

**Технически университет – София**

**Факултет по приложна математика и информатика**

**Курсова работа**

**по**

**Технологии за големи данни**

**на тема**

**Проектиране на модел за разпознаване на емоции на лица**

**Изготвили: Савина Вълчанова, Денис Цолов**

**Фак. номера:** **961324002, 961324006**

**Група: 252**

Съдържание

1. **Въведение и настройка на средата**
   * **Преглед на проекта и цели**
   * **Библиотеки и зависимости**
   * **Конфигурация на средата за използване на GPU**
2. **Подготовка и предварителна обработка на данните**
   * **Преглед на набора от данни FER-2013**
   * **Структура на директориите и разпределение по класове**
   * **Премахване на класа „отвращение“ и увеличаване на данните**
3. **Проектиране и компилация на модела**
   * **Архитектура на последователен модел**
   * **Конволюционни слоеве, нормализация и Dropout**
   * **Функция на загуба, оптимизатор и метрики**
   * **Ранно спиране, запис на модели и логване**
4. **Обучение на модела**
   * **Конфигуриране на стъпки за епохи при обучение и валидация**
   * **Процес на обучение и проследяване на напредъка**
5. **Анализ на резултатите**
   * **Обобщение на постигнатите резултати**
   * **Обзор на проблеми и насоки за подобрения**
6. **Въведение и настройка на средата**

Проектът има за цел да обучи и демонстрира работата на модел за разпознаване на емоции въз основа на набор от данни с изображения на лица - FER-2013. Проектът включва няколко основни етапа, включително предварителна обработка на данните, проектиране на дълбока невронна мрежа, обучение и демонстрация на резултата и чрез код, стартиращ видеокамерата на използвания лаптоп.

Използваните библиотеки и зависимости включват основни библиотеки като os, datetime и NumPy за обработка на данни, TensorFlow и Keras за изграждане на модела и обучението му. За визуализация и анализ на данни и резултати са използвани Matplotlib и Seaborn, а за откриване на лица в изображение – готов модел от библиотеката OpenCv. Също така предварително е конфигурирана GPU-подпомогната среда за разработка на обучаващата машина чрез локално инсталираните библиотеки CUDA и cuDNN.

Конфигурацията на локалната среда гарантира, че TensorFlow може да използва GPU за оптимизиране на процесите. Чрез тези настройки и проверки на наличието на GPU устройство се осигурява по-ефективно използване на ресурсите, за да се ускори процесът на обучение и оптимизация на модела.

1. **Подготовка и предварителна обработка на данните**

Подготовката на данните за модела включва анализ и проспособяване на кода за зареждане на на набора от данни FER-2013, който съдържа изображения на лица, класифицирани по емоции. Наборът от данни е организиран в две основни директории – за обучение и за тестване, всяка от които съдържа поддиректории, съответстващи на различни емоционални категории. Всяка категория включва изображения, които представляват различни емоции –гняв, страх, радост, неутралност, тъга и изненада.

За да се осигури по-добра точност на модела, класът „отвращение“ е премахнат поради значително по-малкия брой примери в сравнение с останалите категории, което би могло да доведе до дисбаланс при обучението. За анализа на данните са използвани графични методи – като бар графики – за визуализация на разпределението по класове, с цел по-добро разбиране на броя изображения във всяка категория. Извършено е и разширяване на набора от данни чрез техники за аугментация, които увеличават броя на обучителните примери чрез трансформации на изображенията. Използвани са функции за промяна на размерите и завъртане на изображения, за да се повиши обобщаващата способност на модела.

1. **Проектиране и компилация на модела**

Моделът е създаден с използване на последователна архитектура (Sequential), състояща се от няколко слоя конволюционни слоеве, нормализация и Dropout. Тази архитектура е подходяща за класифициране на изображения, тъй като конволюционните слоеве позволяват извличане на важни характеристики от изображенията, а нормализацията и Dropout слоевете помагат за предотвратяване на пренасищане (overfitting).

Конволюционните блокове в модела включват три слоя Conv2D със следните характеристики:

* Първият блок започва със слой Conv2D от 32 филтъра, последван от още един слой Conv2D с 64 филтъра, нормализация и MaxPooling слой за намаляване на размерността на данните.
* Вторият и третият конволюционен блок също включват увеличаващ брой филтри и добавена регуляризация за по-добра обобщаваща способност на модела.

В плътната част на модела (Dense слоевете) са включени BatchNormalization и Dropout слоеве, които подобряват стабилността и устойчивостта на обучението. Изходният слой има шест неврона, по един за всяка категория, и използва softmax активация за предсказване на вероятностите за различните емоции.

За компилацията на модела е използвана функция на загуба categorical\_crossentropy, оптимизатор Adam с начална скорост на обучение 0.0001, и метриката accuracy. Тези настройки са подходящи за многокласова класификация и помагат на модела да се адаптира към набора от данни по-ефективно.

Допълнително са дефинирани и четири callback функции (помощни функции, които се изпълняват по време на процеса на обучение на моделит):

* ранно спиране (EarlyStopping) за прекратяване на обучението при липса на подобрение,
* редуциране на скоростта на обучение при застой (ReduceLROnPlateau),
* записване на най-добрата версия на модела (ModelCheckpoint)
* логване на процеса на обучение (CSVLogger).

Тези настройки улесняват управлението на обучението по време на изпълнението му и помагат за по-добро проследяване на резултатите.

1. **Обучение на модела**

Обучението на модела започва след предварителната обработка на данните и конфигурирането на архитектурата на модела и настройване на callback функциите. Използват се конфигурации за steps\_per\_epoch и validation\_steps, които определят броя на стъпките за всяка епоха на обучение и валидация въз основа на размера на набора от данни и големината на партидата (batch size).

Процесът на обучение използва метода fit, който извиква обучаващите и валидиращите данни. В случая са предвидени до 250 епохи, като броят на реалните епохи може да е по-малък в зависимост от ефекта на callback функциите.

Ако точността на валидация (validation accuracy) не се подобрява за последните 7 епохи на обучение поне с конфигурираната минимална зададена стойност (0.001), се намалява стъпката на обучение (ReduceLROnPlateau). Това помага на модела да преодолее потенциални трудности при достигането на оптимални стойности за загубата.

Ако за последните 20 епохи не е регистрирано никакво подобрение, се активира ранното спиране (EarlyStopping), което автоматично прекратява обучението и възстановява най-добрата версия на модела (ModelCheckpoint).

Също така, CSVLogger запазва резултатите от всяка епоха във файл, което позволява последващ анализ и визуализация на процеса на обучение.

Целият процес на обучение се записва и визуализира, което позволява на потребителя да наблюдава и оценява ефективността на модела с напредването на епохите. Това позволява и анализ след обучението за подобрение на параметрите с цел оптимизиране на крайния резултат.

1. **Анализ на резултатите**

Резултатите от обучението, както и цялостния код могат да бъдат прегледани от браузъра при отваряне на гитхъб страницата на проекта от следния линк: <https://github.com/savina01/Emotions-Recognitions-Demo>.

Обучението на модела завършва след 1 час и 52 минути, достигайки 115-та епоха, благодарение на функцията за ранно спиране (EarlyStopping), която прекратява обучението поради липса на подобрения във валидационната точност (validation accuracy). Най-добрата версия на модела е запазена от checkpoint-а на 95-та епоха с регистрирана точност според данните за валидация от val\_accuracy: 0.6909. Това е сравнително добра стойност, която се потвърджвава до голяма степен и при демонстрацията в реални условия чрез активиране на предната камера на обучавашата машина и изпробване с истинско човешко лице. Изпробвани са многобройни подходи за промяна на параметрите и архитектурата на модела, но по-висока точност изглежда недостижима поради лимитации на използвания датасет, а именно:

* Шумни и нискокачествени изображения - FER-2013 съдържа изображения с ниска резолюция (48x48 пиксела), което затруднява модела да улавя фини детайли на лицевите изражения, особено при сходни емоции. Качеството на изображението е критично за задачи като разпознаване на емоции, където дори малки промени в лицевите черти могат да предадат различни емоции. Също така има големи разнообразия в начините, по които е представена дадена емоция и видът лице (някои снимки дори не са на човешки лица).
* Сходство между някои емоции - емоции като "страх" и "изненада" имат подобни лицеви изражения, което води до грешки в класификацията. Моделът може да се обърква между тях поради липсата на ясно изразени разлики в чертите на лицето в тези случаи.
* Липса на контекст - емоциите често са свързани със социалния и визуалния контекст, които в този набор от данни липсва. Липсата на допълнителни фактори ограничава способността на модела да разбере контекста на емоциите - класификацията се осъществава само на базата на изображението на лицето.

Всички тези фактори водят до достигането на плато в размер на около 70% в метриката „точност”, дори при по-задълбочено обучение. Това прави FER-2013 предизвикателен за разпознаване на емоции и показва нуждата от по-богати и качествени набори от данни за по-добри резултати.