اسم الطالب <- استبدله باسمك

**مهمة البرمجة لربيع 2023 CS585 رقم 01**

الموعد النهائي: **الأحد 11 فبراير 2024 الساعة 11:59 مساءً بتوقيت وسط أمريكا**

النقاط: **150**

**تعليمات:**

1. ضع **جميع تسليماتك (كما هو موضح أدناه) في** ملف ZIP واحد باسم:

LastName\_FirstName\_CS585\_Programming01.zip

1. أرسله إلى قسم واجبات Blackboard قبل الموعد المحدد. **لن يتم قبول أي تقدم في وقت متأخر** .

**أهداف:**

1. (50 نقطة) إجراء تحليل توزيع تكرار الكلمات الأساسي لمجموعة نصية.
2. (50 نقطة) حساب احتمالية الجملة.
3. (50 نقطة) التنبؤ بالكلمات النموذجية للغة.

**الإنجازات:**

يجب أن يتضمن طلبك ما يلي:

* **تأكد من التعليق على الكود الخاص بك بشكل كافٍ! لا توجد ملفات Jupyter Notebook!**
* **الجزء أ** : ملف (ملفات) كود بايثون. يجب تسمية ملف py الخاص بك:

cs585\_P01A\_AXXXXXXXXX.py

أين XXXXXXXXX هو رقم IIT A الخاص بك ( هذا مطلوب! ). إذا كان الحل الخاص بك يستخدم ملفات متعددة، فتأكد من تسمية الملف الرئيسي (الذي سيتم تشغيله لحل المشكلة) بهذه الطريقة وأن الآخرين يقومون بتضمين رقم IIT A الخاص بك في أسمائهم أيضًا.

* **الجزء ب** : ملف (ملفات) كود بايثون. يجب تسمية ملف py الخاص بك:

cs585\_P01B\_AXXXXXXXXX.py

أين XXXXXXXXX هو رقم IIT A الخاص بك ( هذا مطلوب! ). إذا كان الحل الخاص بك يستخدم ملفات متعددة، فتأكد من تسمية الملف الرئيسي (الذي سيتم تشغيله لحل المشكلة) بهذه الطريقة وأن الآخرين يقومون بتضمين رقم IIT A الخاص بك في أسمائهم أيضًا.

* **الجزء ج** : ملف (ملفات) كود بايثون. يجب تسمية ملف py الخاص بك:

cs585\_P01C\_AXXXXXXXXX.py

أين XXXXXXXXX هو رقم IIT A الخاص بك ( هذا مطلوب! ). إذا كان الحل الخاص بك يستخدم ملفات متعددة، فتأكد من تسمية الملف الرئيسي (الذي سيتم تشغيله لحل المشكلة) بهذه الطريقة وأن الآخرين يقومون بتضمين رقم IIT A الخاص بك في أسمائهم أيضًا.

* هذه الوثيقة مع النتائج والاستنتاجات الخاصة بك. يجب عليك إعادة تسميته إلى:

LastName\_FirstName\_cs585\_Programming01.doc أو pdf

تنسيقات MS WORD أو PDF فقط، من فضلك.

**الجزء أ [50 نقطة]:**

استخدم حزمة Python's NLTK مع المجموعة:

* بني،
* رويترز،

ل:

1. **[10 نقاط]** احصل على توزيع تكرار الكلمات (بعد إزالة جميع كلمات التوقف، استخدم كلمات التوقف المجمعة لهذا الغرض) لكلا المجموعتين،
2. **[10 نقاط]** عرض الكلمات العشرة الأولى (المرتبة من 1 إلى 10) لكلا المجموعتين على الشاشة (ضعها أيضًا في الجدول أدناه)

|  |  |
| --- | --- |
| **أفضل 10 كلمات** | |
| **بني** | **رويترز** |
|  |  |

1. **[15 نقطة]** قم بإنشاء **مخططات السجل (الرتبة) مقابل السجل (التردد)** لأول 1000 كلمة (المرتبة من 1 إلى 1000) لكلا المجموعتين (يمكنك استخدام حزمة matplotlib أو بعض حزم/أدوات التخطيط الأخرى). ضع كلا المؤامرات في الجدول أدناه.

|  |  |
| --- | --- |
| **سجل (الرتبة) مقابل مؤامرات السجل (التردد).** | |
| **بني** | **رويترز** |
|  |  |
| **هل لاحظت أي شيء مثير للاهتمام عند مقارنة جميع المؤامرات؟ اكتب تعليقاتك أدناه:** | |
|  | |

1. **[15 نقطة]** استخدم أعداد التكرارات التي تم الحصول عليها مسبقًا لحساب احتمالية حدوث unigram للكلمتين ("التقنية" وليست التقنية). استخدم الأحرف الصغيرة أولاً! **اعرض جميع الأعداد والاحتمالات ذات الصلة على الشاشة لكلا المجموعتين (أيضًا: أدخل القيم النهائية في الجدول أدناه)** . يمكن أن يكون صفرًا لبعض الكلمات.

|  |  |
| --- | --- |
| **"تقنية" / نادرًا ما تستخدم في كلمات المحادثة غير الرسمية (على سبيل المثال، "adiabatic"** | |
| **بني** | **رويترز** |
|  |  |
| **كلمة غير فنية / غير رسمية / للاستخدام اليومي (على سبيل المثال "العشاء")** | |
| **بني** | **رويترز** |
|  |  |

**الجزء ب [50 نقطة]:**

استخدم حزمة Python's NLTK مع مجموعة Brown للمهام التالية:

1. **[نقطة واحدة]** اطلب من المستخدم إدخال جملة S من لوحة المفاتيح.
2. **[1 نقطة]** تطبيق الحروف الصغيرة على S.
3. **[45 نقطة]** احسب P(S) بافتراض نموذج لغة 2 جرام ( **افترض أن احتمال أي بيجرام يبدأ أو ينهي الجملة هو 0.25** )
4. **[3 نقاط]** اعرض الجملة S، وأدرج جميع الأرقام الكبيرة الفردية واحتمالاتها، والاحتمال النهائي P(S) على الشاشة. لا بأس إذا كانت صفراً.

**الجزء ج [50 نقطة]:**

استخدم حزمة Python NLTK مع مجموعة Brown (بعد إزالة جميع كلمات التوقف؛ استخدم مجموعة كلمات التوقف لهذا الغرض) للمهام التالية:

1. **[نقطة واحدة]** ابدأ بمطالبة المستخدم بالكلمة الأولية/الرمز المميز W1. قم بتطبيق الحروف الصغيرة على W1 (وجميع الإدخالات المستقبلية). إذا لم تكن الكلمة موجودة في متن النص، فاعرض خيارين:
   1. **اسأل مجددا**
   2. **يترك**
2. **[45 نقطة]** بافتراض نموذج لغة 2 جرام، قائمة تحتوي على أفضل 3 كلمات "من المرجح أن تتبع W1" (وفقًا لتقدير احتمالية W1 والكلمة التالية). على سبيل المثال، إذا بدأ المستخدم بـ W1 = " جيد "، فيمكن عرض ما يلي ( **ملاحظة: لقد قمت بإعداد هذا التحديد وتقديرات الاحتمالية المقابلة** ):

جيد …

ما هي الكلمة التي يجب أن تتبع:

1. صباح ف(صباح الخير) = 0.25
2. مساء ع(مساء الخير) = 0.15
3. بعد الظهر ف(مساء الخير) = 0.14
4. يترك

**كرر (وأضف اختيارات الكلمات اللاحقة إلى "الجملة") حتى يحدد المستخدم (4) وإنهاء.**

إذا اختار المستخدم رقمًا غير 1،2،3 و4، **فافترض أن اختيار المستخدم هو (1)** .