Danuta Mendrala

Marcin Szeliga

Microsoft SQL Server

Modelowanie i eksploracja danych

Helion

Redaktor prowadzący: Michał Mrowiec
Korekta merytoryczna: Radosław Łebkowski
Projekt okładki: Jan Paluch

Fotografia na okładce została wykorzystana za zgodą Shutterstock.com

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: http://helion.pl (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie?sqlsme
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Kody źródłowe wybranych przykładów dostępne są pod adresem:
ftp://ftp.helion.pl/przyklady/sqlsme.zip

ISBN: 978-83-246-3440-8

Printed in Poland.

Spis treści

Rozdział 9.

Naiwny klasyfikator Bayesa firmy Microsoft

- Dlaczego klasyfikator Bayesa nazywany jest naiwnym?
- Jakie są wady i zalety naiwnego klasyfikatora Bayesa firmy Microsoft?
- Jak tworzyć modele eksploracji danych używające naiwnego klasyfikatora Bayesa firmy Microsoft?
- Jak parametryzować naiwny klasyfikator Bayesa firmy Microsoft?
- Jak za pomocą naiwnego klasyfikatora Bayesa firmy Microsoft badać zależności pomiędzy atrybutami?
- Jak zbudować klasyfikujący dokumenty filtr antyspamowy przy użyciu naiwnego klasyfikatora Bayesa firmy Microsoft?

Nazwy wszystkich przedstawionych algorytmów eksploracji danych zawierają określenie *firmy Microsoft* nie dlatego, że algorytmy te zostały wymyślone przez Microsoft, ale dlatego, że to ta firma stworzyła zastosowane w serwerze SQL implementacje tych algorytmów.

Omówienie

Opracowany przez brytyjskiego matematyka i prezbiteriańskiego duchownego Thomasa Bayesa klasyfikator należy do klasycznych algorytmów uczenia przez obserwację[1]. Wyobraźmy sobie, że spędzamy wolny czas, obserwując klientów właśnie otwartego butiku. Interesuje nas, kto (kobieta czy mężczyzna) za chwilę wejdzie do tego sklepu.

[1] Będące podstawą opisywanego klasyfikatora twierdzenie Bayesa zostało opublikowane w wydanym w 1763 roku eseju *Essay Towards Solving a Problem in the Doctrine of Chances*. Dokument ten jest dostępny pod adresem *http://www.stat.ucla.edu/history/essay.pdf*.

Ponieważ w naszym miasteczku mieszka więcej kobiet niż mężczyzn (60% mieszkańców to kobiety, a 40% — mężczyźni), początkowo prawdopodobieństwo, że klientem będzie kobieta, wynosi 60%. Jednak po pewnym czasie zebraliśmy więcej informacji o rozkładzie dnia klientów i zauważyliśmy m.in., że przed południem butik odwiedzają głównie (w 80%) kobiety, a po godzinie 15.00 75% klientów to mężczyźni. Jeżeli od teraz usłyszymy, że ktoś wchodzi do tego sklepu o 11.15, wiemy, że prawdopodobnie jest to kobieta (60%*80% = 48%), a nie mężczyzna (40%*20% = 8%). Natomiast gdybyśmy usłyszeli osobę wchodzącą do butiku o 15.30, mielibyśmy podstawy przypuszczać, że jest to mężczyzna (40%*75% = 30%), a nie kobieta (60%*25% = 15%). Ten uproszczony przykład pokazuje istotę naiwnego klasyfikatora Bayesa.

Naiwny klasyfikator Bayesa zlicza zależności występujące pomiędzy atrybutami wyjściowymi a poszczególnymi atrybutami wejściowymi, uwzględniając warunkowe i bezwarunkowe prawdopodobieństwo ich wystąpienia:

1. Prawdopodobieństwo bezwarunkowe (początkowe) zależy od rozkładu przypadków — w powyższym przykładzie reprezentowane jest ono przez fakt, że 60% mieszkańców miasteczka to kobiety.
2. Warunkowe prawdopodobieństwo zależy od zaobserwowanych zdarzeń — w powyższym przykładzie zaobserwowaliśmy, że 75% klientów odwiedzających butik po południu to mężczyźni.

Obliczone na podstawie twierdzenia Bayesa ($P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$)[2] prawdopodobieństwa są następnie mnożone, a więc każde z nich ma taki sam wpływ na ostateczny wynik. To założenie jest prawdziwe, o ile poszczególne atrybuty wejściowe są od siebie niezależne[3]. W przeciwnym razie wpływ atrybutu skorelowanego z innym atrybutem jest większy, niż być powinien. Ponieważ w praktyce atrybuty bardzo często są ze sobą powiązane (np. wykształcenie wpływa na dochód, wciąż też występuje statystyczna zależność pomiędzy płcią a zawodem), ignorujący je klasyfikator Bayesa nazywa się naiwnym.

Naiwność klasyfikatora Bayesa wynika również z tego, że gdy pewna zależność nie wystąpiła w przypadkach treningowych (np. dotychczas w sobotę butik odwiedzały wyłącznie kobiety), obliczone przez niego prawdopodobieństwo, że klientem butiku w sobotę będzie mężczyzna, wyniesie 0%. Problem ten możemy rozwiązać, dodając 1 do wszystkich przyporządkowań stanów atrybutów do klas wyjściowych, czyli stosując estymację Laplace'a.

Obliczanie wyniku poprzez mnożenie prawdopodobieństw ma jeszcze jedną wadę. Jeżeli te prawdopodobieństwa są bardzo małe, co ma miejsce, gdy lista atrybutów jest długa i gdy atrybuty przyjmują wiele stanów, błędy ich zaokrąglania zaczynają wpływać na wyniki.

---

[2] *P(A)* oznacza prawdopodobieństwo a priori wystąpienia klasy *A*, tj. prawdopodobieństwo, że przypadek należy do klasy *A*; *P(B|A)* oznacza prawdopodobieństwo a posteriori, że przypadek *A* należy do klasy *B*, natomiast *P(B)* — prawdopodobieństwo a priori wystąpienia przypadku *B*.

[3] Nieprzyjęcia założenia o niezależności zmiennych wejściowych wymagałoby obliczenie k^p prawdopodobieństw, gdzie *p* jest liczbą zmiennych, a *p* — liczbą ich stanów. Na przykład dla 30 zmiennych binarnych trzeba by wykonać 2^{30} (1 073 741 824) operacji.

Ograniczenia

Pierwsze ograniczenie wynika ze sposobu działania naiwnego klasyfikator Bayesa — policzenie prawdopodobieństwa wystąpienia danego stanu jest możliwe tylko dla atrybutów dyskretnych, a więc atrybuty ciągłe są ignorowane przez naiwny klasyfikator Bayesa firmy Microsoft.

Drugie ograniczenie jest mniej oczywiste — naiwny klasyfikator Bayesa należy do klasyfikatorów liniowych i nie nadaje się do rozwiązywania problemów nieliniowych, czyli takich, w których stan atrybutu wyjściowego zależy od kombinacji stanów atrybutów wejściowych. Problemem nieliniowym jest np. kwestia określenia koloru pól na szachownicy.

Połowa pól na szachownicy jest biała, druga połowa — czarna. Czy znając kolumnę i wiersz, jesteśmy w stanie określić kolor pola znajdującego się na ich przecięciu? Spróbujmy użyć naiwnego klasyfikatora Bayesa firmy Microsoft do znalezienia odpowiedzi na to pytanie.

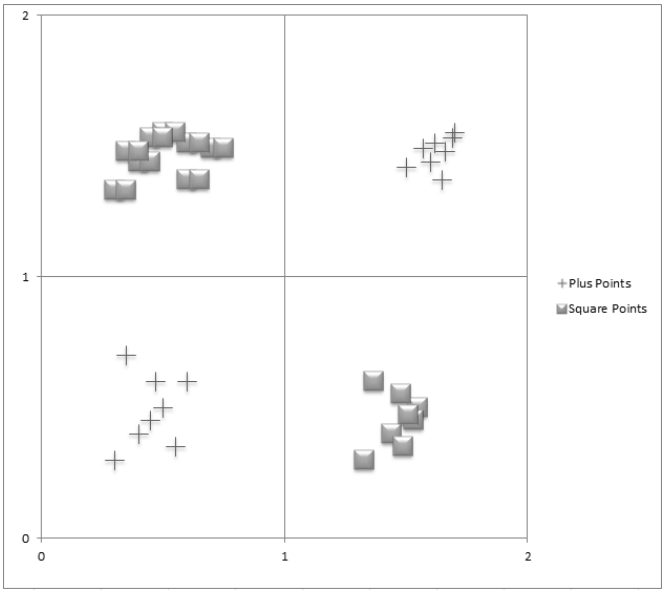
1. Otwórz przykładowy skoroszyt Excela i przejdź do arkusza *Chessboard*.
2. Zaznacz znajdującą się w nim tabelę. Jej pierwsza kolumna zawiera litery kolumn, druga — numery wierszy, a trzecia kolory pól szachownicy.
3. Kliknij znajdujący się w sekcji *Data Modeling* przycisk *Classify*.
4. Jako parametr wyjściowy i wejściowy wybierz `Color`, a jako użyty do klasyfikacji algorytm wybierz *Microsoft Naive Bayes*.
5. Przeznacz wszystkie dane do treningu i zakończ działanie kreatora, tworząc tymczasowy model eksploracji danych.

Okaże się, że algorytm nie znalazł żadnych zależności pomiędzy kolumną i wierszem pola na szachownicy a kolorem pola znajdującego się na ich przecięciu — wszystkie zakładki wizualizatora będą puste, z wyjątkiem zakładki *Dependency Network*, w której znajdziemy wyłącznie wyjściowy atrybut `Color`.

Zastanówmy się, od czego zależy kolor pól szachownicy. Czy zależy on od wierszy? Nie, w każdym wierszu 50% pól jest czarnych, a 50% białych. Nie zależy on również od kolumn, lecz od kombinacji wierszy i kolumn. Ponieważ naiwny klasyfikator Bayesa jest klasyfikatorem liniowym, nie znalazł powyższych zależności nieliniowych.

Tak postawiony problem nie zostałby rozwiązany również przez drzewa decyzyjne, czyli klasyfikator nieliniowy — w każdym wierszu i w każdej kolumnie białych pól jest dokładnie tyle samo co czarnych. Różnicę pomiędzy klasyfikatorami liniowymi i nieliniowymi pokazuje kolejny przykład. Tym razem kształt figury również nie zależy od jego poszczególnych współrzędnych, ale od ich kombinacji (rysunek 9.1).

1. Przejdź do arkusza `Linear`.
2. Przeprowadź klasyfikację znajdujących się w nim danych, wybierając na atrybuty wejściowe kolumny `RangeX`, `RangeY` i `Shape`, a na atrybut wyjściowy kolumnę `Shape`.
3. Jako użyty do klasyfikacji algorytm wybierz *Microsoft Naive Bayes*.



Rysunek 9.1. *W pierwszej i trzeciej ćwiartce znajdują się wyłącznie krzyżyki, w drugiej i czwartej — same kwadraty*

4. Przeznacz wszystkie dane do treningu i zakończ działanie kreatora, tworząc tymczasowy model eksploracji danych.

Tym razem algorytm również nie znajdzie żadnych zależności pomiędzy współrzędnymi a kształtem figur.

Pomimo tych ograniczeń naiwny klasyfikator Bayesa firmy Microsoft dobrze radzi sobie z wykrywaniem zależności pomiędzy poszczególnymi atrybutami, a jego prostota (i związane z nią szybkość oraz małe zapotrzebowanie na pamięć i moc obliczeniową), jak również łatwość interpretacji wyników czynią z niego przydatny i często używany algorytm eksploracji danych.

Parametry

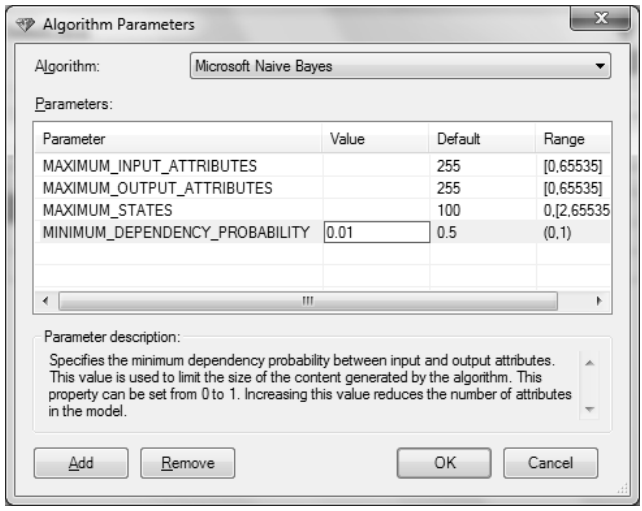
Naiwny klasyfikator Bayesa firmy Microsoft przyjmuje następujące parametry:

1. `MAXIMUM_INPUT_ATTRIBUTES` — parametr dostępny tylko w edycji Enterprise; określa maksymalną liczbę atrybutów wejściowych (objaśniających). Po jej przekroczeniu (domyślna wartość wynosi 255) analizowanych będzie tylko 255 atrybutów wejściowych najsilniej powiązanych z atrybutami wyjściowymi (objaśnianymi). Zmiana tego parametru na 0 spowoduje uwzględnienie wszystkich atrybutów wejściowych[4].

[4] Maksymalna liczba atrybutów wynosi 65 535 i jest tak duża, że w praktyce nie spotkamy się z wynikającymi z niej ograniczeniami. Z pierwszej części książki wiadomo, że dane wejściowe powinny zawierać jak najwięcej informacji (a dokładnie, że entropia atrybutów wejściowych względem wyjściowych powinna być jak największa), tymczasem utworzenie kilkudziesięciu tysięcy atrybutów raczej zmniejszyłoby (a nie zwiększyło) ilość tych informacji. Ponadto dane właściwie reprezentujące wszystkie możliwe zależności pomiędzy tyloma atrybutami byłyby liczone w milionach terabajtów.

2. MAXIMUM_OUTPUT_ATTRIBUTES — parametr dostępny tylko w edycji Enterprise; określa maksymalną liczbę atrybutów wyjściowych. Po jej przekroczeniu (domyślna wartość wynosi 255) zostanie uwzględnionych tylko 255 najczęściej występujących atrybutów wyjściowych. Zmiana tego parametru na 0 spowoduje uwzględnienie wszystkich atrybutów wyjściowych.
3. MAXIMUM_STATES — parametr dostępny tylko w edycji Enterprise; określa maksymalną liczbę uwzględnianych stanów atrybutów. Po jej przekroczeniu (domyślna wartość wynosi 100) analizowanych będzie tylko 100 najczęściej występujących stanów atrybutów, a pozostałe zostaną potraktowane jak wartości brakujące. Zmiana tego parametru na 0 spowoduje uwzględnienie wszystkich stanów atrybutów.
4. MINIMUM_DEPENDENCY_PROBABILITY — określa (w skali od 0 do 1) minimalne prawdopodobieństwo znalezienia zależności pomiędzy atrybutami wejściowymi a wyjściowymi. Zmiana tego parametru nie ma żadnego wpływu na trening algorytmu, a jedynie na liczbę zwracanych (znalezionych) zależności. Domyślna wartość wynosi 0,5 — jest to wartość, przy której wizualizatory tego algorytmu zwracają informacje tylko o zależnościach, których prawdopodobieństwo wystąpienia jest większe od prawdopodobieństwa ich braku.

Żeby przekonać się, jak zmiana parametru MINIMUM_DEPENDENCY_PROBABILITY wpłynie na zdolność naiwnego klasyfikatora Bayesa firmy Microsoft do rozwiązywania problemów nieliniowych, raz jeszcze przeprowadź klasyfikację kolorów pól na szachownicy, tym razem ustawiając wartość tego parametru na 0,01 (rysunek 9.2).



Rysunek 9.2. *Naiwny klasyfikator Bayesa firmy Microsoft to prosty algorytm eksploracji danych; jego działaniem możemy sterować w bardzo ograniczonym zakresie*

Zgodnie z oczekiwaniami obniżenie wartości tego parametru nie wpłynęło na otrzymane wyniki — algorytm nadal nie jest w stanie znaleźć żadnych zależności pomiędzy kolumną i wierszem pola na szachownicy a jego kolorem.

Zastosowania naiwnego klasyfikatora Bayesa

„Naiwność" klasyfikatora Bayesa ogranicza jego stosowanie w modelach klasyfikacyjnych, ale w żaden sposób nie zmniejsza jego wartości dla modeli opisowych. W szczególności jego szybkość i małe wymagania dotyczące pamięci czynią z niego doskonałe narzędzie do oceny danych wejściowych.

Drugi z opisanych poniżej przykładów demonstruje predykcyjne możliwości naiwnego klasyfikatora Bayesa — jeżeli tylko atrybuty wejściowe rzeczywiście są od siebie niezależne lub ewentualne zależności między nimi są nieistotne w ramach przyjętego modelu (jak ma to miejsce np. podczas oceniania wiadomości e-mail na podstawie poszczególnych słów, czy jest ona spamem), algorytm ten okazuje się szybkim i dokładnym klasyfikatorem.

W serwerze SQL klasyfikator Bayesa firmy Microsoft stosowany jest do klasyfikacji i — z pewnymi ograniczeniami — asocjacji.

Badanie zależności pomiędzy atrybutami

Naiwny klasyfikator Bayesa firmy Microsoft doskonale nadaje się (o czym powiedziano w rozdziale 3.) do analizowania zależności pomiędzy atrybutami. W tym punkcie utworzymy model analizujący zależności pomiędzy atrybutami klientów firmy Adventure Works:

1. Uruchom konsolę SSMS i połącz się z serwerem SSAS.
2. Zaznacz bazę analityczną `DataMining` i wyświetl okno edytora DMX.
3. Utwórz w tej bazie poniższy model eksploracji danych (tworząc model za pomocą instrukcji `CREATE MINING MODEL`, automatycznie utworzymy strukturę o nazwie tworzonego modelu, uzupełnioną o sufiks `_Structure`):

```
CREATE MINING MODEL CustomersAnalysis (
    [ID]                    LONG KEY,
    [Age]                   LONG DISCRETIZED(CLUSTERS,5),
    [MaritalStatus]         TEXT DISCRETE PREDICT,
    [Gender]                TEXT DISCRETE PREDICT,
    [TotalChildren]         LONG DISCRETE PREDICT,
    [NumberChildrenAtHome] LONG DISCRETE PREDICT,
    [Education]             TEXT DISCRETE PREDICT,
    [Occupation]            TEXT DISCRETE PREDICT,
    [YearlyIncome]          LONG DISCRETIZED(CLUSTERS,8),
    [HouseOwnerFlag]        TEXT DISCRETE PREDICT,
    [NumberCarsOwned]       LONG DISCRETE PREDICT,
    [TotalAmount]           LONG DISCRETIZED(CLUSTERS,8) PREDICT,
    [TotalQuantity]         LONG DISCRETE PREDICT,
    [BikesQuantity]         LONG DISCRETE PREDICT,
    [BikesAmount]           LONG DISCRETIZED(CLUSTERS,8) PREDICT,
```

```
    [ClothingQuantity]     LONG DISCRETE PREDICT,
    [ClothingAmount]       LONG DISCRETIZED(CLUSTERS,8) PREDICT,
    [AccessoriesQuantity]  LONG DISCRETE PREDICT,
    [AccessoriesAmount]    LONG DISCRETIZED(CLUSTERS,8) PREDICT,
    [MonthsAsCustomer]     LONG DISCRETIZED(CLUSTERS,10) PREDICT )
USING Microsoft_Naive_Bayes
```

Zwróć uwagę, że wszystkie atrybuty są dyskretne lub poddane dyskretyzacji oraz że wszystkie one zostały użyte w roli atrybutów wejściowych i wyjściowych.

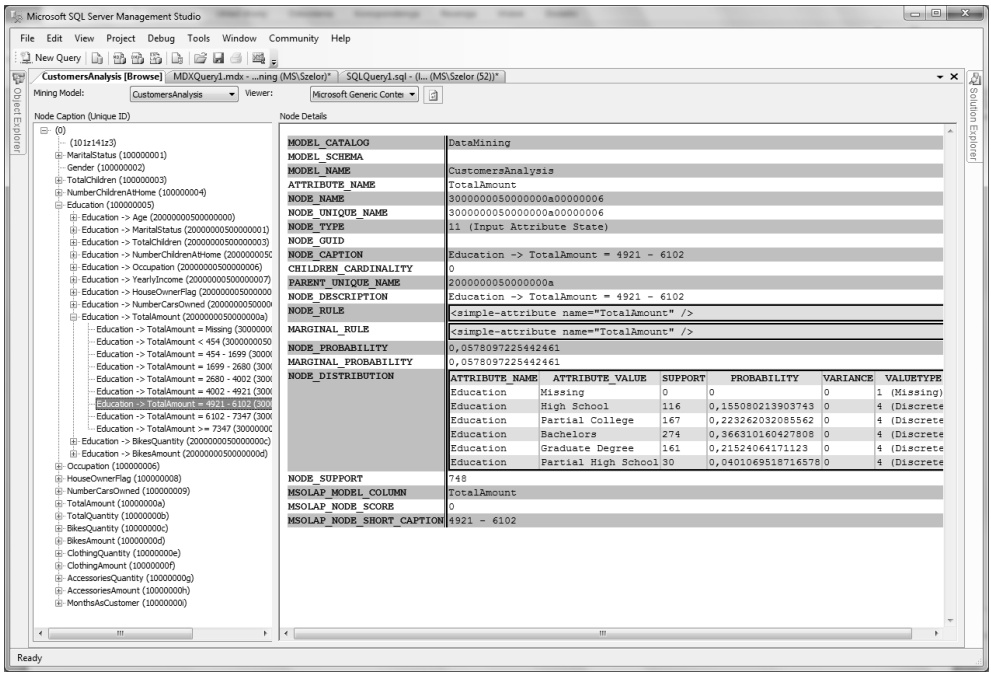
Utwórz, np. korzystając z dołączonego do książki skryptu XMLA, źródło danych `Adventure Works DW` i skonfiguruj nazwę i hasło użytkownika, z którego uprawnieniami serwer SSAS będzie łączył się z tym źródłem danych, a następnie przetwórz ten model, wykonując poniższą instrukcję:

```
INSERT INTO CustomersAnalysis([ID], [Age], [MaritalStatus], [Gender], [TotalChildren]
,[NumberChildrenAtHome], [Education], [Occupation], [YearlyIncome], [HouseOwnerFlag]
,[NumberCarsOwned], [TotalAmount], [TotalQuantity], [BikesQuantity], [BikesAmount]
,[ClothingQuantity], [ClothingAmount], [AccessoriesQuantity], [AccessoriesAmount]
,[MonthsAsCustomer])
OPENQUERY ([Adventure Works DW], 'SELECT [ID], [Age], [MaritalStatus],
[Gender], [TotalChildren], [NumberChildrenAtHome], [Education], [Occupation],
[YearlyIncome], [HouseOwnerFlag], [NumberCarsOwned], [TotalAmount], [TotalQuantity],
[BikesQuantity], [BikesAmount], [ClothingQuantity], [ClothingAmount],
[AccessoriesQuantity], [AccessoriesAmount] ,[MonthsAsCustomer]
FROM [dbo].[CustomersHistoryTrain]')
```

Po wyświetleniu raportu *Dependency Network* (żeby wyświetlić okno z wizualizatorami bezpośrednio z konsoli SSMS, należy kliknąć model eksploracji danych i wybrać z menu kontekstowego *Browse*) przekonamy się, że używany w poprzednich modelach w roli atrybutu wyjściowego atrybut `TotalAmount` jest silnie powiązany nie tylko z atrybutami opisującymi klientów (takimi jak `Age`, `Occupation` czy `TotalChildren`), ale również z atrybutami opisującymi historię zakupów tych klientów (takimi jak `AccessoriesAmount`, `BikesAmount`, `ClothingAmount` czy `TotalQuantity`). Jednak te ostatnie atrybuty są silnie powiązane nie tylko z objaśnianym atrybutem `TotalAmount`, ale również ze sobą nawzajem. Z rozdziału 5. wiadomo, że w modelach klasyfikacyjnych nie należy używać w roli atrybutów wejściowych silnie powiązanych ze sobą atrybutów, dlatego atrybuty te nie były używane w utworzonych wcześniej modelach.

Raport zależności nie zawiera informacji na temat stanów poszczególnych atrybutów. Te dane znajdziemy w pozostałych raportach wizualizatora naiwnego klasyfikatora Bayesa firmy Microsoft lub odczytując strukturę modelu. Wizualizator każdego algorytmu eksploracji danych można zastąpić ogólnym wizualizatorem *Microsoft Generic Content Tree Viewer*, zwracającym informację na temat struktury modelu.

Wyświetl go, a następnie z listy węzłów modelu wybierz węzeł opisujący zależności pomiędzy atrybutem `Education` a poszczególnymi stanami atrybutu `TotalAmount` (rysunek 9.3).



Rysunek 9.3. *Szczegółowe informacje na temat modeli eksploracji danych wraz z ich formatowaniem można skopiować do schowka i wkleić np. do dokumentu Worda*

Modele naiwnego klasyfikatora Bayesa firmy Microsoft liczą tyle węzłów drugiego poziomu (węzłów typu 9.), ile jest zdefiniowanych atrybutów wejściowych (węzłem pierwszego poziomu jest sam model eksploracji danych). Listę tych węzłów wraz z ich identyfikatorami można odczytać, wywołując poniższą procedurę:

```
CALL GetPredictableAttributes ('CustomersAnalysis')
```

Na trzecim poziomie znajdują się węzły zawierające atrybuty wejściowe (węzły typu 10.), a na czwartym (w węzłach typu 11.) — znalezione zależności pomiędzy poszczególnymi atrybutami wejściowymi a atrybutem wyjściowym, nadrzędnym dla danego węzła.

Klasyfikacja dokumentów

Analiza dokumentów tekstowych wymaga ich wcześniejszego podzielenia na frazy — to występowanie lub brak w dokumencie poszczególnych fraz będzie podstawą ich oceny. Analiza dokumentów tekstowych przypomina więc analizę koszykową: koszyki zakupów analizowane są pod kątem występowania w nich poszczególnych towarów, dokumenty tekstowe — pod kątem występowania w nich poszczególnych fraz.

Podzielone na frazy dokumenty mogą być:

1. Klasyfikowane — frazy zapisane w tabeli zagnieżdżonej będą podstawą zaklasyfikowania dokumentu np. jako spam.

2. Dzielone na segmenty na podstawie częstotliwości występowania w nich poszczególnych fraz.
3. Kojarzone ze sobą na podstawie występujących w nich fraz.

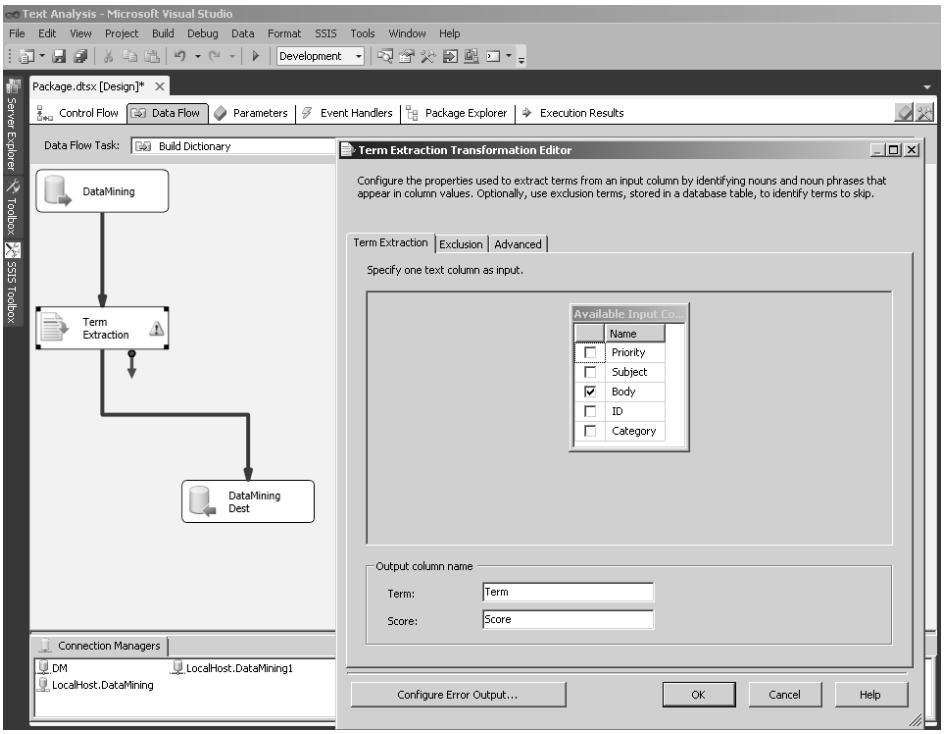
W tym punkcie przeprowadzimy klasyfikację wiadomości e-mail. Wymaga to:

1. Zbudowania słownika zawierającego wszystkie frazy występujące w tych dokumentach.
2. Dekompozycji poszczególnych dokumentów na frazy zapisane w słowniku.
3. Zbudowania modelu klasyfikacyjnego.
4. Rozłożenia ocenianych dokumentów na frazy i sklasyfikowania ich przy użyciu zbudowanego modelu.

Do zbudowania słownika użyjemy transformacji *Term Extraction* usługi SSIS:

1. Uruchom Business Intelligence Development Studio, utwórz nowy projekt typu Integration Services i nazwij go `Text Analysis`.
2. Dodaj do pakietu SSIS zadanie *Data Flow Task* i nazwij je `Build Dictionary`.
3. Przejdź na zakładkę *Data Flow*.
4. Dodaj do zadania `Build Dictionary` transformację *ADO.NET Source* i pobierz za jego pomocą dane z tabeli `EMails`, znajdującej się w przykładowej bazie danych `DataMiningDW`.
5. Dodaj do tego zadania transformację *Term Extraction* i połącz ją z domyślnym (zielonym) wyjściem utworzonego źródła danych.
6. Dwukrotnie kliknij tę transformację — wyświetli się okno edytora Term Extraction Transformation Editor:
   a) Na zakładce *Term Extraction* wybierz kolumnę, w której przechowywane są treści wiadomości e-mail (rysunek 9.4).
   b) Zakładka *Exclusion* pozwala wskazać tabelę zawierającą frazy wykluczone ze słownika.
   c) Przejdź na zakładkę *Advanced*. Pozwala ona skonfigurować sposób rozkładania tekstu na frazy: m.in. to, czy ma on być dzielony na pojedyncze wyrazy lub tylko na frazy, wybrać sposób oceniania fraz (mogą być one oceniane tylko na podstawie częstotliwości występowania w danym dokumencie oraz z uwzględnieniem tego, jak często fraza występowała we wszystkich dokumentach[5]), minimalną liczbę wystąpień fraz oraz ich maksymalną długość w słowach.
7. Zamknij okno edytora przyciskiem *OK*.

[5] Ocena frazy jest tym wyższa, im częściej występuje ona w dokumencie, ale metoda TFIDF dodatkowo obniża oceny fraz często występujących we wszystkich dokumentach.

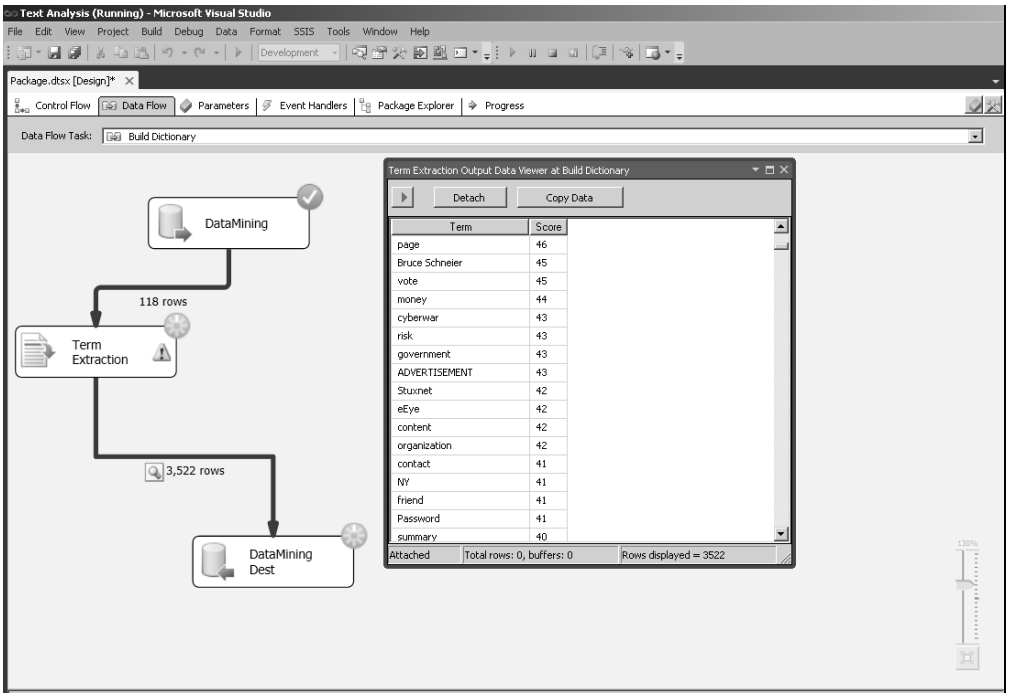


Rysunek 9.4. *Wynikiem transformacji będą dwie nowe kolumny: w kolumnie o domyślnej nazwie Term zostaną zapisane frazy znaczeniowe, w kolumnie Score — punkty obliczone na podstawie częstotliwości ich występowania*

8. Dodaj do zadania transformację *ADO.NET Destination* i utwórz za pomocą tego zadania w bazie danych `DataMining` tabelę `Dictionary`, w której zapisane zostaną frazy i ich oceny.
9. Uruchom pakiet SSIS (rysunek 9.5).

Po zbudowaniu słownika możemy rozłożyć poszczególne wiadomości e-mail na frazy:

1. Przejdź do zakładki *Control Flow*, dodaj do pakietu kolejne zadanie *Data Flow Task* i nazwij je `Decompose Documents`.
2. Połącz zadanie `Build Dictionary` z zadaniem `Decompose Documents` — w ten sposób najpierw zostanie utworzony słownik, który następnie zostanie użyty do dekompozycji wiadomości e-mail.
3. Kliknij dwukrotnie to zadanie lewym przyciskiem myszy — wyświetli się ono w edytorze przepływu danych.
4. Dodaj do zadania `Decompose Documents` transformację *ADO.NET Source* i pobierz za jego pomocą dane z tabeli `EMails` znajdującej się w przykładowej bazie danych `DataMiningDW`.
5. Dodaj do tego zadania transformację *Term Lookup* i połącz ją z domyślnym (zielonym) wyjściem utworzonego źródła danych.



Rysunek 9.5. *Pakiet SSIS tworzący słownik fraz występujących w wiadomościach e-mail (frazy zostały pokazane poprzez kliknięcie strzałki łączącej dwa ostatnie zadania i wybranie opcji Enable Data Viewer)*

6. Dwukrotnie kliknij tę transformację — wyświetli się okno edytora Term Lookup Transformation Editor:

   a) Zakładka *Reference Table* pozwala wskazać tabelę słownikową — połącz się z bazą `DataMiningDW` i wybierz tabelę `Dictionary`.

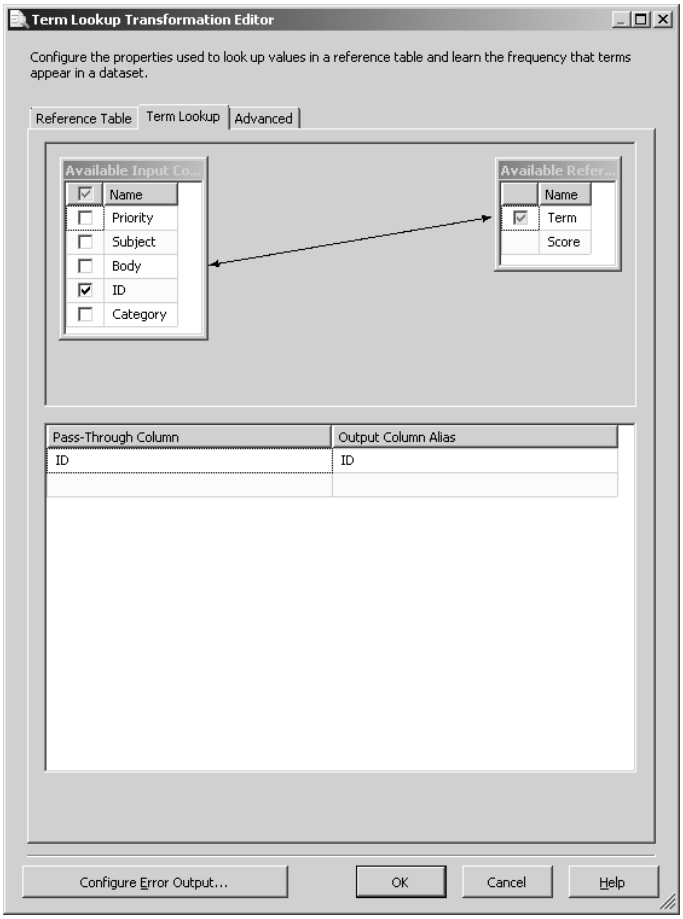
   b) Przejdź na zakładkę *Term Lookup* i połącz kolumnę `Body` tabeli `Emails` z kolumną `Term` tabeli `Dictionary`. Ponieważ tabela utworzona za pomocą tej transformacji będzie musiała zostać powiązana z nadrzędną tabelą `Emials`, dodaj do jej wyniku zawartość kolumny `ID` (rysunek 9.6).

   c) Zatwierdź zmiany przyciskiem *OK*.

7. Dodaj do zadania transformację *ADO.NET Destination* i utwórz za jego pomocą w bazie danych `DataMinigDW` tabelę `EmailsFragments`, w której zapisane zostaną zdekomponowane wiadomości e-mail.

Żeby ponowne uruchomienie pakietu nie powodowało duplikowania wierszy zapisanych w tabelach `Dictionary` i `EmailsFragments`:

1. Dodaj do niego zadanie *Execute T-SQL Statement Task*.
2. Połącz je z lokalnym serwerem SQL.



Rysunek 9.6. *Transformacja Term Lookup pozwoli nam zapisać w tabeli podrzędnej fraz informacje o tym, ile razy wystąpiły one w każdym dokumencie, oraz identyfikatory dokumentów, w których te frazy zostały znalezione*

3. W polu *T-SQL Statement* wpisz poniższe instrukcje:

```
USE DataMiningDW
GO
IF EXISTS (SELECT * FROM sys.tables WHERE name='Dictionary')
TRUNCATE TABLE dbo.Dictionary
GO
IF EXISTS (SELECT * FROM sys.tables WHERE name='EmailsFragments')
TRUNCATE TABLE dbo.EmailsFragments
GO
```

4. Połącz to zadanie z zadaniem `Build Dictionary`.

5. Uruchom i zapisz gotowy pakiet SSIS.

Dysponując przygotowanymi w ten sposób danymi źródłowymi, możemy już zbudować model klasyfikujący dokumenty. Nasz model będzie je klasyfikował wyłącznie na podstawie tematów i priorytetów wiadomości oraz znajdujących się w nich fraz — to, ile razy występuje w nich dana fraza, pominiemy. Dla odmiany model ten utworzymy w środowisku BIDS:

1. Połącz się z analityczną bazą danych `DataMining`.
2. Utwórz nowy widok danych źródłowych i dodaj do niego tabele `Emails` i `EmailsFragments`.
3. Połącz te tabele, przeciągając kolumnę `ID` tabeli `EmailsFragments` do kolumny `ID` tabeli `Emails`.
4. Analizując przykładowe dane, zwróć uwagę, że zaledwie 371 (1,5%) fraz pochodzi z wiadomości oznaczonych jako spam. Ponieważ nasz model ma klasyfikować dokumenty, musimy zmienić rozkład atrybutu wyjściowego, sztucznie zwiększając częstotliwość występowania fraz wskazujących na niechciane wiadomości:
   a) Dodaj do widoku danych źródłowych nazwane zapytanie `SelectedFragments`.
   b) Odczytaj w tym zapytaniu wszystkie fragmenty niechcianych wiadomości uzupełnione o 2% losowo wybranych fragmentów pozostałych wiadomości:

```
SELECT E.ID, Term, Frequency, NEWID() as n
FROM dbo.EmailsFragments AS F
JOIN dbo.EMails AS E ON E.ID=F.ID
WHERE E.Category='SPAM'
UNION ALL
SELECT TOP 2 PERCENT E.ID, Term, Frequency, NEWID()
FROM dbo.EmailsFragments AS F
JOIN dbo.EMails AS E ON E.ID=F.ID
WHERE E.Category<>'SPAM'
ORDER BY NEWID();
```

   c) Połącz utworzone zapytanie z tabelą `Emails`, przeciągając jego kolumnę `ID` do kolumny `ID` tabeli `Emails`.
5. Zapisz zmiany i zamknij edytor widoku danych źródłowych.
6. Z wykorzystaniem kreatora utwórz nową strukturę i model eksploracji danych:
   a) Pobierz dane z tabel relacyjnej bazy danych.
   b) Wybierz naiwny klasyfikator Bayesa firmy Microsoft.
   c) Wskaż widok danych źródłowych utworzony w poprzednich punktach.
   d) Na tabelę nadrzędną (ang. *Case*) wybierz tabelę `Emails`, na tabelę zagnieżdżoną (ang. *Nested*) — nazwane zapytanie `SelectedFragments`.
   e) Zaznacz kolumnę klucza zagnieżdżonego przypadku (kolumnę `Term`), dodaj do listy wejściowych atrybutów kolumny `Category`, `Priorytet` i `Subject`, a na atrybut wyjściowy wybierz kolumnę `Category`.
   f) Użyj wszystkich danych jako przypadków treningowych.
   g) Zwróć uwagę, że naiwny klasyfikator Bayesa firmy Microsoft nie umożliwia zaznaczenia opcji *Allow drill through* (przedstawiona w poprzednim punkcie struktura modeli tego algorytmu jest mało intuicyjna i nie pozwala w prosty sposób powiązać znalezionych zależności z poszczególnymi przypadkami). Zakończ pracę kreatora.

7. Przetwórz zbudowany model. Wyświetlą się dwa ostrzeżenia:

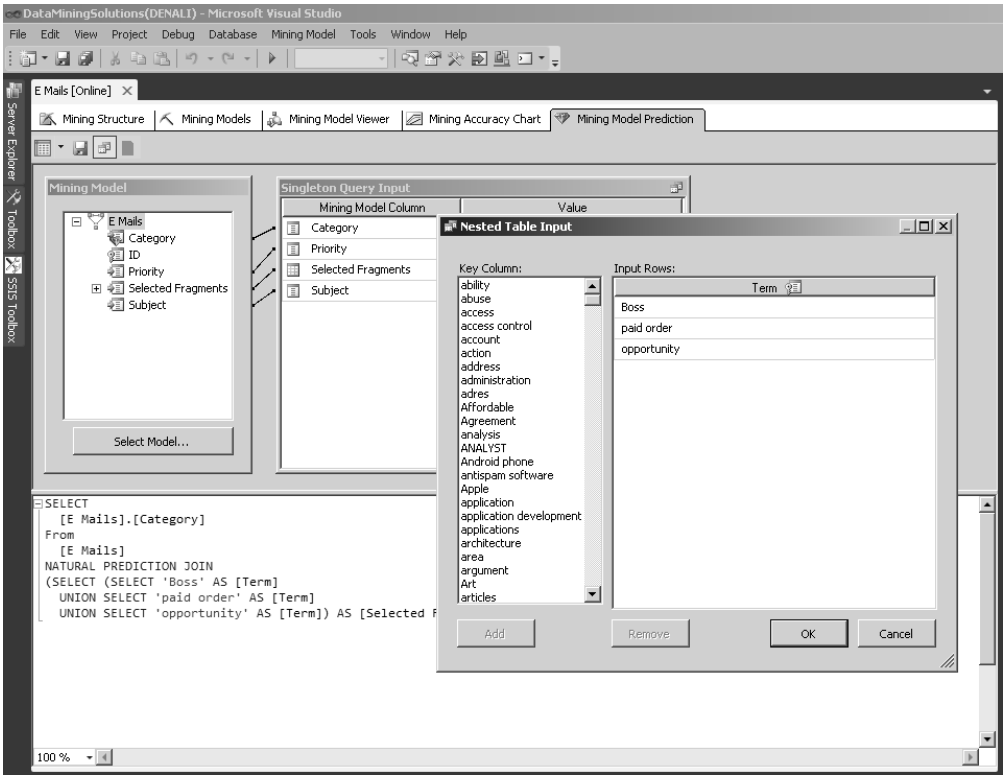
   a) *Automatic feature selection has been applied to model, due to the large number of attributes. Set MAXIMUM_INPUT_ATTRIBUTES and/or MAXIMUM_OUTPUT_ATTRIBUTES to increase the number of attributes considered by the algorithm* — to ostrzeżenie dotyczy zagnieżdżonego atrybutu `Term`.

   b) *Cardinality reduction has been applied on column, Subject of model, E Mails due to the large number of states in that column. Set MAXIMUM_STATES to increase the number of states considered by the algorithm* — to ostrzeżenie dotyczy atrybutu `Subject`.

8. Wyświetl okno właściwości algorytmu i ustaw na `0` wartości parametrów `MAXIMUM_INPUT_ATTRIBUTES` oraz `MAXIMUM_STATES`.

9. Ponownie przetwórz model eksploracji danych i zapoznaj się z jego wynikami.

10. Przejdź na zakładkę *Data Mining Prediction* i wykonaj zapytanie czasu rzeczywistego, oceniające, czy e-mail zawierający wybrane frazy będzie sklasyfikowany jako spam (rysunek 9.7).



Rysunek 9.7. *Business Intelligence Development Studio pozwala wybrać z listy wartości zagnieżdżonego atrybutu te, których chcemy użyć w zapytaniach czasu rzeczywistego*

Skorowidz

A

B

C

D

E

F

G

H

I

J

K

Ł

M

N

O

P

Q

R

S

T

U

W

X

Z

Ź

Poznaj sprawdzone techniki profesjonalnej eksploracji danych!

Eksploracja danych jest najmłodszą i najszybciej rozwijającą się dziedziną informatyki. Łączy zaawansowane algorytmy analizowania danych i znajdowania ukrytych w nich wzorców z klasycznymi technikami statystycznymi, rachunkiem prawdopodobieństwa i technologiami bazodanowymi. Dziedzina ta zyskuje na znaczeniu wraz z lawinowo rosnącą ilością informacji cyfrowych, które każdego dnia na całym świecie są wytwarzane, zapisywane i przeszukiwane przez stale zwiększającą się grupę użytkowników technologii informatycznych. Trzydzieści lat temu świat zrewolucjonizowały komputery PC, a dziś w ogarnięciu tego chaosu pomóc mogą jedynie najskuteczniejsze narzędzia do modelowania i eksploracji danych. *Microsoft SQL Server. Modelowanie i eksploracja danych* to książka, z której analitycy, informatycy i biznesmeni dowiedzą się, jak tworzyć właściwe modele, odpowiednio przygotowywać dane i prawidłowo je eksplorować, a także jak należycie analizować i oceniać otrzymane wyniki przy użyciu narzędzi oferowanych przez oprogramowanie Microsoft SQL Server. Publikacja przybliża zagadnienia związane z poszczególnymi etapami procesu modelowania i eksploracji, prezentując przy tym zastosowanie różnych metod i technik analizy do rozwiązywania praktycznych problemów naukowych i biznesowych.

- Podstawowe techniki i narzędzia wykorzystywane w eksploracji danych
- Instalacja i konfiguracja niezbędnego oprogramowania
- Analiza biznesowa projektu eksploracji danych
- Ocena, przygotowywanie i poprawianie jakości danych
- Przegląd technik eksploracji danych
- Wykorzystywanie serwera SQL w procesie eksploracji danych
- Zasada działania zaawansowanych algorytmów eksploracji danych
- Programowanie predykcyjne

Naucz się wykorzystywać zaawansowane narzędzia do inteligentnej zamiany dużych zbiorów danych w przydatne informacje!

Nr katalogowy: 7481

Księgarnia internetowa
http://helion.pl

Zamówienia telefoniczne:
0 801 339900
0 601 339900

Helion

Sprawdź najnowsze promocje:
http://helion.pl/promocje
Książki najchętniej czytane:
http://helion.pl/bestsellery
Zamów informacje o nowościach:
http://helion.pl/nowosci

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
http://helion.pl

Informatyka w najlepszym wydaniu

helion.pl
księgarnia internetowa

Cena 69,00 zł

ISBN 978-83-246-3440-8

9 788324 634408