AUFGABE - PREISDATENANALYSE
ABSCHLUSSBERICHT

Alexander Prinz

01.07.2020

Zusammenfassung

Ziel des Projektes war zum einen die Entwicklung eines Tools zum crawlen/scrapen von Preisdaten bestimmter Artikel der Marke Lego® von zwei konkreten Online Shops. Das Hauptziel bestand jedoch darin, die so ermittelten Preisdaten derart aufzubereiten, dass eine Analyse auch durch Visualisierung der Daten möglich ist, um die Preisgestaltung beider Shops untersuchen zu können.

Inhalt

Zielsetzung

Web Scraper/Crawler

Das Tool soll in der Lage sein, automatisiert Preisdaten mehrerer Online Shops auf Basis des vom Webserver übermittelten HTML Dokuments pro Artikel und der Dazugehörigen URL zu beziehen.

Das Script iteriert über ein Pandas DataFrame welches die URL‘s von der Datei <URLs.csv> eingelesen hat. Jede Zeile enthält in ihrer ersten Spalte den Artikelnamen sowie in den folgenden Spalten die URL’s der einzelnen Shops. Diese sind durch den Bezeichner <url_store_i> bezeichnet, wobei das <i> für einen Laufindex steht welcher von 0 an hochgezählt wird. Also url_store_0, url_store_1, …, url_store_n-1.

Zum Scrapen der Preise bildet die Klasse <ScraperClass.py> die Basis. Die Kernmethode der Klasse < .parse_n_select()> übernimmt die bei der Instantiierung eines entsprechenden Klassenobjekts definierten Attribute <self.url> und <self.store_id> und wendet abhängig vom Attribut <self.store_id> die zum Web Shop passende Crawling Strategie auf das HTML Dokument an, welches durch das Attribut <self.url> und einem entsprechenden Webserver Request vom Webserver des Shops übermittelt wurde.

Mögliche Laufzeitfehler etwa durch Server Request Probleme, invalider URL’s etc. werden durch entsprechende Exception Catches abgefangen und entsprechend Behandelt. So werden z.B. invalide URL’s in die Log-Datei <err_log.txt> eigetragen um auf Gründe für invalide Scraping-Versuche untersuchen zu können.

Sind die Preise für einen Artikel von der entsprechenden Webseite ermittelt, werden diese in die Datei <prices.csv> geschrieben. Das Schema der Datei ist gleich dem der Datei <URLs.csv>, wobei nun anstatt der URL der Preis in die entsprechende Spalte eingetragen wird.

Data Processing

Das Data Processing baut auf die gescrapten Daten auf. Die durch das Scrapen erzeugte Datei <prices.csv> wird als Pandas DataFrame eingelesen.

Als Metriken zum Untersuchen der Preisegestaltungen der beiden konkreten Web Shops „Amazon“ und „Toys for Fun“ werden sowohl die mittleren Preisdifferenzen als auch die mittleren Preisverhältnisse bezogen auf die 5 gegeben Subbrands [‚Technic‘ ‚Friends‘ ‚Marvel‘ ‚Duplo‘ ‚Harry Potter‘] herangezogen. Das mittlere Verhältnis sei dabei definiert durch:

$$\frac{\overline{Preis_{Amazon\_Artikel\_ID}}}{\overline{Preis_{Toys\ for\ Fun\_Artikel\_ID}}}.$$

Die mittlere Preisdifferenz sei dabei definiert durch:

$$\overline{Preis_{Amazon\_Artikel\_ID}} - \overline{Preis_{Toys\ for\ Fun\_Artikel\_ID}}$$

Zusätzlich werden die mittleren Preise in einer Visualisierung pro Subbrand gegeneinander übergestellt und beide Shops via Boxplots über alle Preise untersucht.

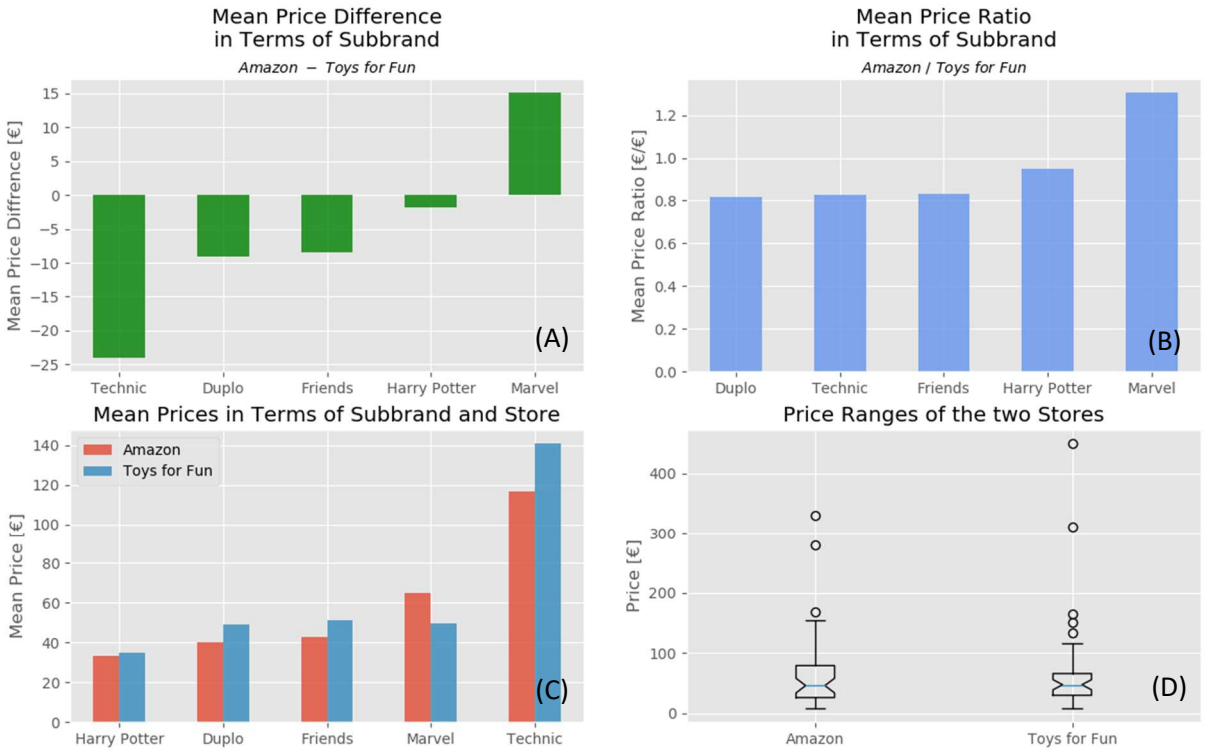
Die Preisdaten bzgl. der Preisverhältnisse sowie Differenzen beider Shops werden als CSV-Datei auf die Festplatte geschrieben.

Ergebnisse

Der Web Scraper/Crawler funktioniert ohne Einschränkungen. Es existieren URL’s ausnahmslos für Amazon, welche keine konkreten Preisdaten enthalten. Diese URL’s werden entsprechend dem Ziel in die log-Datei geschrieben und NAN-Einträge in die Preisliste vorgenommen.

Das Tool ist vollständig über die CSV-Datei der URL’s parametrisiert. Hinzukommende weitere Shops zum Untersuchen bedingen also keiner Veränderung des Hauptskripts. Jedoch hat sich gezeigt, dass das XML-Schema der HTML-Dokumente von Shop zu Shop variieren. Das hat zur Folge, dass in der Kernmethode der Scraper-Klasse für jeden Shop das Schema definiert werden muss.

Alle sonst ermittelbaren Preise werden problemlos in die entsprechende CSV Date eingetragen. Das Datenprozessierungstool funktioniert ebenfalls problemlos. Die Zielsetzungen wurden erfüllt und folgende Abbildung zeigt die Ergebnisse visualisiert.



Ergebnisvisualisierung:

Die Grafik A Zeigt die mittlere Diffrenz zwischen den beiden Shops hinsichtlich der Lego Subbrands. Der Subbrand „Marvel“ zeigt als einziger eine positive Differenz wonach demnach die Preise von Amazon in dieser Kategorie überwiegen. In der Grafik B sind die mittleren Verhältnisse der Preis beider Stores bezogen auf die Subbrands zu sehen. Diese zeigen ebenfalls, dass die Preise der Kategorie „Marvel“ bei „Toys for Fun“ über denen von „Amazon liegen“, da das Verhältnis größer als 1 ist. Die Grafik C zeigt die mittleren Preise beider Shops hinsichtlich der Lego Subbrands als direkten Vergleich durch zwei nebeneinander liegender Balken. Die letzte Grafik veranschaulicht die Preisspanne aller Artikel ungeachtet der Kategorien. Der Median aller Artikel seitens „Amazon“ liegt geringfügig unter dem von „Toys for Fun“. Die Preisspanne von Amazon ist etwas größer als die von „Toys for Fun“. Beide weisen eine schiefe Verteilung auf.