Laboratorium 13

Eksploracja danych tekstowych.

Eksploracja danych tekstowych oraz kroki wstępne przetwarzania tekstu zostaną wykonane zarówno w środowisku SQL, jak i za pomocą narzędzia Oracle Data Miner. Wykorzystanie gotowego narzędzia jest preferowane w sytuacji gdy eksploracja danych tekstowych jest ostatnim krokiem procesu odkrywania wiedzy. Jeśli analiza i eksploracja danych tekstowych są częścią większego systemu informatycznego, wówczas konieczne jest zaimplementowanie algorytmów eksploracji tekstu bezpośrednio na poziomie SQL.

1. Uruchom narzędzie iSQLPlus i połącz się z bazą danych.
2. Usuń obiekty pozostałe po poprzednim uruchomieniu algorytmu (konieczne w przypadku wielokrotnego uruchamiania procedury). Konieczne jest usunięcie tabel, indeksów tekstowych i perspektyw.

```
BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE customers_text';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP INDEX customers_text_idx';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/


BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE preference_terms';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE tf_out';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE term_out';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE terms';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/
```

```
BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE text_categories';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE data_input';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE nontxt_attributes';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  CTX_DDL.DROP_PREFERENCE(preference_name => 'preference');
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/
```

3. Do przedstawienia algorytmów przetwarzania tekstu przygotujemy sobie tabelę utworzoną na podstawie przykładowych tabel umieszczonych w schemacie użytkownika SH.

```
CREATE TABLE customers_text AS
SELECT
 a.CUST_ID,
 a.CUST_GENDER,
 EXTRACT(YEAR FROM SYSDATE) - a.CUST_YEAR_OF_BIRTH AS AGE,
 a.CUST_MARITAL_STATUS,
 c.COUNTRY_NAME,
 a.CUST_INCOME_LEVEL,
 b.EDUCATION,
 b.OCCUPATION,
 b.HOUSEHOLD_SIZE,
 b.YRS_RESIDENCE,
 b.AFFINITY_CARD,
 b.BULK_PACK_DISKETTES,
 b.FLAT_PANEL_MONITOR,
 b.HOME_THEATER_PACKAGE,
 b.BOOKKEEPING_APPLICATION,
 b.PRINTER_SUPPLIES,
 b.Y_BOX_GAMES,
 b.OS_DOC_SET_KANJI,
 b.COMMENTS
FROM
 sh.CUSTOMERS a, sh.SUPPLEMENTARY_DEMOGRAPHICS b, sh.COUNTRIES c
WHERE
 a.CUST_ID = b.CUST_ID
 AND a.COUNTRY_ID  = c.COUNTRY_ID
 AND a.CUST_ID BETWEEN 101501 AND 103000;
```

4. Kolejny wymagany krok to utworzenie indeksu tekstowego na kolumnie zawierającej dane tekstowe.

```
CREATE INDEX customers_text_idx ON customers_text(comments)
  INDEXTYPE IS ctxsys.context PARAMETERS('nopopulate')
/
```

5. Algorytmy eksploracji tekstu bazują na podstawowej czynności związanej z przetwarzaniem tekstu, tj. na ekstrakcji termów. Automatyczna ekstrakcja termów wymaga konfiguracji w postaci określenia preferencji co do narzędzia identyfikacji termów.

```
BEGIN
  CTX_DDL.CREATE_PREFERENCE('preference', 'SVM_CLASSIFIER');
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/
```

6. Kolekcje związanych ze sobą termów będą reprezentowane w postaci tzw. kategorii. Konieczne jest utworzenie tabeli do przechowywania kategorii odkrytych w przetwarzanym tekście.

```
CREATE TABLE text_categories (id NUMBER, cat NUMBER)
/
```

7. Kolejny krok to uruchomienie procedury znajdowania termów w tekście.

```
-- odczytanie termow na podstawie zdefiniowanej preferencji
--         1. term_out   - termy pochodzace z preferencji
--         2. preference_terms - definicje termow
--
--         Parametrami metody sa
--         -- CUSTOMERS_TEXT_IDX - nazwa indeksu typu TEXT
--         -- CUST_ID            - klucz podstawowy tabeli
--         -- text_categories    - tabela przechowujaca kategorie
--         -- id                 - klucz podstawowy tabeli z kategoriami
--         -- cat                - identyfikator kategorii
--         -- preferencje_terms  - definicje termow
--         -- preference - nazwa preferencji
--
CREATE TABLE term_out AS
SELECT *
  FROM TABLE(ctxsys.drvodm.feature_prep (
               'CUSTOMERS_TEXT_IDX',
               'CUST_ID',
               'text_categories',
               'id',
               'cat',
               'preference_terms',
               'preference'));
```

8. Termy odkryte w przetwarzanym tekście są przechowywane w strukturach wewnętrznych bazy danych w postaci nieczytelnej dla aplikacji zewnętrznych. Aby można było je wykorzystać, np. w algorytmach eksploracji danych, muszą być przetransformowane do standardowej postaci tabeli zagnieżdżonej. Poniżej znajduje się zapytanie, które odtwarza część wykorzystywanej wcześniej tabeli CUSTOMERS_TEXT (wybiera wszystkie atrybuty nie-tekstowe), a następnie „skleja” te atrybuty z tabelą zagnieżdżoną właściwą dla danego wiersza.

```
CREATE TABLE terms AS
SELECT A.sequence_id, B.text, A.value
FROM term_out A,
     TABLE(ctxsys.drvodm.feature_explain('preference_terms')) B
WHERE A.attribute_id = B.id;

column text format a45
SELECT *
FROM ( SELECT sequence_id,text,value
       FROM terms
       ORDER BY sequence_id, text )
WHERE ROWNUM < 10;

CREATE TABLE nontxt_attributes AS
SELECT
  CUST_ID,
  CUST_FIRST_NAME,
  CUST_LAST_NAME,
  CUST_GENDER,
  CUST_YEAR_OF_BIRTH,
  CUST_MARITAL_STATUS,
  CUST_STREET_ADDRESS,
  CUST_POSTAL_CODE,
  CUST_CITY,
  CUST_CITY_ID,
  CUST_STATE_PROVINCE,
  CUST_STATE_PROVINCE_ID,
  COUNTRY_ID,
  CUST_MAIN_PHONE_NUMBER,
  CUST_INCOME_LEVEL,
  CUST_CREDIT_LIMIT,
  CUST_EMAIL,
  CUST_TOTAL,
  CUST_TOTAL_ID,
  CUST_SRC_ID,
  CUST_VALID
FROM sh.CUSTOMERS
WHERE CUST_ID BETWEEN 101501 AND 103000;

CREATE TABLE data_input
  NESTED TABLE txt_terms STORE AS txt_terms AS
SELECT N.*,
  CAST(MULTISET(
    SELECT DM_Nested_Numerical (T.text, T.value)
    FROM terms T
    WHERE N.cust_id = T.sequence_id) AS DM_Nested_Numericals) txt_terms
FROM nontxt_attributes N
WHERE cust_id < 102000;
```

9. Porównaj postać tego samego wiersza (klient nr 101555) w danych oryginalnych (przed wstępnym przetworzeniem) i po przygotowaniu do przetwarzania tekstowego.

```
SELECT * FROM sh.supplementary_demographics WHERE cust_id = 101555;

SELECT * FROM data_input WHERE cust_id = 101555;
```

10. Tak przygotowane dane można poddać wielu algorytmom eksploracji, np. odkrywaniu cech, klasyfikacji, grupowaniu. Poniżej przedstawiamy przykład wykorzystania danych tekstowych do zbudowania klasyfikatora za pomocą algorytmu SVM.

11. Usuń obiekty pozostałe po poprzednich uruchomieniach algorytmu.

```
BEGIN
  DBMS_DATA_MINING.DROP_MODEL('Text_SVM_Model');
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE settings';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE normalization';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  EXECUTE IMMEDIATE 'DROP VIEW input_data';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/
```

12. Utwórz tabelę do przechowywania ustawień algorytmu klasyfikacji i wypełnij tabelę parametrami inicjalizacyjnymi algorytmu.

```
CREATE TABLE settings (
  setting_name  VARCHAR2(30),
  setting_value VARCHAR2 (30));

BEGIN
  INSERT INTO settings VALUES (dbms_data_mining.algo_name,
    dbms_data_mining.algo_support_vector_machines);
  INSERT INTO settings VALUES (dbms_data_mining.svms_conv_tolerance,0.01);
  INSERT INTO settings VALUES (dbms_data_mining.svms_kernel_function,
    dbms_data_mining.svms_linear);
  COMMIT;
END;
/
```

13. Wykonaj normalizację atrybutów numerycznych (krok wymagany przez algorytm SVM)

```
BEGIN
  -- zbudowanie tabeli do przechowywania parametrow normalizacji
  DBMS_DATA_MINING_TRANSFORM.CREATE_NORM_LIN (
    norm_table_name => 'normalization');

  -- normalizacja za pomoca metody Min-Max
  DBMS_DATA_MINING_TRANSFORM.INSERT_NORM_LIN_MINMAX (
    norm_table_name => 'normalization',
    data_table_name => 'mining_build_nested_text',
    exclude_list    => DBMS_DATA_MINING_TRANSFORM.COLUMN_LIST (
                       'CUST_ID',
                       'AFFINITY_CARD',
                       'BULK_PACK_DISKETTES',
                       'FLAT_PANEL_MONITOR',
```

```
                        'HOME_THEATER_PACKAGE',
                        'BOOKKEEPING_APPLICATION',
                        'PRINTER_SUPPLIES',
                        'Y_BOX_GAMES',
                        'OS_DOC_SET_KANJI',
                        'COMMENTS'),
    round_num       => 0
  );

  -- utworzenie perspektywy z danymi po normalizacji
  DBMS_DATA_MINING_TRANSFORM.XFORM_NORM_LIN (
    norm_table_name => 'normalization',
    data_table_name => 'mining_build_nested_text',
    xform_view_name => 'input_data');
END;
/
```

14. Zbuduj model klasyfikatora

```
BEGIN
  DBMS_DATA_MINING.CREATE_MODEL(
    model_name          => 'Text_SVM_Model',
    mining_function     => dbms_data_mining.classification,
    data_table_name     => 'input_data',
    case_id_column_name => 'cust_id',
    target_column_name  => 'affinity_card',
    settings_table_name => 'settings');
END;
/
```

15. Wyświetl sygnaturę modelu.

```
column attribute_type format a20
SELECT *
  FROM TABLE(DBMS_DATA_MINING.GET_MODEL_SIGNATURE('Text_SVM_Model'))
ORDER BY attribute_name;
```

16. Przygotuj dane potrzebne do przetestowania klasyfikatora. Konieczne jest, aby dane testowe poddać tym samym transformacjom co dane, na podstawie których został zbudowany klasyfikator.

```
BEGIN
  EXECUTE IMMEDIATE 'DROP VIEW apply_data';
EXCEPTION
  WHEN OTHERS THEN NULL;
END;
/

BEGIN
  -- utworzenie perspektywy pokazujacej znormalizowane dane testowe
  DBMS_DATA_MINING_TRANSFORM.XFORM_NORM_LIN (
    norm_table_name => 'normalization',
    data_table_name => 'mining_apply_nested_text',
    xform_view_name => 'apply_data');
END;
/
```

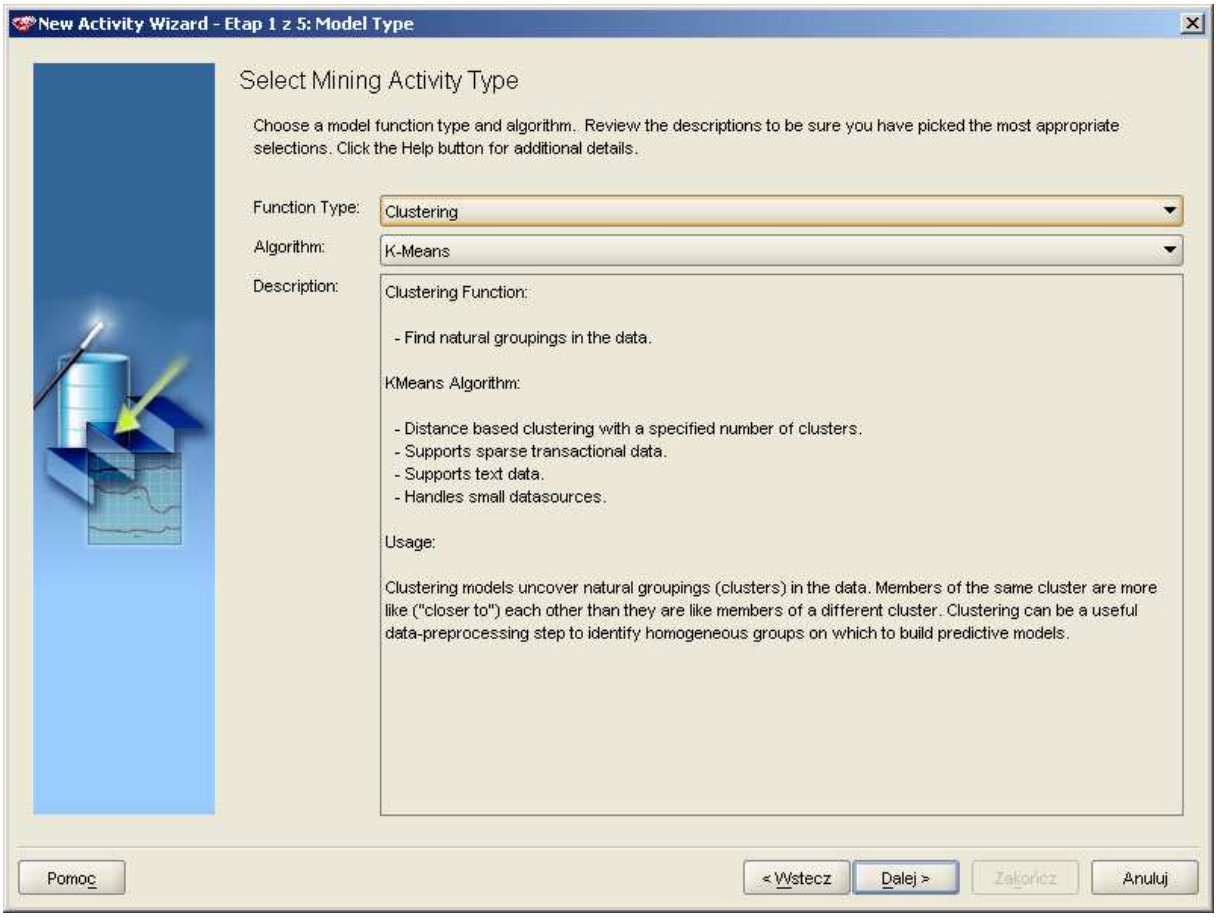
17. Wykonaj przykładowe zapytania wykorzystujące stworzony model.

```
-- 1. Znajdz 10 klientow charakteryzujacych sie
-- najwyzszym prawdopodobieństwem skorzystania z karty kredytowej
-- (wykorzystujac komentarze tekstowe)
--
```

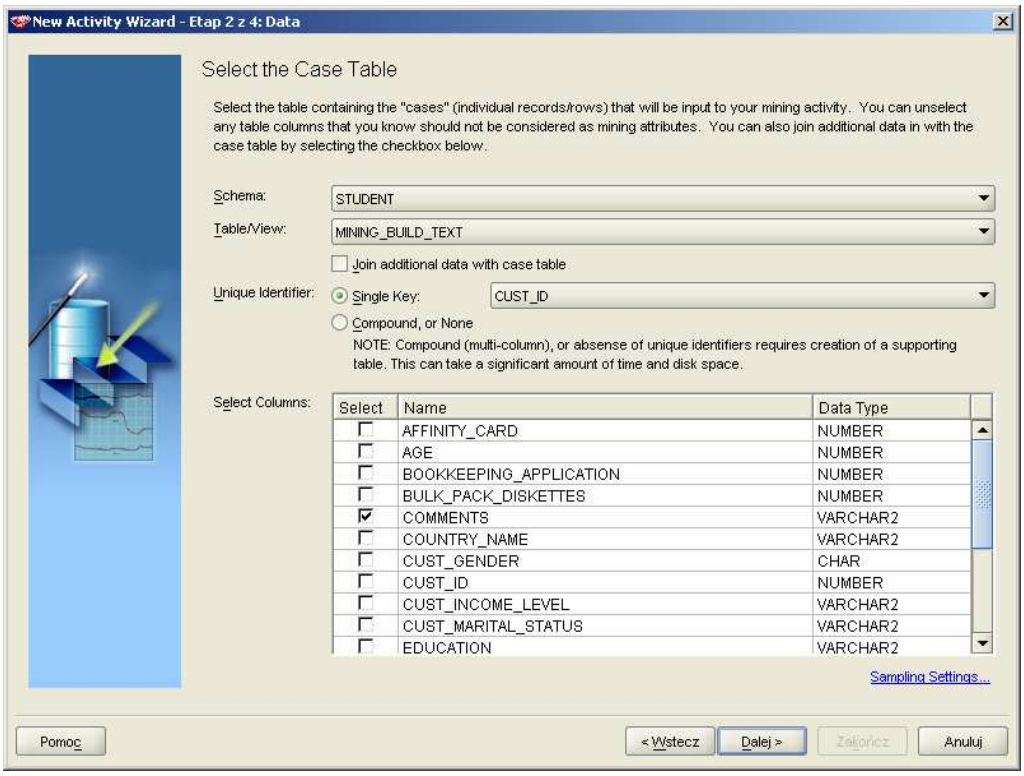
```
SELECT cust_id
FROM (SELECT cust_id
      FROM apply_data
      ORDER BY PREDICTION_PROBABILITY(Text_SVM_Model, 1 USING *) DESC, 1)
WHERE ROWNUM < 11;

-- 2. Wyswietl sredni wiek (z podzialem na plcie) klientow ktorzy
-- prawdopodobnie kupia oferowana karte kredytowa
--
SELECT cust_gender, COUNT(*) AS cnt,
  ROUND(AVG(age * N.scale + N.shift)) AS avg_age
FROM apply_data, normalization N
WHERE PREDICTION(Text_SVM_Model USING *) = 1
  AND N.col = 'AGE'
GROUP BY cust_gender
ORDER BY cust_gender;
```

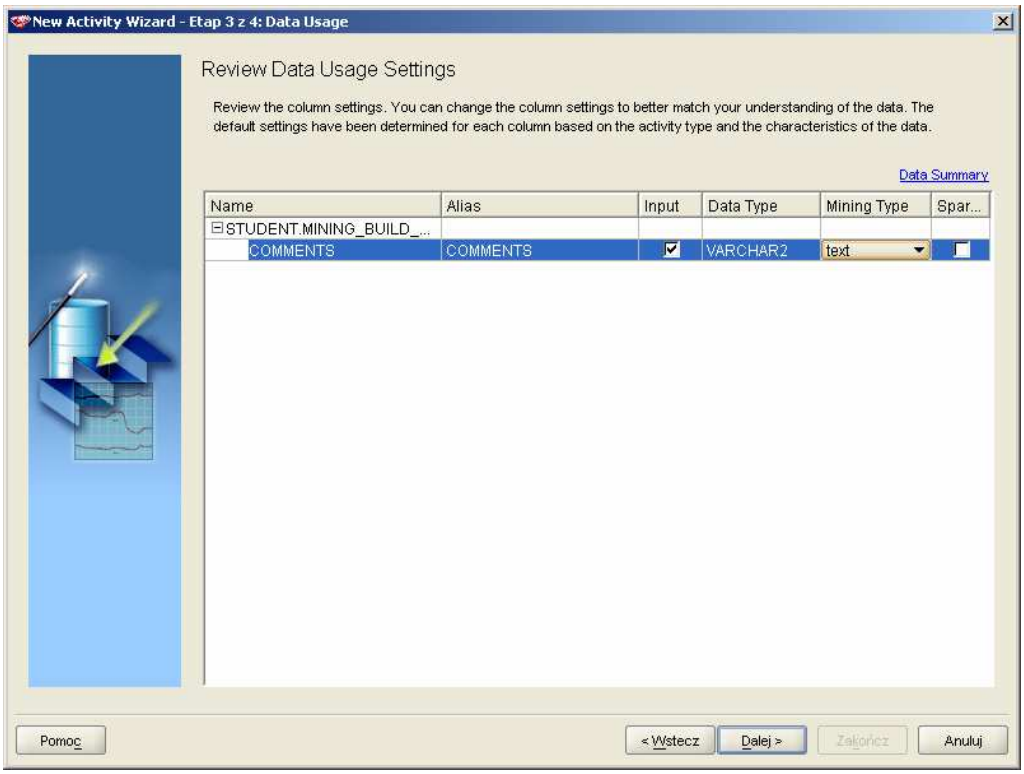
18. W drugiej części ćwiczenia zobaczymy, w jaki sposób można wykorzystać narzędzie Oracle Data Mining do eksploracji danych tekstowych.
19. Uruchom narzędzie Oracle Data Mining i połącz się z bazą danych.
20. Z menu głównego wybierz Activity→Build. Na ekranie powitalnym kliknij przycisk Dalej>.
21. Z listy Function Type wybierz Clustering. Rozwiń listę Algorithm i wybierz z niej algorytm k-Means. Kliknij przycisk Dalej>.



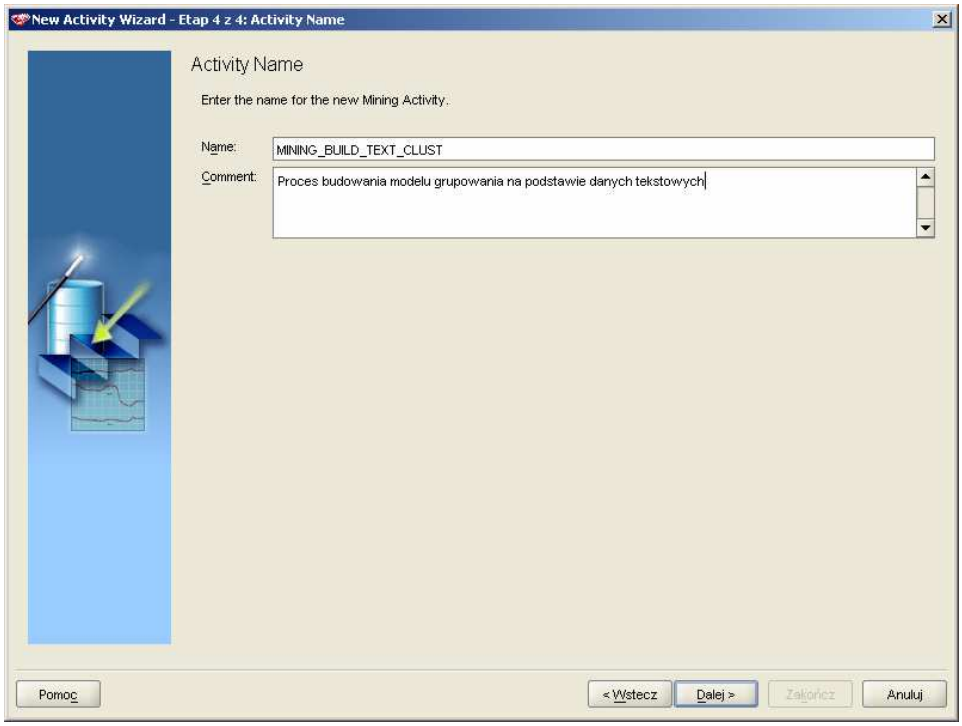
22. Wskaż schemat `STUDENT` i tabelę `MINING_ BUILD_TEXT` jako źródło danych do eksploracji. Jako klucz podstawowy wskaż atrybut `CUST_ID`. Wyłącz z eksploracji wszystkie atrybuty za wyjątkiem atrybutu `COMMENTS`. Kliknij przycisk Dalej>.



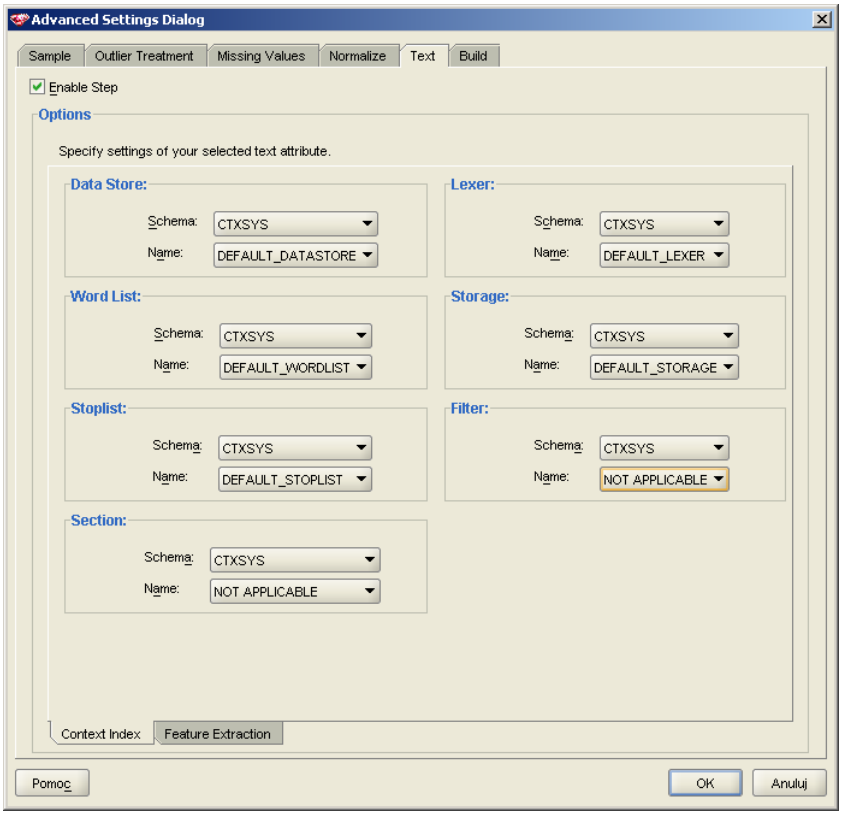
23. Zmień typ eksploracyjny atrybutu `COMMENTS` (kolumna `Mining Type`) na text. Kliknij przycisk Dalej>.



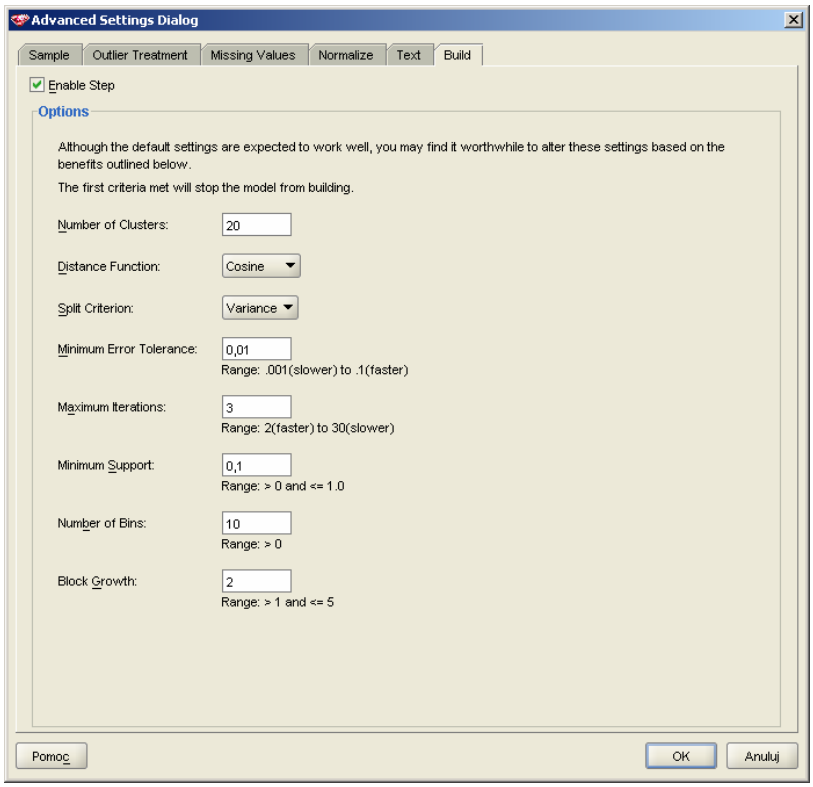
24. Podaj nazwę i krótki opis procesu eksploracji. Kliknij przycisk Dalej>.



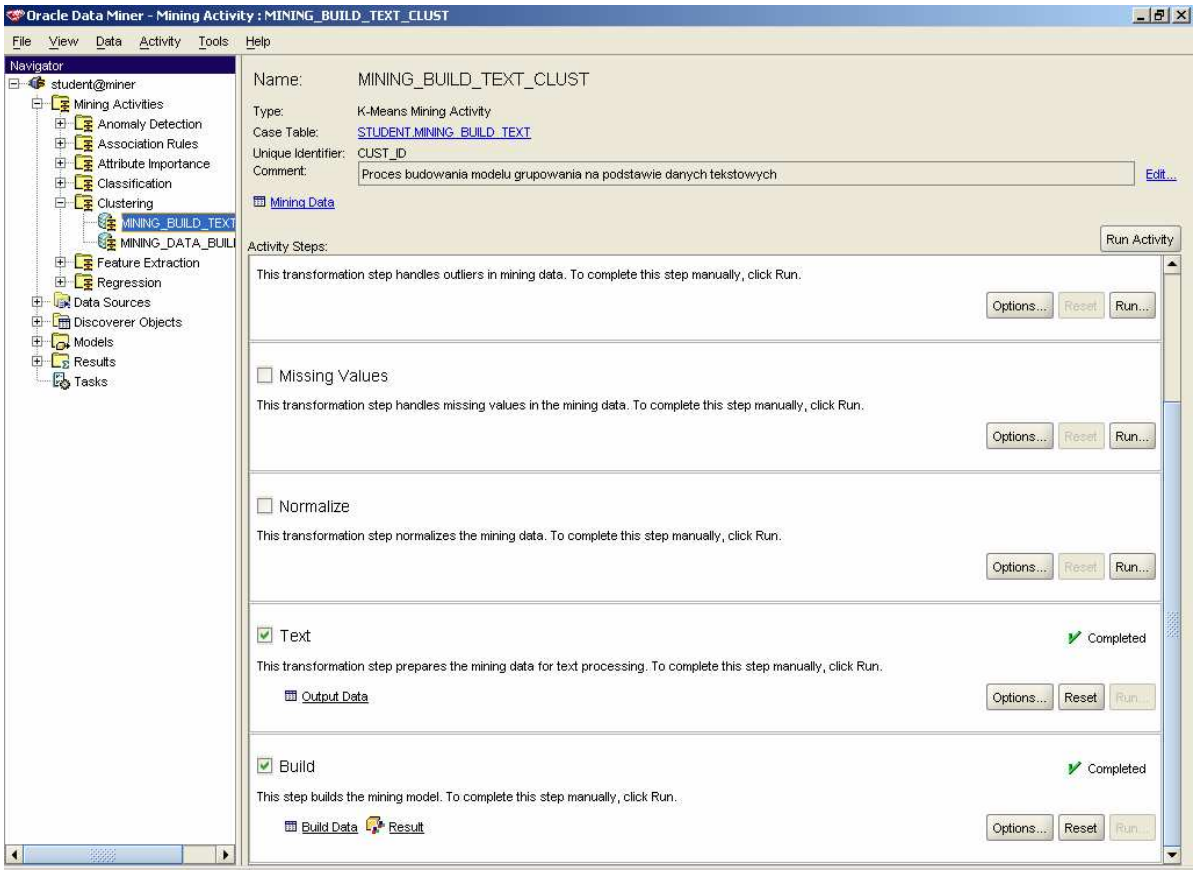
25. Kliknij przycisk Advanced Settings. Upewnij się, że na zakładce Sample opcja próbkowania jest wyłączona (pole wyboru Enable Step jest odznaczone). Analogicznie upewnij się, że wyłączone są kroki Outlier Treatement, Missing Values i Normalize. Przejdź na zakładkę Text i upewnij się, że krok jest włączony.



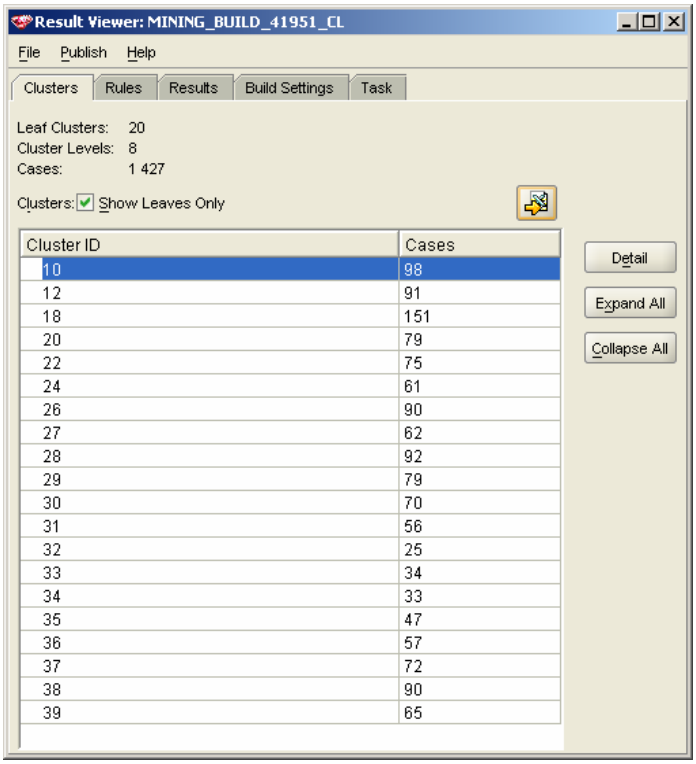
26. Przejdź na zakładkę Build i zmień liczbę grup na 20. Upewnij się, że wybrano cosinusową miarę odległości. Kliknij przycisk OK.



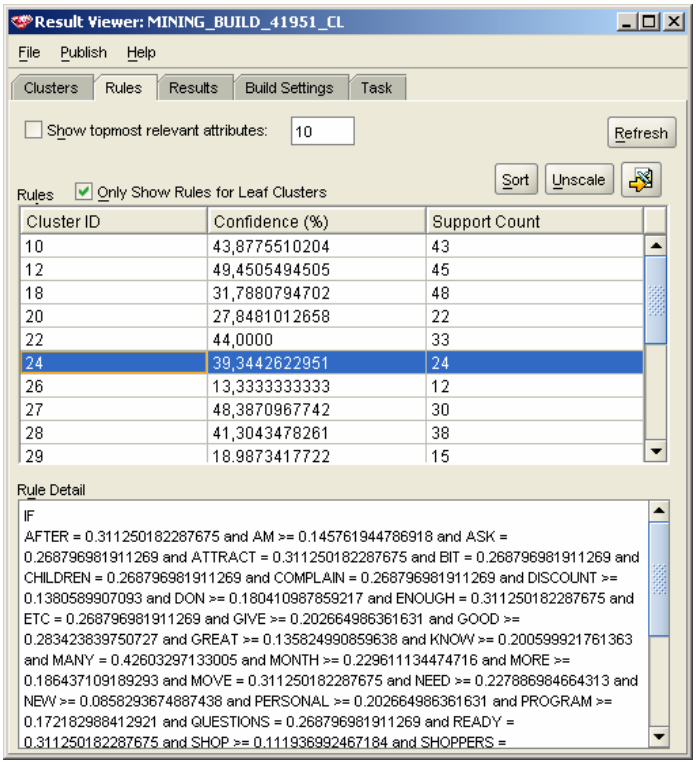
27. Upewnij się, że opcja Run upon finish jest włączona. Kliknij przycisk Zakończ.



28. Kliknij na odnośnik Result w bloku Build. Zaznacz opcję Show Leaves Only.



29. Przejdź na zakładkę Rules. Zaznacz opcję Only Show Rules for Leaf Clusters. Wybierz dowolny klaster i przeanalizuj wystąpienia słów, które trafiają do wybranego klastra. Na ile łatwy, Twoim zdaniem, jest uzyskany wynik do ręcznej weryfikacji i analizy?



Ćwiczenie samodzielne

W Twoim schemacie znajduje się tabela `REUTERS` zawierająca 1000 depeszy agencji informacyjnej Reutersa. Przygotuj tabelę do eksploracji:

1. zbuduj na kolumnie `MESSAGE` indeks tekstowy typu CTXSYS.CONTEXT
2. stwórz preferencję dla ekstrakcji termów
3. zbuduj dodatkową tabelę do przechowywania kategorii
4. zidentyfikuj termy pojawiające się w tekście
5. zbuduj perspektywę zawierającą zwykły atrybut `TITLE` oraz tabelę zagnieżdżoną z termami znalezionymi w każdej depeszy

Do tak przygotowanych danych zastosuj algorytm odkrywania cech i znajdź 10 najczęściej pojawiających się tematów w depeszach.