IWGS2_SS20_Zettel9

July 7, 2020

1 Informatische Werkzeuge in den Geistes- und Sozialwissenschaften II

1.1 Hausaufgabe 9 (Annotationen)

Erschienen: 04.07.2020
Abgabe bis: 12.07.2020

Bitte laden Sie Ihre Notebooks bis 23:59 Uhr am Abgabetag in Ihrer Übungsgruppe bei StudOn hoch.

Wenn Ihnen einige der hier verwendeten Konzepte unbekannt sind oder Sie nicht wissen, wie Sie fortfahren sollen, können Sie die Vorlesungsunterlagen zu Rate ziehen oder jederzeit Fragen im Forum oder auf Slack stellen, im Tutorium nachfragen, oder Ihrem Tutor eine Mail schreiben.

1.2 Präambel

Beatrice kann sich vor Aufregung kaum halten. "Ein Computer lernt, Objekte zu erkennen!", sagt sie jetzt schon zum siebten Mal. Sie hat am Wochenende einen Bericht über Machine Learning gelesen und ist jetzt Feuer und Flamme. Offenbar lernen Computer aus Trainingsbeispielen, wie bestimmte Objekte in Bildern aussehen. Sowas könnte man ja auch auf dem Kirmes-Datensatz machen und da die Menschen drauf erkennen. Sie weiß auch schon ganz genau, wie sie ein ähnliches System bauen würde und bietet Ihnen ein Eis an, wenn Sie ihr helfen.

Ihre vorgeschlagene Arbeitsteilung ist folgende: Sie schreibt in den nächsten Tagen "mal schnell" das Neuronale Netzwerk, was die Bilder erkennen soll. In der Zeit können Sie ja schonmal ein Tool bauen, um dem Computer die Trainingsdaten zu geben. Dazu muss man "nur in Bildern markieren und annotieren können". Ein paar Beispielsannotationen hat sie schon in einem XML-Format erstellt.

Danach geht bestimmt alles wie von selbst...

Disclaimer: Wir bauen in IWGS kein Neuronales Netzwerk. Im DH-Modul 3 werden Sie dann diese Algorithmen kennenlernen und selber eins bauen, welches in Bildern bestimmte Gegenstände erkennt. Ein System, mit dem man Daten annotieren kann, ist dafür aber fast immer nötig.

1.3 Aufgabe 9.1 (Annotations-Tabelle, 30 Punkte)

Zu letzter Woche haben wir ein paar Kleinigkeiten am Datenbank-Layout geändert. In der *Images*-Tabelle speichern wir jetzt auch die Breite und Höhe (Width & Height) jedes Bildes. Diese brauchen wir, um wie in der Vorlesung besprochen einen SVG-Canvas für unsere Annotationen zu erstellen.

Der *metadata*-Ordner hat jetzt außerdem 2 Unterordner *images* und *annotations*. In *images* liegen wie gehabt die CSV-Dateien, aus denen wir unsere Datenbank bisher erstellen. Allerdings haben wir jetzt deutlich mehr von den CSVs, und wir haben den Code, der die CSVs liest, so angepasst, dass er alle CSVs in diesem Ordner liest, nicht nur eines.

Um die Datenmenge etwas in Grenzen zu halten, nehmen wir fürs Erste nur Bilder in die Datenbank auf, für die eine Bilddatei existiert.

Im *metadata/annotations*-Ordner liegen XML-Dateien, die jeweils eine oder mehrere Annotationen beschreiben. Wir haben den Code, der die Dateien liest schon vorgegeben. Dieser befindet sich ebenfalls in *image_database.py*. Falls Sie interessiert daran sind, hier eine kurze Erklärung. Jede der XML-Dateien hat in etwa folgendes Format:

```
<annotation>
    <filename>0000352841.jpg</filename>
    <folder>users/pbell77//gypsies</folder>
    <source><submittedBy>Peter Bell</submittedBy></source>
    <imagesize><nrows>648</nrows><ncols>650</ncols></imagesize>

    <object>
        <name>dance</name>
        <deleted>0</deleted>
        <verified>0</verified>
        <occluded>no</occluded>
        <attributes>row</attributes>

        <parts>
            <hasparts></hasparts>
            <ispartof></ispartof>
        </parts>

        <date>12-Apr-2019 17:01:10</date>
        <id>0</id>
        <type>bounding_box</type>

        <polygon>
            <username>Peter</username>
            <pt><x>102</x><y>272</y></pt>
            <pt><x>391</x><y>272</y></pt>
            <pt><x>391</x><y>350</y></pt>
            <pt><x>102</x><y>350</y></pt>
        </polygon>
    </object>
```

```
    ... Hier können noch mehr <object />-Tags kommen ...
</annotation>
```

Der vorgegebene Code verwendet die `xml.etree`-Bibliothek zum "Parsen", also dem Lesen der Datei. Jeder `<object>`-Tag beschreibt eine Annotation. Die Daten, an denen wir interessiert sind, sind in jedem XML: - Der `<filename>` ganz oben, der ein Bild referenziert - Für jeden `<object>`-Tag: - Der `<name>` ist die Beschreibung der Annotation. Im Beispiel oben ist dort wohl eine tanzende Person zu sehen. - Der `<type>` beschreibt die Form der Annotation. Derzeit unterstützen wir nur Bounding-Boxes, also Rechtecke. - `<polygon>` beschreibt dann die 4 Ecken der Box. Unser Code wandelt die 4 Ecken dann um in die Darstellung: (X, Y, Breite, Höhe).

Aufgabe: In *image_database.py* finden Sie 2 Stellen, die mit `TODO` markiert sind.

Erstellen Sie in der Datenbank eine Tabelle `Annotations`, die folgende Spalten hat: - ID, Integer, Primary Key - ImageID, Integer, Foreign Key auf ID in der Images-Tabelle - Description, String - X, Integer - Y, Integer - Width, Integer - Height, Integer

Fügen Sie außerdem die Annotationen, die aus dem XML gelesen werden, in die Datenbank ein.

1.4 Aufgabe 9.2 (Anzeigen der Annotationen, 70 Punkte)

Das Skript *image_server.py* fragt in der `/details`-Route, die wir letzte Woche geschrieben haben, die verfügbaren Annotationen ab und übergibt sie an *details.tpl*. Dort werden derzeit einfach die Annotationen in einer HTML-Liste angezeigt. Um Ihren Code aus Teilaufgabe 1 zu testen, können Sie die *details*-Seite eines Bildes besuchen, für das Annotationen existieren. Das letzte Bild in der Datenbank (*Dorpskermis op het feest van de H. Joris*) sollte einige Annotation haben.

Nochmal zur Wiederholung, wie wir Annotationen in IWGS anzeigen (siehe auch die Vorlesung): Unsere Annotationen sind derzeit einfache Rechtecke. Wir werden also mithilfe von SVG das in der Datenbank gespeicherte Bild anzeigen und mit den Rechtecken überlagern. Wenn BenutzerInnen mit der Maus über die Rechtecke fahren (hovern), dann wird unter dem Bild die zur jeweiligen Annotation passende Beschreibung angezeigt. Dabei machen wir uns ein paar CSS-Tricks zunutze. Das CSS ist allerdings schon in *details.tpl* vorgegeben.

Aufgabe: Ihre Aufgabe ist es, die Annotationen in *details.tpl* so anzuzeigen, wie wir es in der Bildvorlesung besprochen haben. Dazu zeigen Sie zunächst das Bild an. Danach generieren Sie für jede Annotation ein `<rect>` und ein `<text>`-Tag und setzen die entsprechenden Daten ein. Überlegen Sie sich genau, welches Element wo platziert werden soll.

[ ]: