NLP QUIZ

Külső forrás

* <https://towardsdatascience.com/my-take-on-30-questions-to-test-a-data-scientist-on-natural-language-processing-with-interactive-b8664c53be13>
* <https://medium.com/@SeoJaeDuk/my-take-on-30-questions-to-test-a-data-scientist-on-natural-language-processing-with-interactive-e45d93e6d713>

Saját kérdések

1. Hogyan nevezzük azt a módszer, amely a szöveget feladat specifikusan elemekre bontja azért, hogy azt a gépi tanuló eljárásokkal feldolgozzuk.
   1. Vektorizálás
   2. Tokenizálás (Helyes)
   3. Beágyazás
   4. Tenzorizálás
2. Jelölje meg az elemek közül Tokenizálási technikákat
   1. Szavakra bontás (Helyes)
   2. Karakterekre bontás (Helyes)
   3. N-gramm (Helyes)
   4. P-gramm
   5. Mondatokra bontás
3. Hogyan nevezzük azt a módszer amely Tokenekből a gépitanul eljárások által feldolgozható adatokat állít elő
   1. Vektorizálás (Helyes)
   2. Tokenizálás
   3. Beágyazás
   4. Tenzorizálás
4. Jelölje meg az elemek közül a Vektorizálási technikákat
   1. One-Hot (Helyes)
   2. Ont-Hot hashin trick (Helyes)
   3. Beágyazás (Helyes)
   4. N-gram
   5. bag-of-word
5. Jelölje meg a beágyazási technikákat
   1. Word2Vec (Helyes)
   2. GloVe (Helyes)
   3. One-Hot
   4. N-Gram
6. Miért hasznos a szó beágyazás
   1. Mert a szavakhoz egy olyan tömör és jól strukturált vektor teret hozunk létre, amely szemantikai jelentést kapcsol a szavakhoz (Helyes)
   2. Mert a szavakhoz egy olyan tömör és jól strukturált vektor teret hoz létre, amely növeli a gépi tanuló módszerek hatékonyságát (Helyes)
   3. Mert a szavakhoz egy olyan tömör és jól strukturált vektor teret hoz létre, amely előnyösebb a gradiens számításokhoz.
   4. Mert a szavakhoz egy olyan tömör és jól strukturált vektor teret hoz létre, amely csökkenti a veszteség függvények számítási költségét.
7. Miért hasznos az „pretrained word embeddings”
   1. Mert jó kiinduló beágyazott vektor tér kevés adattal rendelkező feladatokhoz (Helyes)
   2. Mert a nagyobb adathalmazon előre létrehozott beágyazott vektor tér minden esetbe jobban általánosít, mint a cél feladathoz meglévő adathalmazon létrehozott beágyazott vektortér.
   3. Mert ilyenkor nincs szükségünk a cél feladathoz tartozó adatokon történ beágyazásra
   4. Mert csökkenti a gépi tanuló modell összetettséget.
8. Minek a rövidítésé a RNN és mire használjuk
   1. Recurrent neural network, speciális mesterséges neurális hálózat architektúra melynek használat akkor előnyös, ha a feldolgozott adatok sorrendje között olyan összefüggés van, amely javítja a hálózat pontosságát.
   2. Repeated neural network , speciális mesterséges neurális hálózat architektúra melynek használat akkor előnyös, ha a feldolgozott adatok sorrendje között olyan összefüggés van, amely javítja a hálózat pontosságát.
   3. Recurrent neural network, speciális mesterséges neurális hálózat architektúra melynek használat akkor előnyös ha a feldolgozott adatok között ismétlődő elemek találhatóak és azok figyelembe vétele javítja a hálózat pontosságát.
   4. Repeated neural network, speciális mesterséges neurális hálózat architektúra melynek használat akkor előnyös ha a feldolgozott adatok között ismétlődő elemek találhatóak és azok figyelembe vétele javítja a hálózat pontosságát.
9. Jelölje meg a RNN típusú neurális hálózatokat
   1. GRU (Helyes)
   2. LSTM (Helyes)
   3. simple RNN (helyes)
   4. CNN
   5. Dense
10. Mi a Gated Recurrent Unit (GRU) neurális hálózat
    1. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével nem csak a közvetlen hanem azt megelőző visszacsatolásokból származó információk releváns részeit is felhasználjuk. (Helyes)
    2. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével nem csak a közvetlen hanem azt megelőző visszacsatolásokból származó összes információt felhasználjuk.
    3. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével a megelőző visszacsatolásokból származó információk hatását csökkentjük.
    4. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével a megelőző visszacsatolásokból származó információk hatását növeljük.
11. Mi a Long Short-Term Memory (LSTM) neurális hálózat
    1. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével nem csak a közvetlen hanem azt megelőző visszacsatolásokból származó információk releváns részeit is felhasználjuk. (Helyes)
    2. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével nem csak a közvetlen hanem azt megelőző visszacsatolásokból származó összes információt felhasználjuk.
    3. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével a megelőző visszacsatolásokból származó információk hatását csökkentjük.
    4. Egy speciális RNN hálózat, amely minden visszacsatolásnál kiszámol egy speciális carry értéket, amely segítségével a megelőző visszacsatolásokból származó információk hatását növeljük.
12. Mi a különbség a GRU és az LSTM között
    1. Semmi a két hálózatműködése teljesen megegyezik
    2. A GRU olyan számítási formulát használ, amelynek számítási költsége kisebb, mint az LSTM hálózaté. (Helyes)
    3. A LSTM olyan számítási formulát használ, amelynek számítási költsége kisebb, mint az GRU hálózaté.
    4. A GRU az LSTM hálózattal szemben nem csak a múltbél összefüggéseket hanem a jövőbeni összefügéseket is figyelembe veszi.
13. Minek a rövidítése BiLSTM és mikor használjuk?
    1. Bidirectional Long Short Term Memory: Akkor használjuk, ha az adathalmazban lévő sorrendi összefüggéseiből többlet információhoz jutunk, ha ellőről hátra és hatúról előre is feldolgozzuk az adatokat. (Helyes)
    2. Binary Long Short Term Memory: Akkor használjuk, ha az LSTM hálózat előnyeit bináris adatokon akarjuk felhasználni.
    3. Bidirectional Long Short Term Memory: Akkor használjuk, ha az adathalmazban lévő sorrendi összefüggéseiből többlet információhoz jutunk, ha minden egyes visszacsatolásnál nem csak a megelőző egység kimenetét, hanem a rákövetkező egység bemenetét is visszacsatoljuk.
    4. Binary Long Short Term Memory: Akkor használjuk, ha az adathalmazban lévő sorrendi összefüggéseiből többlet információhoz jutunk, ha minden egyes visszacsatolásnál a megelőző egység kimenetének bináris leképzését csatoljuk vissza a gyorsabb számítási teljesítmény eléréshez.
14. Az alábbiak közül milyen típusú konvolúciós háló használható a szöveg feldolgozáshoz?
    1. 1 dimenziós CNN (helyes)
    2. 2 dimenziós CNN
    3. 3 dimenziós CNN
    4. Egyik sem

Referencia

* Deep Learning with Python, François Chollet, 2018, Deep learning for text and sequences, chapter 6