Filipa Popova

smartfit

Table of Contents

[smartfit 0](#_Toc201099605)

[Въведение 2](#_Toc201099606)

[Избор на алгоритъм 2](#_Toc201099607)

[**✅ RandomForestClassifier** 2](#_Toc201099608)

[**📌 Причини за избора:** 2](#_Toc201099609)

[Имплементация на модела 3](#_Toc201099610)

[Данни 6](#_Toc201099611)

[Оценка на ефективността 8](#_Toc201099612)

[Избор на признаци 9](#_Toc201099613)

[Интеграция в уеб приложение 10](#_Toc201099614)

# Въведение

В съвременната ера на онлайн търговията, потребителите все по-често избират да пазаруват дрехи през интернет платформи. Въпреки удобството, един от основните проблеми остава **трудността при избора на правилен размер**, което води до чести връщания, неудовлетвореност и допълнителни разходи за търговците.

Проектът **SMARTFIT** цели да адресира този проблем чрез изграждането на **интелигентна система за препоръчване на подходящ размер дреха**, базирана на индивидуалните мерки и телосложение на потребителя, както и на характеристиките на самата дреха. Чрез използване на алгоритми за машинно обучение и внимателно проектирана архитектура, системата предоставя **персонализирани размерни препоръки**, които повишават точността при онлайн пазаруване и минимизират риска от грешки.

Проектът е реализиран като **уеб базирано приложение** с различни потребителски роли (краен потребител, администратор, търговец) и включва събиране на данни, анализ чрез ML модел, както и визуално представяне на препоръките. Използвани са съвременни технологии като Python, Flask, SQLite и React. Системата позволява гъвкаво надграждане и внедряване в реални електронни магазини.

# Избор на алгоритъм

С оглед на целта на проекта — предоставяне на надеждни препоръки за размер на дрехи въз основа на потребителски и продуктов профил — изборът на подходящ модел за машинно обучение бе от критично значение. След тестване и анализ на няколко алгоритъма, включително логистична регресия, K-близки съседи и decision trees, се избра:

✅ RandomForestClassifier

📌 Причини за избора:

**Поддръжка на числови и категориални данни** – което е от решаващо значение при работа със смесени характеристики като пол, материя и размери.

**Вградена устойчивост на преобучаване (overfitting)** чрез използване на множество decision дървета.

**Възможност за оценка на важността на характеристиките**, което подпомага обяснимостта на препоръките.

**Стабилна и предсказуема точност**, особено при по-големи и хетерогенни тренировъчни множества.

# Имплементация на модела

**3.1. Детайли за имплементацията**

Моделът е създаден с цел прогнозиране на най-подходящия размер дреха въз основа на телесни и някои допълнителни характеристики. Използван е **Random Forest Classifier**, който предлага баланс между точност, устойчивост срещу overfitting и интерпретируемост.

**Основни стъпки:**

1. **Зареждане на данните** – Данните се прочитат от CSV файл и преминават през етап на предварителна обработка.
2. **Предварителна обработка**:
   * Категориалните колони (напр. gender, body\_type, material) се трансформират чрез **OneHotEncoding**.
   * Числовите колони се скалират чрез **StandardScaler**.
3. **Разделяне на данните** – Множеството се разделя на **обучаваща** и **тестваща** извадка.
4. **Трениране на модела** – Random Forest моделът се обучава върху предварително обработените данни.
5. **Оценка на модела** – Изчисляват се точност (accuracy) и **Brier Score**, за да се провери както коректността, така и калибрацията на прогнозите.
6. **Запазване на модела** – Обученият модел и трансформерите се запазват във файл (model.pkl) чрез joblib.

**3.2. Използвани библиотеки**

|  |  |
| --- | --- |
| Библиотека | Роля |
| pandas | Зареждане и обработка на данни |
| numpy | Работа с числови стойности |
| scikit-learn | Моделиране (RandomForestClassifier), preprocessing (OneHotEncoder, StandardScaler), оценка (accuracy, Brier score) |
| joblib | Сериализация на модела и трансформерите |
| logging | Регистриране на ключови съобщения и метрики |

**3.3. Диаграма на имплементацията**

Диаграмата по-долу показва цялостния поток на обработка и моделиране:A screenshot of a computer

Description automatically generated

A diagram of a diagram

Description automatically generated

# Данни

**4.1. Източник на данните**

Данните са събрани с цел изграждане на модел за препоръка на подходящ размер дреха въз основа на физическите характеристики на потребителя и спецификата на облеклото. Датасетът е съставен от реалистични стойности, базирани на типични размери и кройки, използвани в модната индустрия.

📌 *Данните са записани в CSV формат и съдържат както числови, така и категориални променливи.*

**4.2. Предварителна обработка**

За да бъде моделът ефективен и точен, над суровите данни са приложени няколко етапа на предварителна обработка:

* **Категориални променливи** (gender, body\_type, material, garment\_type) са трансформирани чрез **OneHotEncoding**, така че да бъдат използваеми от машинния модел.
* **Числови променливи** (height, weight, waist, chest, garment\_width) са скалирани със **StandardScaler**, за да се уеднаквят техните мащаби.
* Целевата променлива е size – тя е категориална стойност, представляваща препоръчания размер на дрехата (напр. XS, S, M, L, XL).

**4.3. Структура на данните**

Следните променливи се използват като вход към модела:

|  |  |  |
| --- | --- | --- |
| Колона | Тип | Описание |
| height | Числова | Височина на потребителя в см |
| weight | Числова | Тегло на потребителя в кг |
| waist | Числова | Обиколка на талията в см |
| chest | Числова | Обиколка на гърдите в см |
| gender | Категориална | Пол (напр. male, female) |
| body\_type | Категориална | Телосложение (напр. slim, average, large) |
| material | Категориална | Материя на дрехата (elastic, non-elastic) |
| garment\_type | Категориална | Тип дреха (t-shirt, pants и др.) |
| garment\_width | Числова | Ширина на дрехата в см |
| size | Категориална (целева) | Размер на дрехата (XS, S, M, и т.н.) |

# Оценка на ефективността

**Използвани метрики**

* **Accuracy (точност)** – измерва общия дял правилно класифицирани примери върху тестовия и тренировъчния набор.
* **Classification Report** – съдържа precision, recall и f1-score за всяка от класите, което позволява по-прецизен анализ на представянето.
* **Brier Score** – метрика, която измерва калибрираността на вероятностите, предсказани от модела. Колкото по-нисък е Brier score, толкова по-добре.
* **Confusion Matrix** – матрица, която показва детайлно разпределението на вярно и грешно класифицирани примери за всеки клас.

**Резултати и анализ**

* Моделът достига обща точност от **99.57%** според анализа от ml\_metrix.
* Precision, recall и f1-score са много високи (около 1.00) за повечето класове, което показва отлична класификация.
* Конфузионната матрица (confusion matrix) показва малко грешки при класовете **S** и **XS**, но те са минимални:
* От ml\_evaluation виждаме, че:
  + Точност при трениране: **99.82%**
  + Точност при тестване: **98.57%**
  + Среден Brier score: **0.0053**, което показва много добра увереност на предсказанията.

# Избор на признаци

**Критерии за избор**

* Признаците са избрани на база тяхната **важност**, измерена чрез feature importance от двата файла.
* Водещи по значимост са следните признаци:
  + **chest**
  + **weight**
  + **waist**
  + **height**
* Признаците **gender**, **garment\_type**, **material** и **body\_type** имат много ниска важност и минимално влияние върху представянето на модела.

**Тестване и селекция**

* При премахване на по-слабо значимите признаци, точността и Brier score остават почти непроменени, което потвърждава правилния избор на основните признаци.
* Анализът на confusion matrix при различни конфигурации на входните признаци показва стабилно представяне на модела при запазване на ключовите характеристики.

A graph with numbers and squares

Description automatically generated📊 Feature Importance:

|  |
| --- |
| feature importance |
| chest 0.518857 |
| weight 0.072571 |
| height 0.043143 |
| waist 0.040714 |
| material 0.001143 |
| body\_type 0.000857 |
| gender 0.000000 |
| garment\_type 0.000000 |

# Интеграция в уеб приложение