세히 기술된 바와 같이 관련된 디바이스의 마지막 동기화 이후 다른 모든 디바이스들들과의 동기화에 의해 제공된 텍스트 입력을 나타내는 데이터를 포함한다.
- [0200] 백업 델타언어모델은 에러완화 프로세스의 일환이다. 백업 델타언어모델은 디바이스가 에러 없이 델타언어모델을 수신한 것을 확인한 관련된 디바이스의 마지막 동기화 이후 다른 모든 디바이스들과 동기화에 의해 제공된 텍스트 입력을 나타내는 데이터를 포함한다. 그러므로, 다른 디바이스들이 동기화됨에 따라, 이들로부터 수신된 텍스트 데이터는 1차 델타언어모델과 백업 델타언어모델 모두에 포함된다. 델타언어모델의 수신 확인은 다음 동기화의 시작시 메카니즘에 파라미터(R)의 전송을 통해 발생한다. 따라서, 전송 에러가 전혀 없었다면, 백업 델타언어모델은 관련된 디바이스의 끝에서 두번째 동기화 이후 다른 디바이스들과의 동기화로부터 텍스트 데이



터를 포함할 것이다. 전송시 이전에 에러가 있었다면, 백업 델타언어모델은 관련된 디바이스의 앞선 동기화 이후 수집된 텍스트 데이터를 포함할 것이다.

[0201] 도 11로 돌아가, 메카니즘은 디바이스로부터 파라미터(R)를 수신한다. 파라미터(R)가 동기화의 시작시 파라미터(S)와 같다면, 마지막 동기화가 성공적이었고 1차 델타언어모델에 대한 식별자와 함께 1차 델타언어모델이 디바이스로 전송된다. 파라미터(S)는 델타언어모델과 함께 전송된 식별자에 설정된다. 이 경우, 백업 델타언어모델은 필요치 않다. 백업 델타언어모델은 삭제될 수 있고, 1차 델타언어모델이 백업 델타언어모델에 복사된다. 1차 델타언어모델은 그런 후 빈 델타언어모델이 다른 디바이스들의 장래의 동기화로부터 텍스트 데이터를 수신할 준비하게 삭제된다.

[0202] 동기화의 시작시 파라미터(S)가 파라미터(S)와 같지 않으면, 마지막 동기화가 성공되지 못한다. 이런 경우이면, 백업 델타언어모델이 디바이스에 전송된다. 백업 델타언어모델은 마지막에 성공하지 못한 전송으로 보내진 텍스트 데이터와 다른 디바이스들과 연속 동기화로 인한 텍스트 데이터를 포함한다. 그러므로, 1차 델타언어모델 대신 백업 델타언어모델을 디바이스로 전송함으로써, 디바이스는 전송시 에러로 인해 손실될 수 있는 텍스트 데이터를 수신한다. 파라미터(S)는 백업 델타언어모델의 식별자에 설정된다. 1차 델타언어모델은 빈 델타언어모델이 다른 디바이스들의 장래 동기화로부터 텍스트 데이터를 수신할 준비를 하게 삭제된다.

[0203] 에러가 전혀 검출되지 않고(즉, R이 S와 같고), 디바이스의 마지막 동기화 이후로 어떠한 새 텍스트 데이터도 다른 디바이스로부터 수신되지 않았다면(그리고 1차 델타언어모델이 비어 있다면), 어떠한 델타언어모델도 디바이스에 전송될 필요가 없다. 이 경우, 백업 델타언어모델이 삭제된다.

[0204] 몇몇 실시예에서, 메카니즘은 디바이스에 도달된 것이 확인된 데이터를 나타내는 각 디바이스와 관련된 장래의 파라미터(C)를 포함할 수 있다. 즉, 각 동기화 이후, C는 디바이스로부터 전송된 수신된 텍스트 파라미터(R)와 같게 설정된다. 에러가 전송동안 발생하면(따라서 R이 파라미터 S와 같지 않으면), 성공적인 전송이 그 사이에 전혀 나타나지 않았기 때문에 R이 업데이트되지 않을 것이므로, 다음 동기화 동안 메카니즘에 의해 수신된 파라미터(R)는 파라미터(C)와 같아야 한다. R이 S 또는 C와 같지 않으면, 예상치 못한 심각한 오류가 발생한 것이고, 디바이스는 다이나믹 언어모델을 삭제하고 이를 메카니즘에 저장된 누적언어모델로 대체하여 디바이스가 다른 디바이스들과 정확히 동기화되는 것을 보장해야 한다. 이는 (가령, 파라미터(R)를 한정하지 않음으로써) 디바이스가 누적모델을 요청할 수 있도록 디바이스로 에러신호를 보내는 메카니즘을 통해 혹은 대안으로 메카니즘을 소개함으로써 가능해질 수 있고, 이로써 누적모델이 델타모델 대신 디바이스에 다운로드되고 디바이스가 다이나믹 언어모델과 병합되기보다 교체되도록 신호 받는다.

[0205] 바람직하기로, 고유 식별자는 디바이스에서 메카니즘으로 전송된 각 텍스트 데이터의 세트와 관련되고, 델타언어모델과 관련된 식별자(I)는 텍스트 입력 히스토리를 제공하도록 언어모델내에 포함된 텍스트 데이터의 고유 식별자들로부터 발생된다(가령, 고유 식별자(I)는 텍스트 데이터 세트와 관련된 고유 식별자 스트링이다). 식별자(I)는 델타언어모델에 포함될 가장 최신의 텍스트 데이터와 관련된 고유 식별자와 같을 수 있다. 바람직하기로, 텍스트 데이터의 식별자는 메카니즘에 의해 발생된다. 대안으로, 텍스트 데이터의 식별자는 디바이스에 의해 발생될 수 있고 메카니즘으로 전송될 수 있다. 바람직한 실시예에서, 텍스트 데이터의 식별자는 타임스탬프로부터 발생된다. 대안으로, 텍스트 데이터의 식별자는 타임스탬프, 연속 정수, 난수, 또는 디바이스 식별자들 중 하나 이상으로부터 발생될 수 있다. 언어모델이 아닌(가령 "-") 또는 빈 언어모델(가령, "0")을 나타내는데 표준 식별자가 이용될 수 있다.

[0206] 고유 식별자를 디바이스로부터 전송된 각각의 텍스트 데이터 세트와 연관시키는 것은 (가령, 동일한 사용자에게 속하는 2개의 다른 사용자 계정들을 병합시킨다면) 다수의 델타언어모델들의 병합을 허용한다. 고유 식별자가 연속이면(가령, 타임스탬프를 이용하면), 이들은 사용자 입력 히스토리의 순서를 식별하는데 이용될 수 있고, 따라서 언어모델이 (가령, 더 최근의 데이터에 우선순위를 주며) 어떻게 병합되어야 하는지 판단할 수 있다.

[0207] 몇몇 실시예에서, 다른 디바이스들로부터 메카니즘에 의해 수신된 텍스트 데이터는 델타언어모델 대신에 저장될 수 있다. 필요한 델타언어모델이 동기화 동안 이 데이터로부터 발생될 수 있다. 예컨대, 텍스트 데이터가 디바이스 델타언어모델의 형태인 실시예에서, 메카니즘은 디바이스 델타언어모델을 저장할 수 있다. 메카니즘은 관련된 디바이스 델타언어모델을 병합시켜 동기화되고/되거나 동기화 동안 디바이스 델타언어모델들을 병합시킴으로써 누적 언어모델을 발생시킬 수 있는 디바이스에 한정된 디바이스 델타언어모델을 발생시킬 수 있다.

[0208] 몇몇 실시예에서, 백업 델타언어모델은 다른 디바이스들의 연속 동기화와 더불어 업데이트될 수 없다. 이 경우, 백업 델타언어모델은 관련된 디바이스의 마지막 동기화 이전에 다른 디바이스로부터 메카니즘으로 전송된 텍스트

트 데이터만을 포함한다. 1차 델타언어모델은 통상 관련된 디바이스의 마지막 동기화 이후 다른 디바이스들로부터 메카니즘으로 전송된 텍스트 데이터를 포함한다. 따라서, 전송 에러가 있고, 백업 델타언어모델이 필요하다면, 1차 및 백업 델타언어모델 모두로부터의 데이터가 디바이스에 전송되어야 한다. 이는 디바이스에 전송하기 전에 2개의 델타언어모델을 병합시킴으로써 달성될 수 있으며, 2개 델타언어모델들이 별개로 전송될 수 있다.

[0209] 바람직한 실시예에서, 디바이스의 동기화는 디바이스에 의해 초기화되고 요청-응답 기반으로 진행된다. 이 경우, 디바이스로부터 메카니즘으로 텍스트 데이터의 전송에 에러가 있는 경우 반드시 에러 완화를 제공할 필요가 없을 수 있는데, 이는 전송시 에러는 동기화 동안 메카니즘으로부터 응답(또는 응답 결여)에 의해 디바이스에 통신될 수 있기 때문이다. 디바이스가 (텍스트 데이터를 포함한) 요청과 동기화를 개시할 수 있고, 델타언어모델은 서버에 의해 응답시 디바이스로 반환된다. 따라서, 요청이 성공적이었다면(그리고 텍스트 데이터가 에러 없이 전송되면), 메카니즘이 응답할 것이다. 그런 후 메카니즘이 이를 안전하게 수신했음이 알려지기 때문에, 디바이스는 텍스트 데이터를 삭제할 수 있다. 메카니즘이 예상한 바와 같이 응답하지 않으면, 디바이스는 새 동기화 요청으로 다시 텍스트 데이터를 전송할 수 있다.

[0210] 몇몇 실시예에서, 디바이스는 데이터의 수신 후 즉시 델타언어모델의 수신 확인을 전송할 수 있다. 이 경우, 메카니즘은 수신이 확인되면 상기 디바이스에 한정된 델타언어모델을 삭제할 것이다. 수신이 확인되지 않으면, 델타언어모델은 삭제되지 않을 것이다. 이 수신 확인은 상술한 바와 같이 1차 및 백업 델타언어모델을 포함한 에러 완화 프로세스에 대한 필요성을 없앨 것이다.

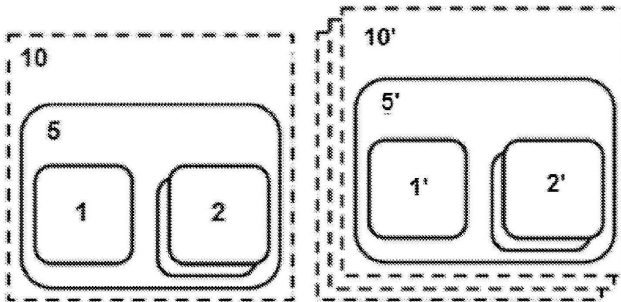
[0211] 그러나, 몇몇 실시예에서, 디바이스로부터 메카니즘으로 텍스트 데이터가 전송을 위한 에러 완화 프로세스를 갖는 것이 이점적일 수 있다. 이는 동기화가 서버에 의해 개시되고 요청-응답 기반으로 진행되는 경우이거나, 비동기화 통신이 있는 경우일 수 있다. 이 경우, 상술한 바와 실질적으로 유사한 에러 완화 프로세스가 실행될 수 있다. 이 경우, 디바이스는 텍스트 데이터의 1차 세트와 텍스트 데이터의 백업 세트의 2개의 텍스트 데이터 세트를 저장할 것이다. 각 동기화 시에, 전송된 텍스트 데이터와 관련된 식별자가 디바이스에 의해 저장될 수 있고, 텍스트 데이터와 함께 메카니즘으로 전송될 수 있다. 메카니즘이 다음 동기화시 매칭 식별자를 반환하면, (마지막 동기화 이후 수집된 텍스트 데이터를 포함하는) 1차 텍스트 데이터가 메카니즘으로 전송된다. 메카니즘으로부터 수신된 식별자가 저장된 식별자와 일치하지 않으면, (메카니즘에 의해 수신된 것으로 확인된 마지막 전송 이후 수집된 모든 텍스트 데이터를 포함한) 백업 텍스트 데이터가 메카니즘으로 전송된다. 이런 에러 완화 시스템은 델타언어모델에 대한 상술한 에러 완화 시스템의 상세 내용들 중 어느 하나를 필요한 부분만 약간 수정해 포함할 수 있다.

[0212] 상술한 바로부터 명백한 바와 같이, 본 발명은 복수의 다이나믹 언어모델을 동기화하기 위한 방법을 제공한다. 방법은 동기화를 위한 메카니즘에서 텍스트 데이터를 수신하는 단계; 상기 메카니즘으로 텍스트 데이터를 적어도 하나의 언어모델에 포함시키는 단계; 및 메카니즘으로 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하는 단계를 포함한다. 한가지 방법은 동기화를 위한 메카니즘에서 복수의 디바이스들에 사용자가 입력한 텍스트를 수신하는 단계를 포함하고, 각 디바이스는 다이나믹 언어모델을 포함한다. 상기 방법은 메카니즘으로 텍스트에 대한 적어도 하나의 언어모델을 훈련시키는 단계, 및 메카니즘으로 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하는 단계를 더 포함한다. 또 다른 방법은 동기화를 위한 메카니즘에서 디바이스에 사용자가 입력한 텍스트에 대해 훈련된 디바이스 델타언어모델을 수신하는 단계를 포함한다. 이 방법은 상기 메카니즘으로 디바이스 델타언어모델을 적어도 하나의 언어모델과 병합시키는 단계; 및 메카니즘으로 디바이스를 동기화하기 위한 적어도 하나의 언어모델을 제공하는 단계를 더 포함한다. 상기 방법의 다른 태양들은 도 3-9의 상기 설명으로부터 쉽게 명백해질 것이다.

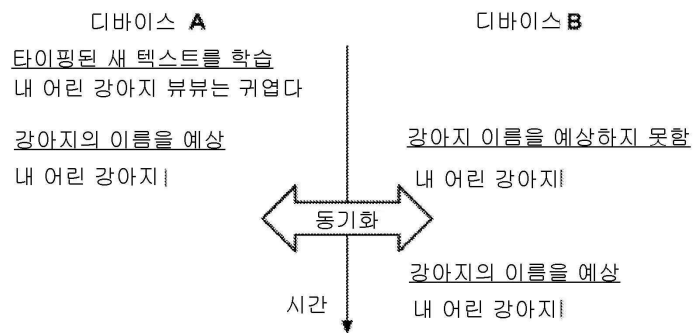
[0213] 이 설명은 단지 예인 것이 이해될 것이다; 특허청구범위에 정의된 바와 같이 본 발명의 범위로부터 벗어남이 없이 설명된 실시예들에 대한 변형 및 변경들이 행해질 수 있다.

도면

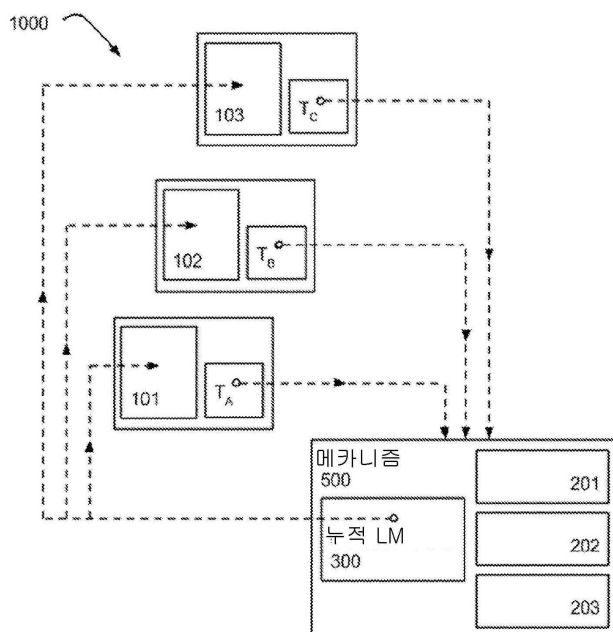
도면1



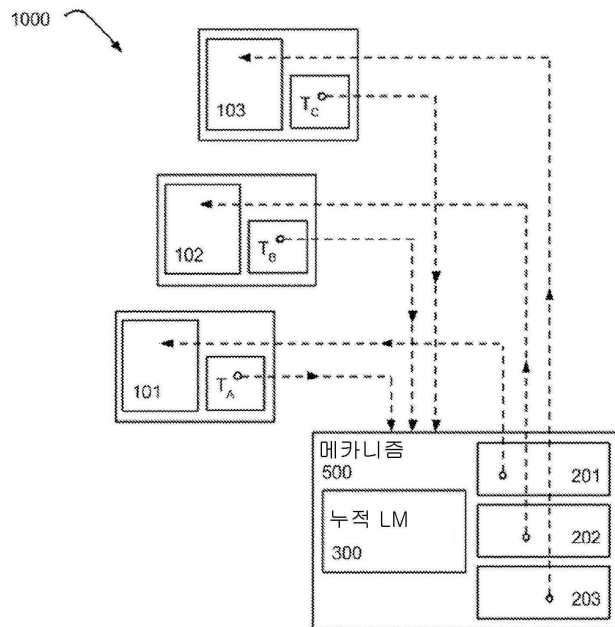
도면2



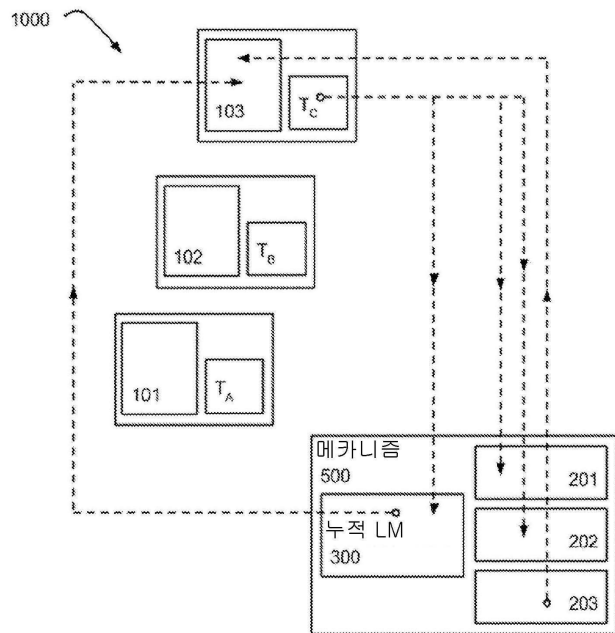
도면3a



도면3b

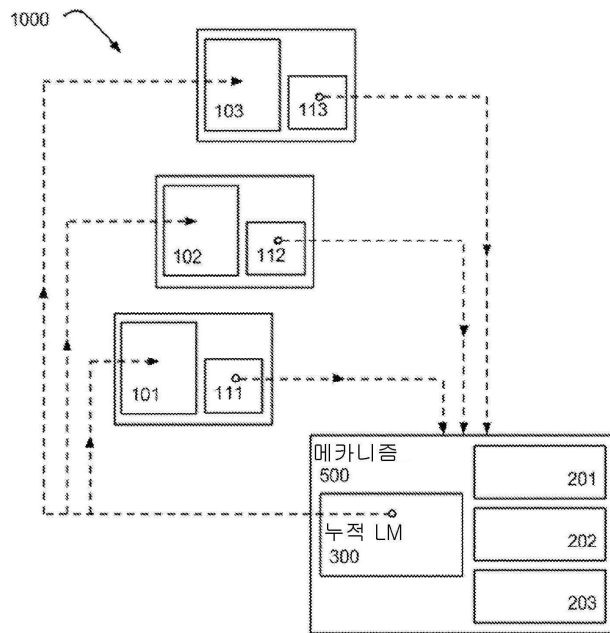


도면3c

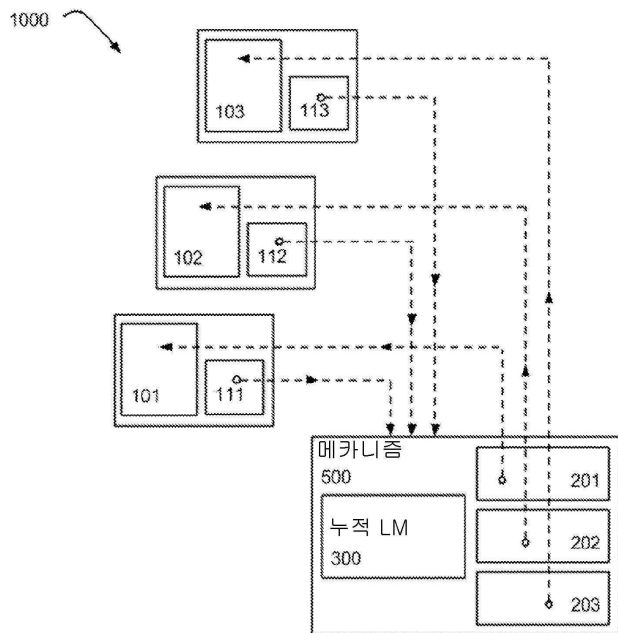




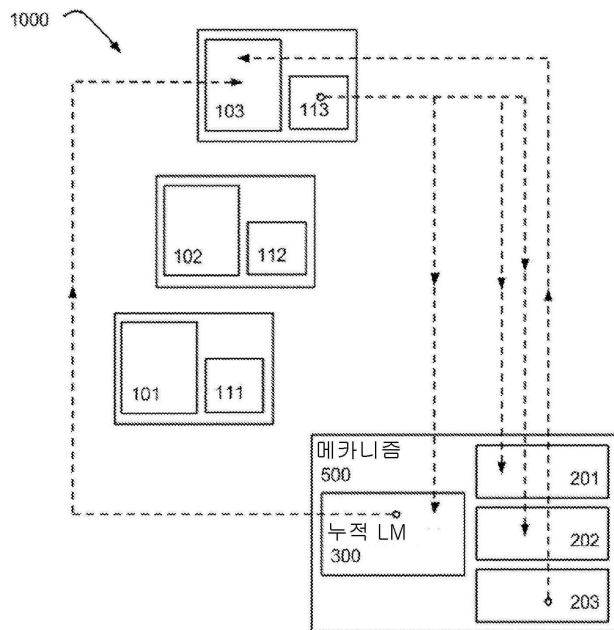
도면4a



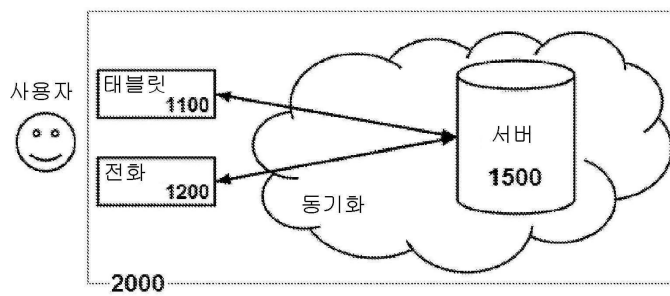
도면4b



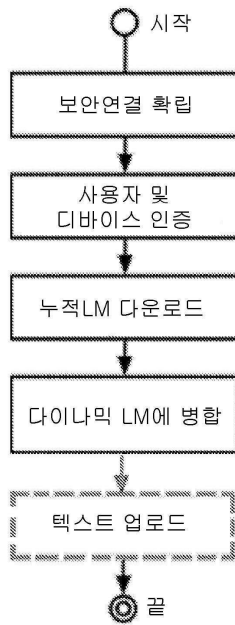
도면4c



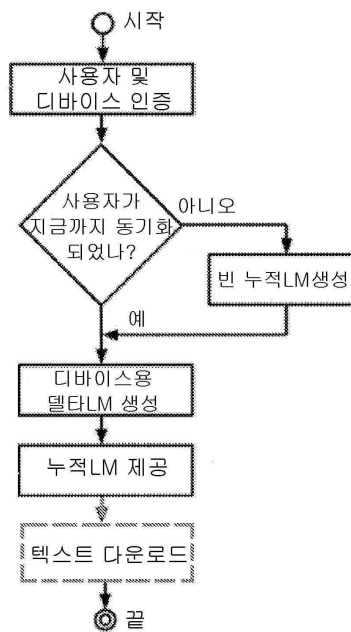
도면5



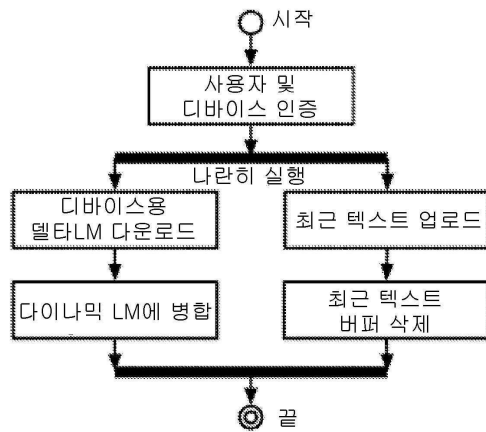
도면6



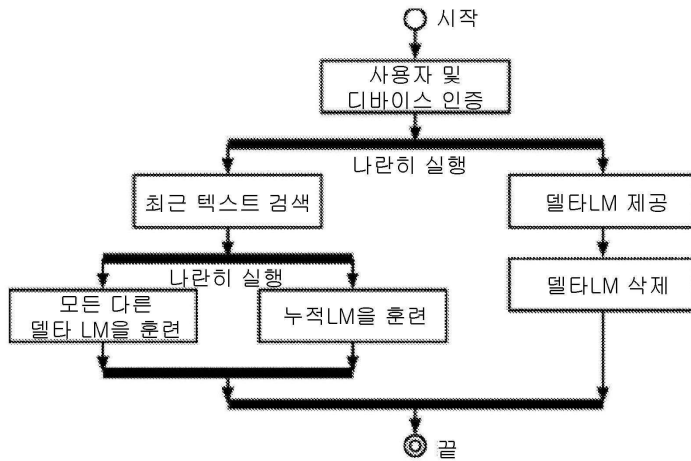
도면7



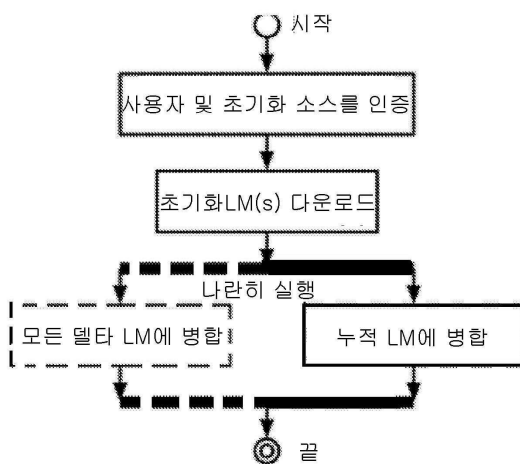
도면8



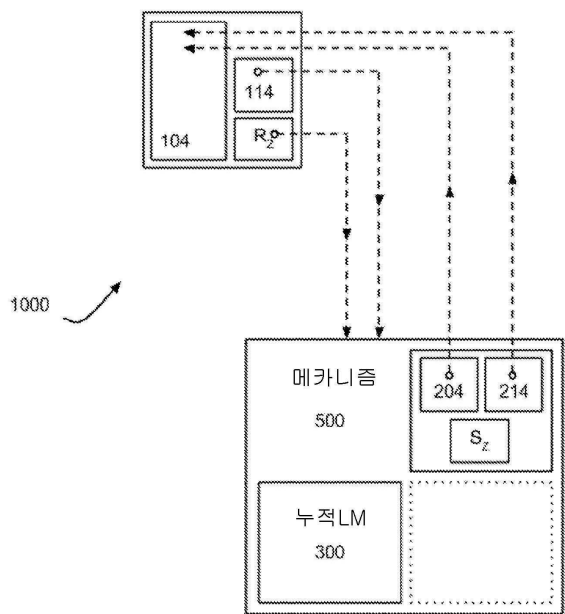
도면9



도면10



도면11



도면12

