

Systemy rozmyte

Dokumentacja projektu “Ocena zamówienia”

Daniel Partyka

Rzeszów

2022

1. Cel i zakres projektu.

Celem projektu było stworzenie rozmytego systemu, który miał na celu wyznaczanie końcowej oceny sprzedającego dany towar i wystawienie mu opinii (wyrażonej w liczbie 0-10). Projekt został wykonany w języku Python z wykorzystaniem biblioteki skfuzzy. Zakres projektu obejmował stworzenie kilku zmiennych wejściowych oraz jedną zmienną wyjściową, stworzenie odpowiednich funkcji przynależności do zmiennych wejściowych i wyjściowych, zdefiniowanie odpowiednich reguł przynależności, przeprowadzenie symulacji działania i wygenerowanie wyników w postaci graficznej.

2. Logika Rozmyta

Logika rozmyta – jedna z logik wielowartościowych, stanowi uogólnienie klasycznej dwuwartościowej logiki. Została zaproponowana przez Lotfi Zadeha, jest ściśle powiązana z jego teorią zbiorów rozmytych. W logice rozmytej między stanem 0 (fałsz) a stanem 1 (prawda) rozciąga się szereg wartości pośrednich, które określają stopień przynależności elementu do zbioru.

Na systemy rozmyte składają się te techniki i metody, które służą do obrazowania informacji nieprecyzyjnych, nieokreślonych bądź niekonkretnych. Pozwalają one opisywać zjawiska o charakterze wieloznacznym, których nie jest w stanie ująć teoria klasyczna i logika dwuwartościowa.

Charakteryzują się tym, że wiedza jest przetwarzana w postaci symbolicznej i zapisywana w postaci rozmytych reguł.

Systemy rozmyte znajdują zastosowanie tam, gdzie nie posiadamy wystarczającej wiedzy o modelu matematycznym rządzącym danym zjawiskiem oraz tam gdzie odtworzenie tegoż modelu staje się nieopłacalne lub nawet niemożliwe. Tak więc możemy je spotkać w bazach danych, sterowaniu oraz dziedzinach zajmujących się przetwarzaniem języka naturalnego.

3. Opis projektu.

W projekcie zostały wykorzystane 4 zmienne wejściowe: “Zgodność z opisem”, “Czas wysyłki”, “Koszt wysyłki” i “Kontakt ze sprzedawcą”. Każda ze zmiennych przyjmuje

wartości z przedziałów i wynoszą one odpowiednio: [0-5], [0-30], [0-100] i [0-5]. Zgodność z opisem i kontakt ze sprzedawcą ocenia się za pomocą wartości od 0 do 5. Czas wysyłki został ustalony na przedział od 0 do 30 dni, ponieważ większość sklepów w Polsce ma czas do 30 dni na wysyłkę zamówienia. (Oczywiście sklepy zagraniczne takie jak np. Aliexpress mają dłuższy czas oczekiwania, lecz w tym przypadku zostało to uproszczone do wartości 30 dni). Ostatnią zmienną wejściową koszt wysyłki, którego zakres wynosi od 0 do 100. W tym przypadku też zostało to uproszczone, gdyż często wartość wysyłki zależy od rodzaju wysyłanego towaru, lecz z czasem przyjęto, że cena za wysyłkę powyżej 15 zł wpływa na jakość całego zamówienia.

Jako zmienną wyjściową (jedyną) ustalono "Ocena zamówienia". Zakres tej zmiennej wynosi od 0 do 10 i jest ona generowana na podstawie wnioskowania rozmytego.

```
# pierwsza zmienna wejsciowa = zgodnosc z opisem
compliance_with_the_description = ctrl.Antecedent(np.arange(0, 6), 'compliance_with_the_description (rate 0-5)')

# druga zmienna wejsciowa = czas wysylki (maksymalne 30 dni, ustalamy ze osoba sprzedaje produkt w tym samym kraju)
shipping_time = ctrl.Antecedent(np.arange(0, 31), 'shipping_time (days 0-30)')

# trzecia zmienna wejsciowa = koszty wysylki
shipping_cost = ctrl.Antecedent(np.arange(0, 101), 'shipping_cost (cost 0-100)')

# czwarta zmienna wyjsciowa = kontakt ze sprzedawca
contact_with_the_seller = ctrl.Antecedent(np.arange(0, 6), 'contact_with_the_seller (rate 0-5)')

# zmienna wyjsciowa = ocena zamowienia
evaluation_of_the_order = ctrl.Consequent(np.arange(0, 11), 'evaluation_of_the_order (rate 0-10)')
```

Rys.1 Zmienne wejściowe i zmienna wyjściowa

Każda zmienna wejściowa i zmienna wyjściowa mają przyporządkowane odpowiednie trapezoidalne funkcje przynależności, które na podstawie zadanych wartości generują wartości lingwistyczne. Zmienne wyjściowe mają po 3 wartości lingwistyczne. Przykładowo "Zgodność z opisem" posiada cechy: "LOW", "MEDIUM", "HIGH". Natomiast czas wysyłki "FAST", "MEDIUM", "LONG". Wartości te są ustawione przez subiektywne odczucia twórcy programu, lecz zachowują sens. Zmienna wyjściowa posiada 4 wartości lingwistyczne i są one następujące: "UNACCEPTABLE", "ACCEPTABLE", "GOOD" i "VERY_GOOD".

```

# funkcje przynależności do pierwszej zmiennej wejściowej (zgodność z opisem)
compliance_with_the_description['LOW'] = fuzz.trapmf(compliance_with_the_description.universe, [0, 0, 1, 2])
compliance_with_the_description['MEDIUM'] = fuzz.trimf(compliance_with_the_description.universe, [1, 2, 4])
compliance_with_the_description['HIGH'] = fuzz.trapmf(compliance_with_the_description.universe, [2, 4, 5, 5])
compliance_with_the_description.view()

# funkcje przynależności do drugiej zmiennej wejściowej (czas wysyłki)
shipping_time['FAST'] = fuzz.trapmf(shipping_time.universe, [0, 0, 1, 2])
shipping_time['MEDIUM'] = fuzz.trimf(shipping_time.universe, [1, 3, 5])
shipping_time['LONG'] = fuzz.trapmf(shipping_time.universe, [3, 5, 15, 30])
shipping_time.view()

# funkcje przynależności do trzeciej zmiennej wejściowej (cena wysyłki)
shipping_cost['LOW'] = fuzz.trapmf(shipping_cost.universe, [0, 0, 6, 12])
shipping_cost['MEDIUM'] = fuzz.trimf(shipping_cost.universe, [6, 12, 20])
shipping_cost['HIGH'] = fuzz.trapmf(shipping_cost.universe, [20, 50, 75, 100])
shipping_cost.view()

# funkcje przynależności do czwartej zmiennej wejściowej (kontakt ze sprzedawcą)
contact_with_the_seller['POOR'] = fuzz.trapmf(contact_with_the_seller.universe, [0, 0, 1, 2])
contact_with_the_seller['MEDIUM'] = fuzz.trimf(contact_with_the_seller.universe, [1, 2, 3])
contact_with_the_seller['GOOD'] = fuzz.trapmf(contact_with_the_seller.universe, [2, 3, 5, 5])
contact_with_the_seller.view()

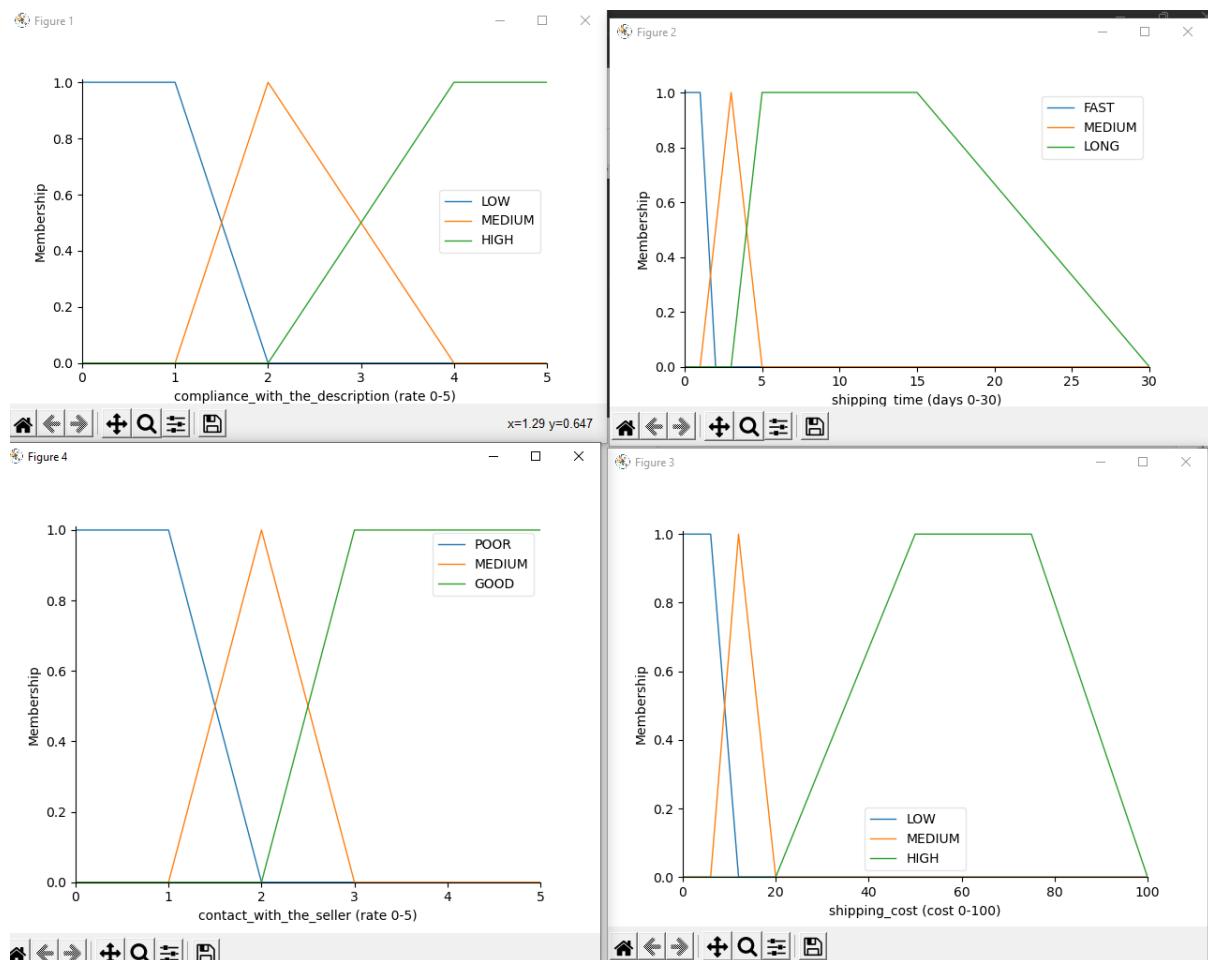
# funkcje przynależności do zmiennej wyjściowej
evaluation_of_the_order['UNACCEPTABLE'] = fuzz.trapmf(evaluation_of_the_order.universe, [0, 0, 2, 4])
evaluation_of_the_order['ACCEPTABLE'] = fuzz.trimf(evaluation_of_the_order.universe, [2, 4, 6])
evaluation_of_the_order['GOOD'] = fuzz.trimf(evaluation_of_the_order.universe, [4, 6, 8])
evaluation_of_the_order['VERY_GOOD'] = fuzz.trapmf(evaluation_of_the_order.universe, [7, 8, 10, 10])
evaluation_of_the_order.view()

```

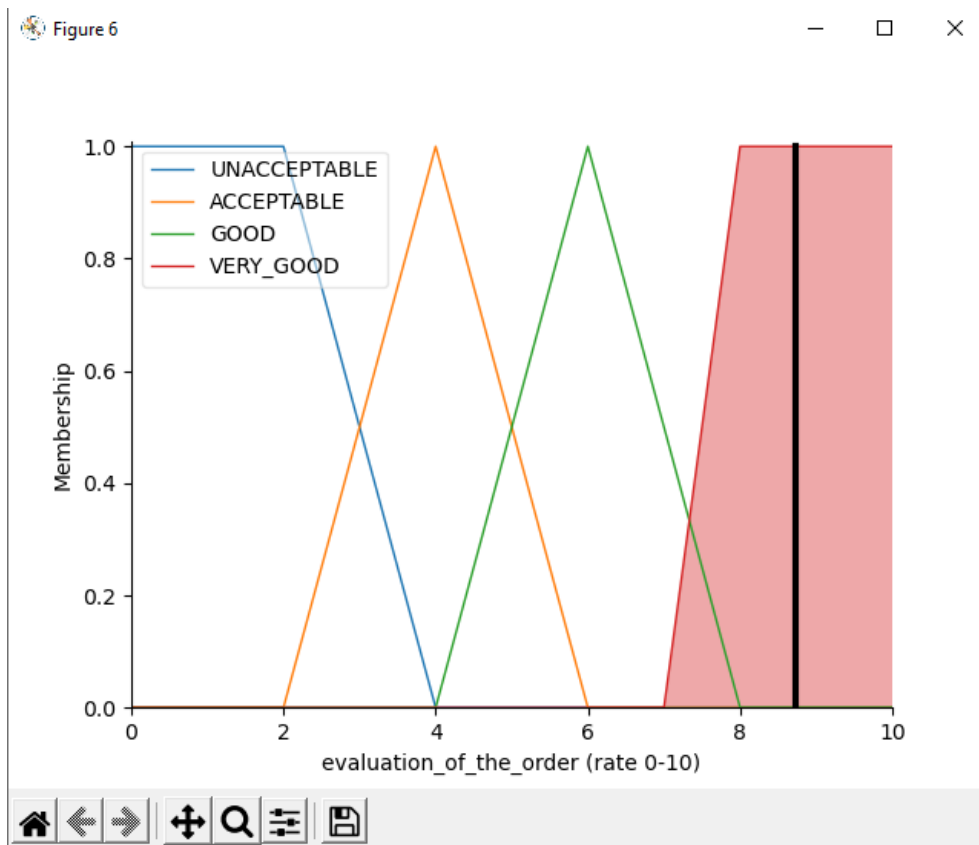
Rys. 2 Funkcje przynależności do zmiennych wejściowych i zmiennej wyjściowej

Cechy dla funkcji wyjściowych są nadawane poprzez zastosowanie wnioskowania rozmytego na podstawie zadanych reguł decyzyjnych. W przypadku programu "ocena_zamówienia" zostało zadeklarowanych 177 reguł decyzyjnych. Liczba ta była spowodowana aż 4 zmiennymi wejściowymi, przez co należało pokryć wszystkie warianty zmiennych lingwistycznych. (być może część tych reguł się powtarza, lecz bardzo ciężko by było znaleźć takie powtórzenia).

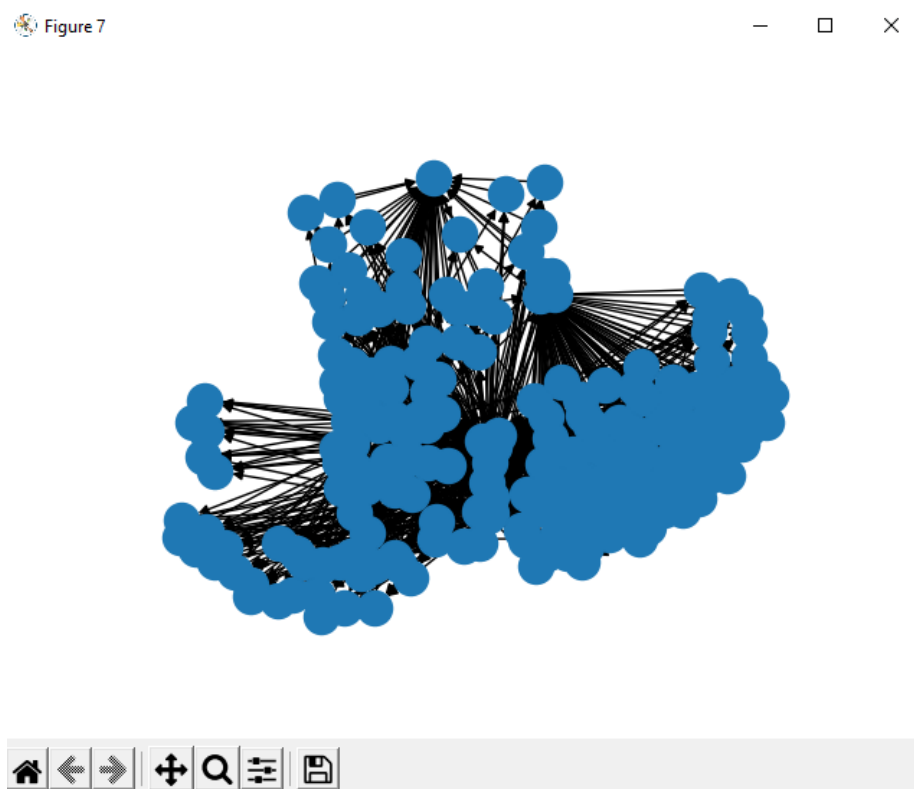
Po uruchomieniu programu programu pojawi się 7 wykresów przedstawiających funkcje przynależności zmiennych wejściowych i zmiennej wyjściowej, wnioskowanie rozmyte i rozkład reguł za pomocą grafów a także wartość liczba



Rys. 3 Graficzne przedstawienie funkcji przynależności do zmiennych wejściowych



Rys. 4 Graficzne przedstawienie funkcji przynależności do zmiennych wyjściowych



Rys. 5 Rozkład reguł przedstawiony w postaci grafów

4. Wnioski Końcowe

W ramach projektu został zrealizowany projekt realizujący ocenę zamówień na podstawie podanych wcześniej parametrów. System ten jednak nie jest w pełni autonomiczny, ponieważ polega on zasadniczo na ludzkiej inteligencji i subiektywności w podawaniu danych wejściowych. System ten mógłby być użyty na przykład w automatycznej ocenie jakiś ankiet.