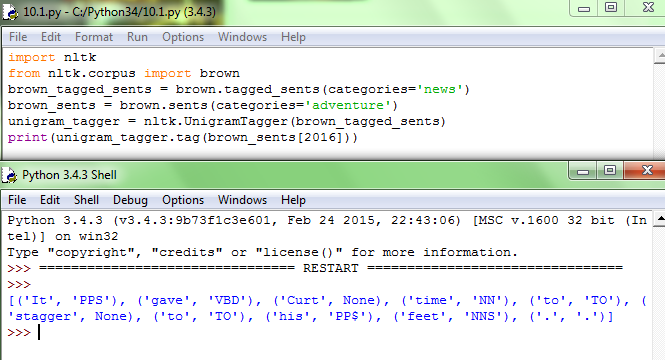
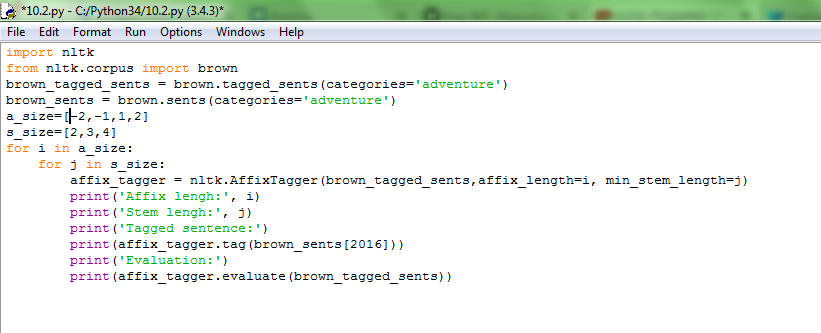
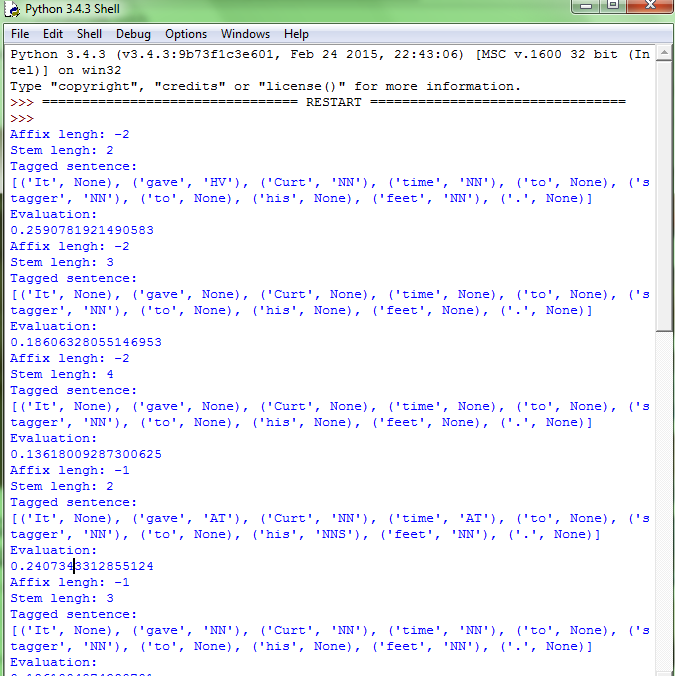
1.*Здійсніть тренування юніграм аналізатора на основі частини корпуса, який відповідає першій або другій літері прізвища студента та виконайте аналіз тексту з частини корпуса, яка відповідає першій або другій літері імені студента. Результати поясніть. Чому для деяких слів не встановлені теги.*



Для деяких слів не встановлено теги, тому що цих слів не було у тесті для тренування аналізатора. Осільки юніграм аналізатор просто шукає відовідник слова у тексті для тренування, він просто проставляє «None», для всіх слів, яких не може знайти.

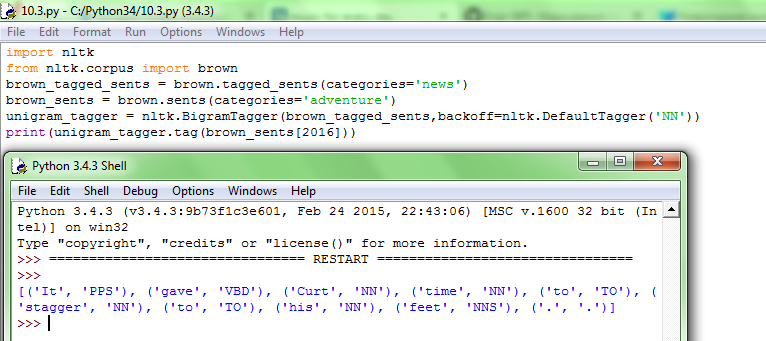
2. *Прочитати файл допомого про морфологічний аналізатор на основі афіксів (help(nltk.AffixTagger)). Напишіть програму, яка викликає аналізатор на основі афіксів в циклі, з різними значеннями довжини афіксів і мінімальними довжинами слів. При яких значеннях можна отримати кращі результати.*

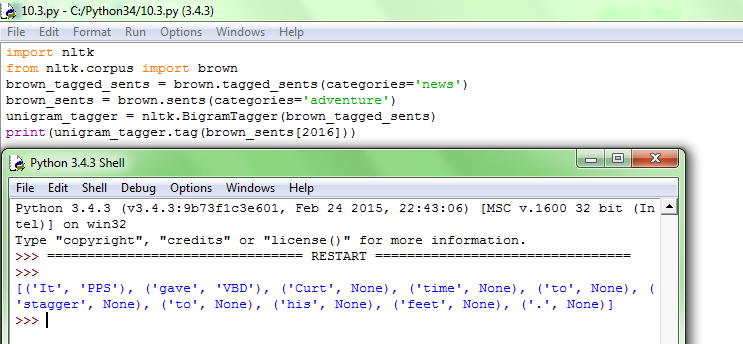




Найкращий результат отримано при довжині афікса =-2 (суфікс) та довжині основи 2.

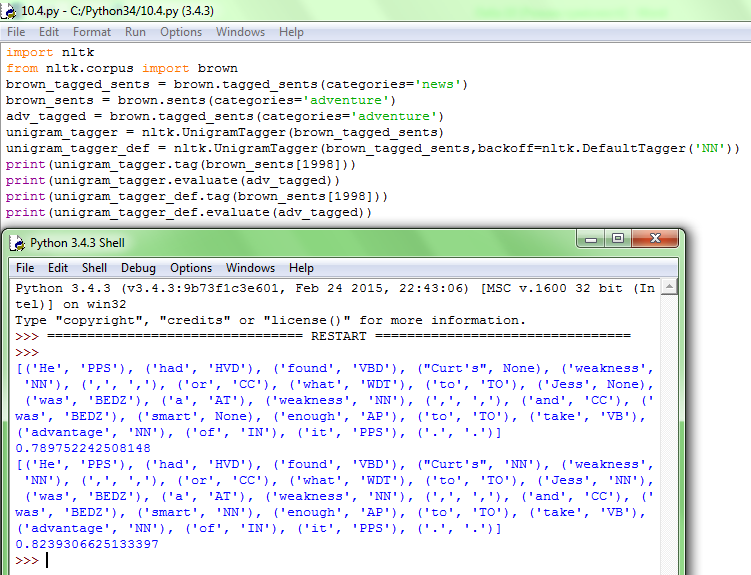
3. *Здійсніть тренування біграм аналізатора на частинах корпуса з вправи 3.1 без backoff аналізатора. Перевірте його роботу. Що відбулося з продуктивністю аналізатора? Чому?*



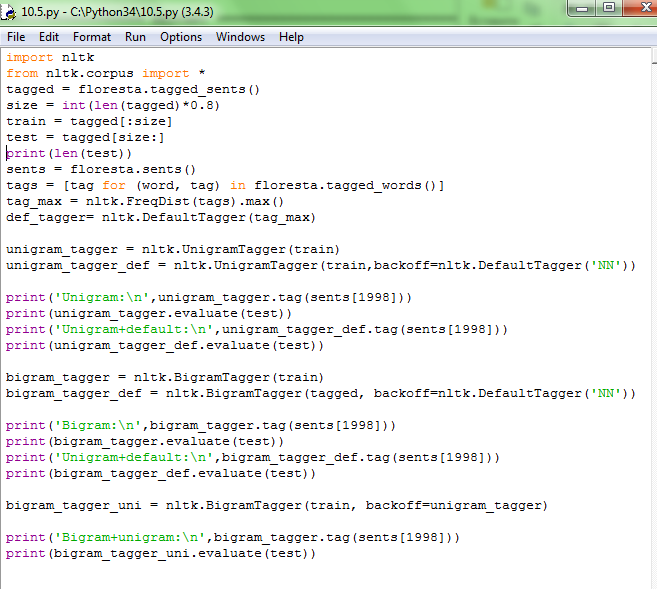


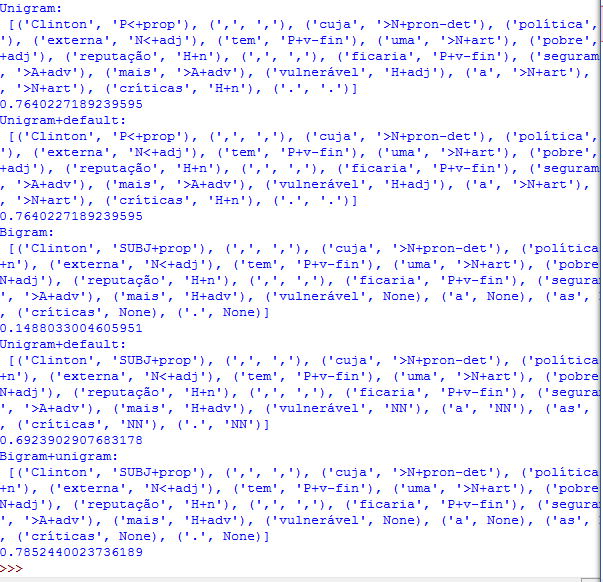
Як тільки в тексті зустрілося нове слово, аналізатор не може для нього встановити тег. Так само аналізатор не маркує наступне слово, навіть якщо воно зустрічалося при тренуванні, оскільки це слово в даних для тренування ніколи не зустрічалося після слова тег якого None . Це приводить до того що і всі наступні слова в реченні не маркуються і точність роботи аналізатора надзвичайно низька

4. *Дослідити наступні проблеми. що виникають при роботі з аналізатором на основі підстановок: що відбудеться з продуктивністю аналізатора, якщо опустити backoff аналізатор (дослідити на частині броунівського корпусу, яка відповідає першій або другій літері прізвища студента); на основі рис.1. та відповідного фрагмента програми встановити точку максимальної продуктивності незважаючи на розмір списку (об’єм оперативної пам’яті) і точку достатньої продуктивності при мінімальному розмірі списку*.



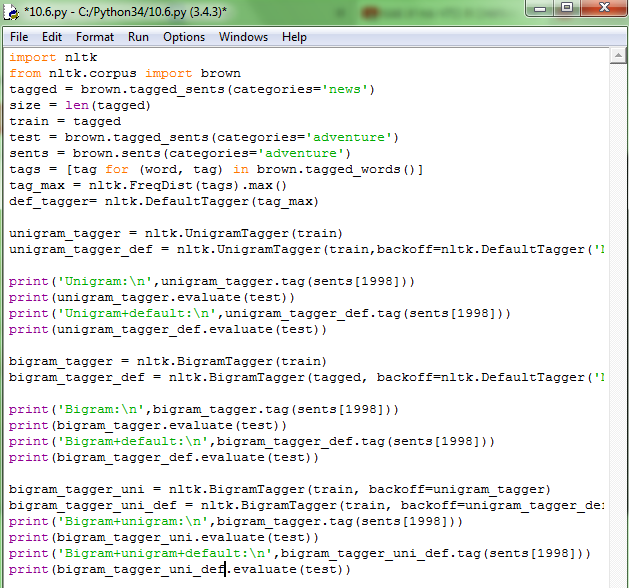
5*. Знайдіть розмічені корпуси текстів для інших мов які вивчаєте або володієте (українська, польська, німецька, російська, італійська, японська). Здійсніть тренування та оцініть продуктивність роботи різних аналізаторів та комбінацій різних аналізаторів. Точність роботи аналізаторів порівняйте з точністю роботи аналізаторів для англійських корпусів. Результати поясніть.*

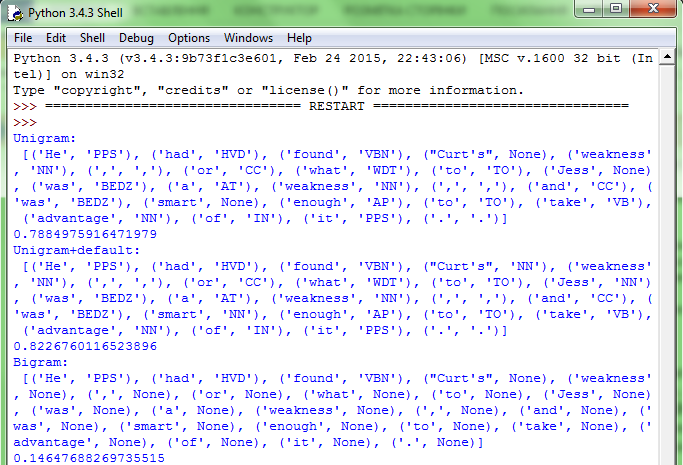




Було проаналізовано корпус португальської мови Флореста. Найкращі результати отримано за допомогою поєднання біграм та юніграм аналізаторів. Найменш ефективним виявився біграм аналізатор, якщо використовувати лише його. Як видно з результатів наступної задачі, ті самі комбінації аналізаторів дають кращі результати на корпусах англійської мови.

*6. Створити аналізатор по замовчуванню та набір юніграм і n-грам аналізаторів. Використовуючи backoff здійсніть тренування аналізаторів на частині корпуса з вправи 3.2. Дослідіть три різні комбінації поєднання цих аналізаторів. Перевірте точність роботи аналізаторів. Визначіть комбінацію аналізаторів з максимальною точністю аналізу. Змініть розмір даних на яких проводилось тренування. Повторіть експерименти для змінених даних для тренування. Результати порівняйти і пояснити.*

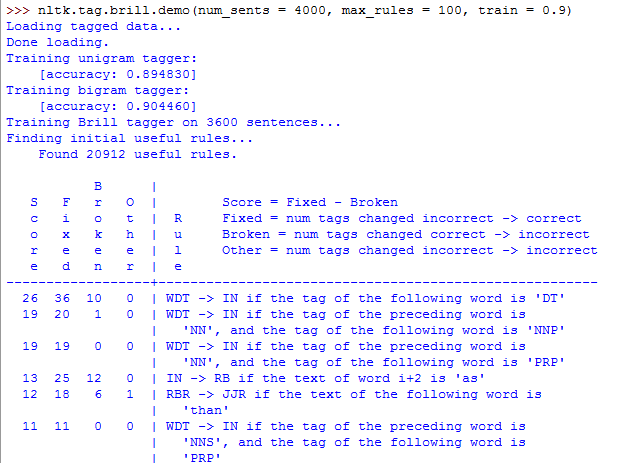






Судячи з результатів найменш ефективним є застосування лише біграм-аналізатора. Найкращий результат дає поєднання біграм-, юніграм-, та аналізатора за замовчуванням.

7. *Прочитати стрічку документування функції demo Brill аналізатора. Здійснити експерименти з різними значення параметрів цієї функції. Встановити який взаємозв’язок є між часом тренування (навчання аналізатора) і точністю його роботи*

**