



دانشگاه اصفهان

دانشکده مهندسی‌کامپیوتر

**پایان‌نامه کارشناسی**

رشته مهندسی کامپیوتر

بسته اصلی: هوش مصنوعی و رباتیکز

بسته فرعی: نرم افزار

**عنوان**

**پویشگر و تحلیلگر ارزش مواد غذایی**

**استاد راهنما:**

سرکارخانم دکتر افسانه فاطمی

**پژوهشگر:**

علی پورقیصری

شهریور 1403



دانشگاه اصفهان

دانشکده مهندسی‌کامپیوتر

پروژه کارشناسي رشته‌ي مهندسي کامپیوتر

**آقاي علی پورقیصری**

**تحت عنوان**

**پویشگر و تحلیلگر ارزش مواد غذایی**

در تاريخ / / 14 توسط هيأت داوران زير بررسي و با نمره به تصويب نهايي رسيد.

1. استاد راهنماي پروژه

دکتر امضا

1. استاد داور

دکتر امضا

امضای‌ مدیرگروه

**تشکر و قدرداني**

سپاس بیکران از پروردگار که بار دیگر فرصت یادگیری را به من ارزانی داشت. از پدر و مادر عزیز و مهربانم که در طول زندگی با تحمل زحمات فراوان، همواره برای موفقیت و خوشبختی من تلاش کردند، صمیمانه قدردانی می‌کنم. همچنین از استاد ارجمند، دکتر افسانه فاطمی، که با تلاش‌های بی‌دریغ خود در تحقق این پایان‌نامه سهم بسزایی داشتند، نهایت تشکر را دارم. قدردان دوستان عزیزم نیز هستم که با حمایت‌هایشان در به سرانجام رساندن این پایان‌نامه نقش مهمی ایفا کردند.

لازم می‌دانم از تمامی افرادی که در بهبود و توسعه هوش مصنوعی در زبان فارسی سهمی داشته‌اند، تشکر کنم و امیدوارم که با تلاش توسعه‌دهندگان، دیگر هیچ محدودیتی برای انجام پژوهش‌های فارسی در مقایسه با سایر زبان‌ها وجود نداشته باشد.

**تقدیم به**

**خانواده عزیزم**

**چکیده:**

در این پروژه، یک برنامه کاربردی توسعه داده شده است که به کاربران امکان می‌دهد با استفاده از تکنیک‌های پیشرفته پردازش تصویر و بینایی ماشین، اطلاعات مهمی را از روی برچسب‌های ارزش غذایی محصولات فروشگاهی استخراج و بخوانند. این برنامه با پویش برچسب‌های موجود بر روی محصولات، اطلاعاتی نظیر میزان انرژی محصول برحسب کیلوکالری، و مقدار مواد مختلف از جمله قند، نمک، چربی، و اسیدهای چرب ترانس برحسب گرم را نمایش می‌دهد. علاوه بر این، برنامه توانایی تشخیص و ارائه مقدار ماده معیار در محصول را برحسب گرم یا میلی‌لیتر دارد، که می‌تواند به کاربران در تحلیل بهتر ارزش غذایی محصول کمک کند.

در مرحله بعد، برنامه با استفاده از اطلاعاتی که به دست آورده، دست به تجزیه و تحلیل می‌زند و مقدار انرژی مصرفی را با مقادیری از فعالیت‌های بدنی روزمره مقایسه می‌کند. نهایتاً، برنامه شماره پروانه بهداشتی ساخت محصول را نیز تشخیص می‌دهد و با استعلام این شماره، کاربران قادر خواهند بود به اطلاعات فنی دقیق‌تری همچون نام تجاری محصول، نام کارخانه تولیدکننده، تاریخ دریافت و انقضای پروانه بهداشتی، و آدرس کارخانه دسترسی پیدا کنند. هدف اصلی این پروژه، افزایش آگاهی مصرف‌کنندگان در انتخاب محصولات سالم‌تر و بهبود کیفیت خرید آنها با ارائه اطلاعات شفاف و قابل اعتماد می‌باشد.

فهرست مطالب

|  |  |
| --- | --- |
| صفحه | عنوان |

[فصل اول مقدمه و بیان مسئله 10](#_Toc175484539)

[۱-۱- توضیحات کلی 10](#_Toc175484540)

[۲-۱- اهداف پروژه 11](#_Toc175484541)

[۳-۱- دامنه و محدوده پروژه 12](#_Toc175484542)

[فصل دوم ادبیات پژوهش 18](#_Toc175484543)

[۱-۲- مقدمه 18](#_Toc175484544)

[4-2- ادبیات ارزیابی 29](#_Toc175484545)

[5-2- تجزیه و تحلیل اطلاعات 32](#_Toc175484546)

[6-2- جمع بندی: 34](#_Toc175484547)

[فصل سوم پیشینه پژوهش 35](#_Toc175484548)

[۱-3- مقدمه 35](#_Toc175484549)

[۲-3- تاریخچه و توسعه OCR 35](#_Toc175484550)

[3-3-مروری بر مطالعات مرتبط 36](#_Toc175484551)

[4-3- جمع بندی 38](#_Toc175484552)

[فصل چهارم ساختار و آموزش مدل‌ها 39](#_Toc175484553)

[۱-4- مقدمه 39](#_Toc175484554)

[2-4- مدل تشخیص برچسب 39](#_Toc175484555)

[5-4- جمع بندی 47](#_Toc175484556)

[فصل پنجم ارزیابی مدل‌ها و استفاده از اطلاعات 48](#_Toc175484557)

[۱-5- مقدمه 48](#_Toc175484558)

[۲-5- ارزیابی مدل تشخیص برچسب 48](#_Toc175484559)

[3-5- ارزیابی مدل تشخیص کلمات 50](#_Toc175484560)

[4-5- ارزیابی مدل خوانش حروف و ارقام 51](#_Toc175484561)

[5-5- تجزیه و تحلیل انرژی محصول 53](#_Toc175484562)

[۲-3- اطلاعات فنی محصول 54](#_Toc175484563)

[6-5- جمع بندی 55](#_Toc175484564)

[فصل ششم جمع بندی و پیشنهاد‌هایی برای بهبود پژوهش 56](#_Toc175484565)

[1-6- جمع بندی کلی 56](#_Toc175484566)

[2-6- بهبود‌های قابل اعمال 57](#_Toc175484567)

[منابع 59](#_Toc175484568)

فهرست اشکال

|  |  |
| --- | --- |
| صفحه | عنوان |

[تصویر 1- برچسب استاندارد مد نظر در این پروژه 14](#_Toc175484851)

[تصویر 2- نمونه ای از برچسب غیر استاندارد 15](#_Toc175484852)

[تصویر 3- نمونه ای از برچسب با عکس برداری نامناسب 15](#_Toc175484853)

[تصویر 4- نمونه ای از برچسب با اعداد ناخوانا 16](#_Toc175484854)

[تصویر 5- نمونه ای از برچسب مخدوش 16](#_Toc175484855)

[تصویر 6- محیط لیبل استودیو و فرایند برچسب‌کذاری 21](#_Toc175484856)

[تصویر 7- محیط وب‌سایت سایتاک 22](#_Toc175484857)

[تصویر 8- ساختار کلی مدل یولو 26](#_Toc175484858)

[تصویر 9- ساختار کرفت 29](#_Toc175484859)

[تصویر 10- مقایسه سرعت و تعداد پارامترهای مدل خوانش متن 29](#_Toc175484860)

[تصویر 11- برچسب ارزش مواد غذایی آمریکایی 37](#_Toc175484861)

[تصویر 12- قالب قابل قبول یولو 41](#_Toc175484862)

[تصویر 13- عملیات تشخیص برچسب 41](#_Toc175484863)

[تصویر 14- یک ردیف از برچسب هر عکس 42](#_Toc175484864)

[تصویر 15- مدل کرفت، قبل از آموزش 42](#_Toc175484865)

[تصویر 16- قالب قابل قبول کرفت 43](#_Toc175484866)

[تصویر 17- تشخیص کلمات و ارقام به وصیله کرفت 44](#_Toc175484867)

[تصویر 18- برچسب‌های تصویر قبل 44](#_Toc175484868)

[تصویر 19- خروجی مدل اوسی‌آر قبل از آموزش 45](#_Toc175484869)

[تصویر 20- خروجی مدل اوسی‌آر بعد از آموزش 46](#_Toc175484870)

[تصویر 21- قالب قابل قبول مدل اوسی‌آر 47](#_Toc175484871)

[تصویر 22- نمودار سه معیار دقت، صحت و بازخوانی برای مدل یولو 49](#_Toc175484872)

[تصویر 23- نمودار سه معیار دقت، صحت و میانگین هارمونیک برای مدل کرفت 50](#_Toc175484873)

[تصویر 24- نمودار معیارهای دقت و میانگین خطای نرمالیزه شده در OCR 51](#_Toc175484874)

[تصویر 25- تابع زیان داده‌های آموزش و آزمایش برای مدل OCR 52](#_Toc175484875)

[تصویر 26- خروجی برنامه بعد از تجزیه و تحلیل انرژی هر سهم محصول 54](#_Toc175484876)

[تصویر 27- اطلاعات فنی و نمای وب‌سایت پس از استعلام محصول 55](#_Toc175484877)

فهرست جداول

|  |  |
| --- | --- |
| صفحه | عنوان |

[جدول 1- ماتریس درهم‌ریختگی 30](#_Toc175484692)

[جدول 2- انواع مدل‌های یولو 39](#_Toc175484693)

[جدول 3- مقایسه آمار اولین و آخرین دوره آموزشی در یولو 49](#_Toc175484694)

[جدول 4- مقایسه آمار اولین و آخرین دوره آموزشی در کرفت 51](#_Toc175484695)

[جدول 5- مقایسه آمار دقت و میانگین خطای نرمالیزه‌شده در اولین و آخرین دوره 52](#_Toc175484696)

[جدول 6- مقایسه آمار مقادیر توابع زیان داده‌های آموزش و آزمایش در اولین و آخرین دوره 53](#_Toc175484697)

# فصل اول مقدمه و بیان مسئله

## ۱-۱- توضیحات کلی

در دنیای امروز که سرعت زندگی و مشغله‌های روزمره افزایش یافته، اهمیت دسترسی به اطلاعات دقیق و قابل اطمینان درباره محصولات غذایی بیشتر از همیشه احساس می‌شود. مصرف‌کنندگان به دنبال اطلاعاتی هستند که به آنها کمک کند تا با آگاهی کامل‌تر محصولات غذایی مناسب‌تر و سالم‌تری را انتخاب کنند. این موضوع به ویژه با توجه به افزایش نگرانی‌ها درباره تأثیرات مضر مواد غذایی نظیر قند، نمک، چربی‌های اشباع و اسیدهای چرب ترانس بر سلامت انسان، اهمیت بیشتری پیدا کرده است.

یکی از چالش‌های اساسی مصرف‌کنندگان در این زمینه، عدم دسترسی سریع و آسان به اطلاعات دقیق و جامع درباره ارزش غذایی محصولات است. برچسب‌های مواد غذایی که به طور معمول بر روی محصولات درج می‌شوند، به دلیل پیچیدگی اطلاعات یا اندازه کوچک نوشته‌ها، ممکن است برای بسیاری از مصرف‌کنندگان خوانا نباشند یا به راحتی قابل فهم نباشند. علاوه بر این، عدم آگاهی کافی از محتوای دقیق محصولات می‌تواند منجر به انتخاب‌های نادرست و در نهایت تهدید سلامت مصرف‌کنندگان شود.

از سوی دیگر، نیاز به دسترسی به اطلاعات فنی محصولات، نظیر نام تجاری، کارخانه تولیدکننده، و تاریخ انقضا، از اهمیت بسزایی برخوردار است. این اطلاعات نه تنها به مصرف‌کنندگان کمک می‌کند تا از کیفیت و ایمنی محصولات مطمئن شوند، بلکه امکان پیگیری و استعلام شماره پروانه بهداشتی ساخت محصول را نیز فراهم می‌آورد.

پروژه حاضر با هدف ارائه راه‌حلی نوآورانه برای این چالش‌ها طراحی شده است. در این پروژه، یک برنامه کاربردی توسعه داده شده است که به کاربران امکان می‌دهد با پویش برچسب‌های ارزش غذایی محصولات فروشگاهی، اطلاعات مندرج بر روی آنها را با استفاده از تکنولوژی پردازش تصویر[[1]](#footnote-1) و بینایی ماشین[[2]](#footnote-2) به صورت خودکار بخوانند. این برنامه قادر است میزان انرژی، قند، نمک، چربی‌ها، اسیدهای چرب ترانس و سایر اطلاعات مهم تغذیه‌ای را استخراج کرده و به شکلی ساده و قابل فهم در اختیار کاربران قرار دهد. علاوه بر این، امکان استعلام شماره پروانه بهداشتی ساخت محصول نیز فراهم شده است که به کاربران اجازه می‌دهد اطلاعات فنی محصول را بررسی کرده و از صحت و سلامت آن اطمینان حاصل کنند.

این پروژه با هدف تسهیل دسترسی مصرف‌کنندگان به اطلاعات تغذیه‌ای و فنی محصولات غذایی، افزایش آگاهی عمومی، و ارتقای سلامت جامعه، نقش مهمی در انتخاب‌های آگاهانه‌تر و سالم‌تر ایفا می‌کند.

## ۲-۱- اهداف پروژه

پروژه حاضر با هدف ارتقای سطح آگاهی و توانمندسازی مصرف‌کنندگان در انتخاب محصولات غذایی مناسب و سالم‌تر طراحی شده است. برای دستیابی به این هدف کلی، اهداف جزئی‌تری نیز تعریف شده‌اند که در ادامه به تفصیل به آنها پرداخته می‌شود:

**1-2-1- افزایش آگاهی مصرف‌کنندگان از محتوای تغذیه‌ای محصولات غذایی**

هدف اصلی این پروژه فراهم کردن دسترسی آسان و سریع مصرف‌کنندگان به اطلاعات تغذیه‌ای محصولات غذایی است. با استفاده از این برنامه، کاربران می‌توانند به‌راحتی میزان انرژی، قند، نمک، چربی‌ها، اسیدهای چرب ترانس و سایر مواد مغذی را مشاهده و تحلیل کنند. این اطلاعات به مصرف‌کنندگان کمک می‌کند تا با شناخت بهتری از محتوای محصولات، تصمیمات بهتری برای سلامتی خود بگیرند.

**1-2-2- تسهیل فرآیند انتخاب محصولات غذایی سالم و مناسب**

یکی دیگر از اهداف مهم این پروژه، ساده‌سازی فرآیند انتخاب محصولات غذایی سالم‌تر است. با استفاده از تکنولوژی پردازش تصویر و بینایی ماشین، برنامه قادر است اطلاعات پیچیده و گاه ناخوانای برچسب‌های مواد غذایی را به سرعت و به صورتی قابل فهم در اختیار کاربران قرار دهد. این امر به کاربران امکان می‌دهد تا در کوتاه‌ترین زمان ممکن، محصولاتی را انتخاب کنند که با نیازها و اهداف سلامتی آنها هماهنگ باشد.

**1-2-3- ارائه اطلاعات فنی و بهداشتی محصولات**

علاوه بر اطلاعات تغذیه‌ای، این پروژه به کاربران اجازه می‌دهد تا با بررسی شماره پروانه بهداشتی ساخت محصول، به اطلاعات فنی مهمی نظیر نام تجاری محصول، نام کارخانه تولیدکننده، تاریخ دریافت و انقضا پروانه بهداشتی، و آدرس کارخانه دسترسی پیدا کنند. این اطلاعات به کاربران کمک می‌کند تا از کیفیت و ایمنی محصولات مطمئن شوند و انتخاب‌های آگاهانه‌تری داشته باشند.

**1-2-4- افزایش شفافیت در صنعت غذا**

یکی دیگر از اهداف این پروژه، افزایش شفافیت در بازار محصولات غذایی است. با ارائه اطلاعات دقیق و معتبر درباره محتوا و مشخصات فنی محصولات، این پروژه به تقویت اعتماد مصرف‌کنندگان نسبت به اطلاعات ارائه‌شده توسط تولیدکنندگان کمک می‌کند و به ایجاد بازاری شفاف‌تر و قابل‌اعتمادتر منجر می‌شود.

**1-2-5- حمایت از تولیدکنندگان متعهد به کیفیت**

این پروژه می‌تواند به تولیدکنندگان محصولات غذایی که به کیفیت و شفافیت اطلاعات محصولات خود اهمیت می‌دهند، کمک کند تا در بازار رقابتی جایگاه بهتری پیدا کنند. با ارائه ابزارهایی که به مصرف‌کنندگان امکان دسترسی به اطلاعات دقیق و قابل‌اعتماد را می‌دهد، این پروژه به ترویج محصولات باکیفیت و افزایش اعتماد مشتریان به برندهای معتبر کمک می‌کند.

**1-2-6- بهبود سلامت عمومی جامعه**

هدف نهایی این پروژه ارتقای سلامت عمومی جامعه است. با افزایش دسترسی مصرف‌کنندگان به اطلاعات تغذیه‌ای و فنی دقیق، این پروژه به کاهش مصرف مواد مضر مانند قند، نمک و چربی‌های ناسالم کمک می‌کند. این امر می‌تواند به کاهش بروز بیماری‌های مرتبط با تغذیه نادرست، نظیر چاقی، دیابت، و بیماری‌های قلبی و عروقی منجر شود و در نهایت سطح کلی سلامت جامعه را بهبود بخشد.

این اهداف در مجموع به تحقق یک محیط غذایی سالم‌تر و شفاف‌تر کمک کرده و مصرف‌کنندگان را قادر می‌سازد تا با اطمینان و آگاهی بیشتری محصولات غذایی خود را انتخاب کنند.

## ۳-۱- دامنه و محدوده پروژه

**1-3-1- جمع‌آوری و آماده‌سازی داده‌ها**

* + - * **مجموعه داده**: برای آموزش و ارزیابی مدل‌های مورد استفاده، مجموعه‌ای از **1000 تصویر** از محصولات غذایی مختلف از چندین فروشگاه جمع‌آوری شده است. این تصاویر به منظور تشخیص برچسب محصول، تشخیص نوشته‌های داخل برچسب، و خواندن متن‌ها، برچسب‌گذاری[[3]](#footnote-3) و آماده‌سازی شده‌اند. این مجموعه داده برای آموزش سه مدل که در فصل‌های بعد به آنها اشاره خواهیم کرد، استفاده خواهدشد. در اینجا، استفاده این تصاویر و برپسب‌گذاری آنها به ترتیب آمده است.

1. **جمع آوری تصاویر از فروشگاه‌های مختلف**
2. **بازبینی تصاویر و تمیز کردن مجموعه داده**
3. **برچسب‌گذاری مجموعه داده برای مدل تشخیص‌دهنده[[4]](#footnote-4) برچسب و یافتن آن از داخل تصویر**
4. **برچسب‌گذاری مجموعه داده برای مدل تشخیص‌دهنده کلمات**
5. **برچسب‌ گذاری مجموعه داده برای مدل خواننده کلمات (OCR)**

برای این برچسب‌گذاری‌ها، از محیط‌های مختلفی استفاده شده که در فصل بعد آنها را معرفی می‌کنیم.

* + - * **برچسب‌گذاری و آماده‌سازی داده‌ها**: تصاویر سه بار و برای سه مدل برچسب‌گذاری شده‌اند:

1. **تشخیص برچسب محصول**: شناسایی موقعیت برچسب روی محصول.
2. **تشخیص نوشته‌های داخل برچسب**: شناسایی و تعیین مناطق متنی داخل برچسب.
3. **خواندن نوشته‌ها**: استخراج و تفسیر متن‌های موجود در برچسب.

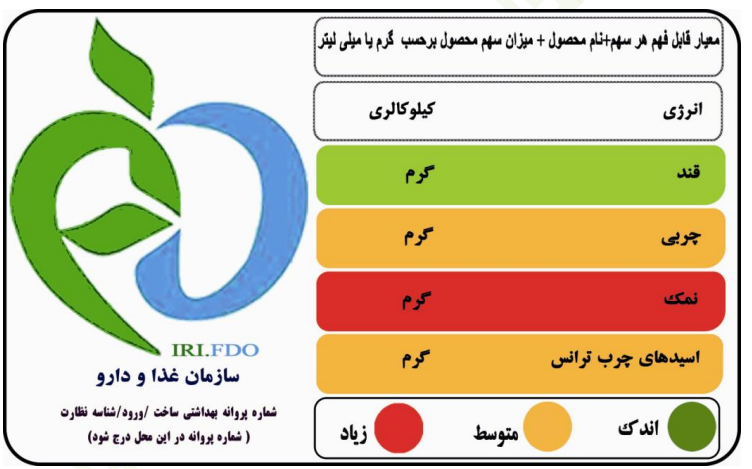
* **آموزش مدل‌ها[[5]](#footnote-5)**: مدل‌های یادگیری ماشین و پردازش تصویر با استفاده از این داده‌ها آموزش دیده‌اند تا به دقت بالا در شناسایی و استخراج اطلاعات برسند. نتیجه این آموزش‌ها و دقت مدل‌ها در فصل‌های بعد به نمایش گذاشته خواهد شد.

**1-3-2- عملکرد برنامه در محیط واقعی**

* **پویش و تحلیل برچسب‌ها**: برنامه طراحی شده است تا در محیط‌های فروشگاهی و با استفاده از دوربین تلفن همراه، برچسب‌های مختلف را پویش کرده و اطلاعات را به صورت دقیق استخراج کند.
* **کاربری و رابط کاربری**: رابط کاربری برنامه فعلا به گونه ای نیست که کاربران بتوانند برروی تلفن همراه خود و یا از طریق سایت به آن دسترسی داشته باشند.

**1-3-3- قالب استاندارد برچسب ارزش مواد غذایی**

این برنامه توانایی خواندن اکثر برچسب‌های موجود در فروشگاه‌ها را دارد اما قالب استانداردی که اداره کل نظارت و ارزیابی فراوده های خوراکی، آرایشی و بهداشتی سازمان غذاو دارو تایین کرده[1]، به عنوان قالب و چارچوب استاندارد این پروژه می‌باشد که به شرح زیر است.

تصویر 1- برچسب استاندارد مد نظر در این پروژه

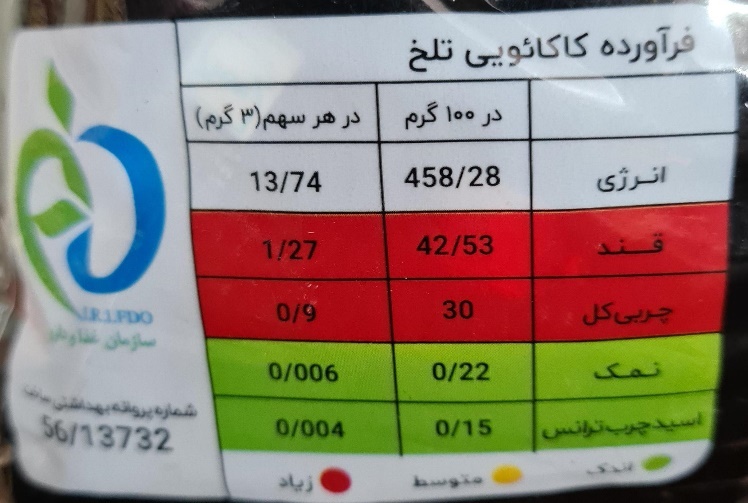
مواردی که در این برچسب به آن باید اشاره کرد به صورت زیر است.

* در بالاترین قسمت این برچسب، باید معیار قابل فهم هر سهم (برای مثال: یک قاشق غذاخوری از محصول یا هر 2 عدد از محصول)، نام محصول، و میزان سهم محصول برحسب گرم یا میلی‌لیتر قرار گیرد. در این قسمت، فقط بخش آخر که میزان سهم محصول است از برچسب خوانده می‌شود.
* در قسمت دوم برچسب، انرژی ای که در هر سهم از محصول موجود است و در قسمت اول ذکر شد، نوشته می‌شود. معیار اندازه گیری این قسمت، همیشه کیلوکالری (Kcal) است و اعداد، قبل از این معیار اندازه گیری قرار می‌گیرند.
* بعد از این دو قسمت، چهار قسمت رنگی وجود دارد که هرکدام یک ماده مغزی را نشان‌ می‌دهد. ترتیبی که این چهار ماده در برچسب دارند به ترتیب: قند، چربی، نمک و اسید‌های چرب ترانس است. معیار اندازه گیری این چهار ماده، گرم است و مانند انرژی، اعداد قبل از معیار قرار می‌گیرند.
* شماره پروانه بهداشتی پروانه ساخت نیز یکی دیگر از مواردیست که باید پویش شود و محل قرار گیری آن، پایین در سمت چپ است.

سایر مواردی که در برچسب هستند، در این پروژه کاربردی ندارند و پویش نخواهند شد.

**1-3-4- خارج از دامنه پروژه**

* **محصولات غیر بسته‌بندی**: پروژه محدود به محصولات غذایی بسته‌بندی‌شده با برچسب‌های واضح و قابل خواندن است و شامل مواد غذایی فاقد برچسب، برچسب غیر استاندارد سازمان غذا و دارو، برچسب‌هایی که به طور کامل و صحیح عکس‌برداری نشده و یا برچسب‌های بسیار آسیب‌دیده نمی‌شود. چند نمونه از این برچسب‌ها را می‌توان در زیر مشاهده کرد.



تصویر 2- نمونه ای از برچسب غیر استاندارد



تصویر 3- نمونه ای از برچسب با عکس برداری نامناسب



تصویر 4- نمونه ای از برچسب با اعداد ناخوانا



تصویر 5- نمونه ای از برچسب مخدوش

* **پشتیبانی از ویژگی‌های خاص**: برنامه به طور خاص برای استخراج و نمایش اطلاعات تغذیه‌ای و فنی طراحی شده است و ویژگی‌های خاصی مانند تجزیه و تحلیل تغذیه‌ای پیشرفته یا مشاوره‌های بهداشتی را شامل نمی‌شود. همچنین رنگ‌ها در برچسب ارزش مواد غذایی، ارزیابی نمی‌شوند.

**4-1- چالش‌ها**

* سیستم‌های OCR معمولاً بر روی متن‌های معمولی آموزش داده می‌شوند که توسط جملات و پاراگراف‌ها تشکیل می‌شوند، اما برچسب‌های تغذیه‌ای، جدولی هستند و تاکید زیادی بر اعداد دارند. ثانیاً، این نمودارها از نظر رنگ‌ها، روشنایی، فونت و اندازه متن، الگوها، بافت‌ها و غیره بسیار متغیر هستند. که توسعه یک سیستم عمومی که قادر به تعمیم همه این شرایط باشد، بسیار چالش برانگیز است.
* تصاویر گرفته‌شده از برچسب‌ها ممکن است شامل نویزهای پس‌زمینه‌ای مانند بازتاب نور، سایه، یا حتی سایر محصولات مجاور باشند که می‌توانند باعث کاهش دقت در تشخیص متن شوند. فیلتر کردن این نویزها بدون از دست دادن اطلاعات مهم از جمله چالش‌های اساسی است.
* برخی برچسب‌ها دارای اطلاعات بسیار فشرده هستند که در یک فضای کوچک قرار داده شده‌اند. تشخیص دقیق این متون فشرده که ممکن است از فونت‌های بسیار کوچک استفاده کنند، چالشی برای OCR به حساب می‌آید.
* در صورتی که برنامه باید در زمان واقعی عمل کند، یعنی کاربر بتواند به محض گرفتن عکس از برچسب، اطلاعات را مشاهده کند، چالش‌هایی مانند بهینه‌سازی سرعت پردازش و کاهش تاخیر زمانی مطرح می‌شوند. این موضوع به خصوص در دستگاه‌های موبایل با منابع محدود مانند حافظه و پردازنده، چالش‌برانگیز است.

این چالش‌ها نه تنها اهمیت پیچیدگی پروژه را نشان می‌دهند، بلکه نیاز به استفاده از رویکردهای نوآورانه و بهینه‌سازی‌های مداوم را برای دستیابی به نتایج مطلوب تأکید می‌کنند.

# فصل دوم ادبیات پژوهش

## ۱-۲- مقدمه

در این فصل، به توضیح مفاهیم کلیدی مطرح‌شده در این پایان‌نامه پرداخته و به اختصار هر یک را بررسی می‌کنیم. در انجام این پژوهش، از ابزارها، روش‌ها و الگوریتم‌های مختلفی در حوزه های مختلف استفاده شده است که در این فصل به کاربرد هر یک از آن‌ها در این پروژه اشاره خواهیم کرد.

مطالب این فصل به 4 بخش تقسیم شده است: سه بخش اول ادبیات داده، ادبیات مدل و ادبیات ارزیابی هستند. در هر بخش، مباحث مرتبط با آن حوزه به طور جداگانه و در ارتباط با پروژه حاضر مورد بررسی قرار می‌گیرند. همچنین بخش آخر، مربوط به بحث کالری مصرفی و به دست آوردن اطلاعات فنی محصول می‌باشد.

**2-۲- ادبیات داده**

بخش مهمی از این پایان‌نامه به ایجاد و توسعه مجموعه‌داده‌ای برای برچسب‌های محصولات داخلی اختصاص یافته است. در این مسیر، از ابزارها و کتابخانه‌های متعددی استفاده شده که هر یک نقش کلیدی در فرآیند برچسب زدن، استخراج و پیش‌پردازش داده‌ها ایفا کرده‌اند. این ابزارها و کتابخانه‌ها نه تنها برای جمع‌آوری داده‌ها، بلکه برای پاک‌سازی، اصلاح، و آماده‌سازی آنها جهت استفاده در مدل‌های مختلف به کار گرفته شده‌اند.

در ادامه، برخی از این ابزارها و کتابخانه‌ها به تفصیل معرفی خواهند شد و نقش و کاربرد دقیق هر یک از آن‌ها در این پایان‌نامه مورد بررسی قرار می‌گیرد. همچنین، به نحوه استفاده از این ابزارها در مراحل مختلف پروژه، از جمله استخراج داده‌های خام، پردازش اولیه، و آماده‌سازی نهایی برای تجزیه و تحلیل و مدل‌سازی، پرداخته می‌شود. هدف این بخش ارائه یک دیدگاه جامع از ابزارها و تکنیک‌هایی است که به پیشبرد تحقیق کمک کرده‌اند و نقش آنها در ایجاد یک مجموعه‌داده معتبر و قابل‌اعتماد برای زبان فارسی را تبیین می‌کند.

**2-2-1- فرایند برچسب گذاری تصویر[[6]](#footnote-6)**

فرایند برچسب‌گذاری تصویر یک مرحله مهم در آماده‌سازی داده‌ها برای پروژه‌های یادگیری ماشین، به‌ویژه در حوزه بینایی ماشین و تشخیص اشیاء، است. این فرایند شامل شناسایی و برچسب‌گذاری نواحی خاص در تصاویر است که مدل‌های یادگیری ماشین بر اساس آن آموزش می‌بینند. در ادامه، مراحل اصلی این فرایند را توضیح می‌دهم:

1. **جمع‌آوری تصاویر**

اولین مرحله شامل جمع‌آوری مجموعه‌ای از تصاویر مرتبط با هدف پروژه است. این تصاویر می‌توانند از منابع مختلفی مانند دوربین‌ها، پایگاه‌های داده عمومی، اینترنت، و غیره جمع‌آوری شوند.

1. **انتخاب ابزار برچسب‌گذاری**

ابزارهای فراوانی برای برچسب‌گذاری وجود دارد که چند نمونه در قسمت بعدی توضیح داده شده است.

1. **تعریف برچسب‌ها**

قبل از شروع برچسب‌گذاری، باید دسته‌بندی‌ها یا کلاس‌هایی که قرار است برچسب‌گذاری شوند، تعریف شود. برای مثال، در این پروژه، برچسب‌ها شامل برچسب ارزش غذایی، اعداد داخل برچسب، مواد مغذی، واحدهای اندازه گیری و پروانه بهداشتی ساخت می‌باشند.

1. **برچسب‌گذاری نواحی**

در این مرحله، کاربر با استفاده از ابزار برچسب‌گذاری، نواحی مربوط به هر دسته‌بندی را در تصاویر انتخاب می‌کند. این کار می‌تواند شامل ترسیم کادرهای محدودکننده[[7]](#footnote-7)، ماسک‌های تقسیم‌بندی[[8]](#footnote-8) یا نقاط کلیدی [[9]](#footnote-9)باشد که به شناسایی دقیق نواحی مورد نظر کمک می‌کند.

1. **بازبینی و تأیید**

پس از برچسب‌گذاری اولیه، نواحی برچسب‌گذاری‌شده بازبینی می‌شوند تا اطمینان حاصل شود که برچسب‌ها به درستی و با دقت اعمال شده‌اند. در پروژه‌های بزرگتر، این مرحله ممکن است توسط افراد مختلف یا با استفاده از الگوریتم‌های خودکار انجام شود.

1. **ذخیره‌سازی داده‌های برچسب‌گذاری‌شده**

پس از تأیید، برچسب‌ها به همراه تصویر در قالب‌های استاندارد مانند ایکس‌ام‌ال[[10]](#footnote-10)، جیسان[[11]](#footnote-11)، یا فایل‌های متنی ذخیره می‌شوند. این داده‌ها سپس برای آموزش مدل‌های یادگیری ماشین استفاده می‌شوند.

1. **آماده‌سازی برای آموزش مدل**

در نهایت، داده‌های برچسب‌گذاری‌شده برای آموزش مدل آماده می‌شوند. این مرحله شامل تقسیم‌بندی داده‌ها به مجموعه‌های آموزشی، اعتبارسنجی و آزمایش است که طبق ساختاری که برای آموزش هر مدل مشخص شده است، مرتب می‌شوند.

فرایند برچسب‌گذاری تصویر، یک گام اساسی برای اطمینان از دقت و عملکرد بالای مدل‌های یادگیری ماشین در پروژه‌های بینایی ماشین است. کیفیت داده‌های برچسب‌گذاری‌شده به طور مستقیم بر عملکرد نهایی مدل تأثیر می‌گذارد.

**2-2-2- پلتفرم[[12]](#footnote-12) روبوفلو[[13]](#footnote-13)**

روبوفلو یک پلتفرم جامع برای توسعه پروژه‌های بینایی ماشین است که به ویژه در بخش برچسب‌گذاری تصاویر بسیار قدرتمند است. این پلتفرم ابزارهای متنوعی برای برچسب‌گذاری دستی، نیمه‌خودکار، و خودکار ارائه می‌دهد که کاربران می‌توانند با آن‌ها به سادگی نواحی موردنظر در تصاویر را مشخص کنند. روبوفلو از انواع مدل‌های بینایی ماشین مانند تشخیص اشیاء، تقسیم‌بندی، و طبقه‌بندی پشتیبانی می‌کند.

همچنین، این پلتفرم امکاناتی برای مدیریت نسخه‌ها، همکاری تیمی، و بهینه‌سازی داده‌ها فراهم می‌کند. به لطف یکپارچگی با فریم‌ورک‌های یادگیری ماشین محبوب، داده‌های برچسب‌گذاری‌شده می‌توانند مستقیماً برای آموزش مدل‌ها استفاده شوند، که باعث افزایش دقت و سرعت فرآیند توسعه مدل‌های بینایی ماشین می‌شود.[2].

**2-2-3- پلتفرم لیبل استودیو[[14]](#footnote-14)**

لیبل استودیو یک پلتفرم متن‌باز**[[15]](#footnote-15)**  برای برچسب‌گذاری داده‌ها است که از انواع مختلف داده‌ها از جمله تصاویر، ویدئوها، متن‌ها و فایل‌های صوتی پشتیبانی می‌کند. این ابزار به کاربران امکان می‌دهد تا به راحتی پروژه‌های برچسب‌گذاری خود را مدیریت کرده و داده‌هایشان را برای مدل‌های یادگیری ماشین آماده کنند. با قابلیت‌های سفارشی‌سازی بالا، کاربران می‌توانند رابط کاربری و ابزارهای برچسب‌گذاری را متناسب با نیازهای خاص پروژه‌های خود تنظیم کنند.

یکی از ویژگی‌های برجسته این پلتفرم، امکان همکاری همزمان چندین کاربر در برچسب‌گذاری داده‌ها است که باعث افزایش سرعت و دقت کار می‌شود. مدیریت پروژه‌های برچسب‌گذاری نیز با این پلتفرم بسیار ساده‌تر می‌شود، به طوری که کاربران می‌توانند فرآیندهای کاری، تخصیص وظایف و مراحل بازبینی را به طور دقیق کنترل کنند. این ابزار همچنین با فریم‌ورک‌های یادگیری ماشین یکپارچه می‌شود، که داده‌های برچسب‌گذاری‌شده را مستقیماً برای آموزش مدل‌ها قابل استفاده می‌کند.

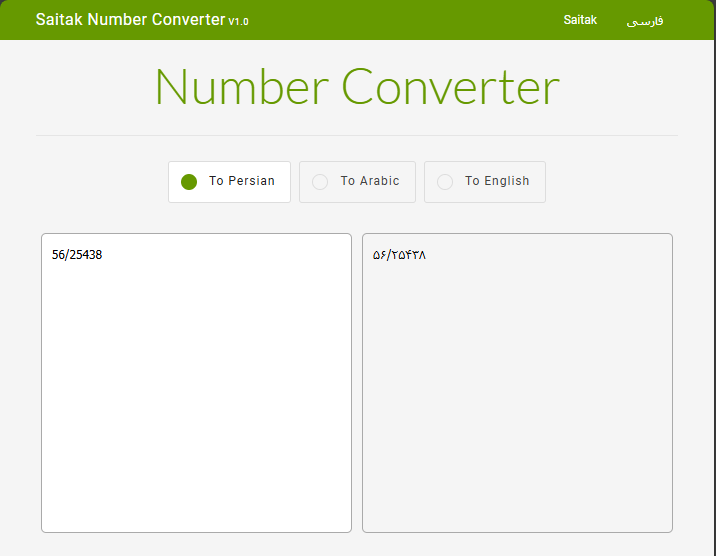
علاوه بر این، از افزونه‌ها و ای‌پی‌آی‌[[16]](#footnote-16)های متنوعی پشتیبانی می‌کند که امکان یکپارچگی با سایر سیستم‌ها و گسترش قابلیت‌های پلتفرم را فراهم می‌سازد. این ویژگی‌ها به کاربران اجازه می‌دهد تا این پلتفرم را با ابزارهای موجود خود ترکیب کرده و از آن در محیط‌های مختلف استفاده کنند[3].



تصویر 6- محیط لیبل استودیو و فرایند برچسب‌کذاری

**2-2-4- وب‌سایت سایتاک[[17]](#footnote-17)**

این وبسایت، بستری برای تبدیل کاراکتر اعداد انگلیسی به کاراکتر اعداد فارسی فراهم‌ می‌کند. این تبدیل از این جهت مفید است که در پلتفرم‌های برچسب گذاری، اعداد فارسی با استفاده از صفحه‌کلید، در دسترس نیستند و برای آموزش مدل‌ها، باید اعداد فارسی بر روی برچسب را، با استفاده از کاراکتر‌های فارسی نوشت[4].



تصویر 7- محیط وب‌سایت سایتاک

**2-2-5- پلتفرم اوپن‌سی‌وی[[18]](#footnote-18)**

این کتابخانه یک کتابخانه متن‌باز است که به طور گسترده برای برنامه‌های بینایی ماشین و پردازش تصویر استفاده می‌شود. این کتابخانه ابتدا توسط اینتل[[19]](#footnote-19) توسعه یافت و اکنون توسط یک جامعه کاربری بزرگ نگهداری می‌شود. اوپن‌سی‌وی شامل بیش از 2500 الگوریتم بهینه‌سازی شده است که برای تشخیص و ردیابی اشیاء، شناسایی چهره، تحلیل حرکات، و پردازش تصاویر مورد استفاده قرار می‌گیرند. یکی از ویژگی‌های برجسته این کتابخانه این است که از چندین زبان برنامه‌نویسی پشتیبانی می‌کند و همچنین با سیستم‌عامل‌های مختلفی مانند ویندوز، لینوکس[[20]](#footnote-20)، و مک‌او‌اس[[21]](#footnote-21) سازگار است.

اوپن‌سی‌وی در طیف وسیعی از کاربردها مانند بینایی ماشین، رباتیک، تحلیل ویدئو، و واقعیت افزوده استفاده می‌شود. از جمله قابلیت‌های این کتابخانه می‌توان به تشخیص لبه‌ها، فیلترگذاری تصاویر، شناسایی ویژگی‌ها، و پردازش ویدئو اشاره کرد. اوپن‌سی‌وی به دلیل سرعت و کارایی بالا در پردازش تصویر، در پروژه‌های تحقیقاتی و تجاری به طور گسترده‌ای مورد استفاده قرار می‌گیرد. این کتابخانه به توسعه‌دهندگان امکان می‌دهد تا با استفاده از ابزارهای قدرتمند و الگوریتم‌های از پیش ساخته شده، برنامه‌های پیچیده بینایی ماشین را به سرعت پیاده‌سازی کنند.

در این پروژه از اوپن‌سی‌وی عموماً برای خواندن عکس‌ها، نمایش اطلاعات خوانده شده بر روی هر عکس، بازبینی دقت و عملکرد مدل‌ها و ذخیره عکس‌ها استفاده شده‌است.

**2-2-6- کتابخانه پیل[[22]](#footnote-22)**

پیل یک کتابخانه قدرتمند و کاربردی برای پردازش تصاویر در پایتون است. این کتابخانه امکان باز کردن، تغییر، و ذخیره انواع مختلف تصاویر را فراهم می‌کند و از فرمت‌های تصویری متعددی مانند جی‌پی‌جی[[23]](#footnote-23)، پی‌ان‌جی[[24]](#footnote-24)، گیف[[25]](#footnote-25)، و بی‌ام‌پی[[26]](#footnote-26) پشتیبانی می‌کند. با استفاده از پیل، توسعه‌دهندگان می‌توانند عملیات مختلفی مانند تغییر اندازه، برش، فیلترگذاری، و تبدیل فرمت تصاویر را به سادگی انجام دهند. اگرچه توسعه رسمی پیل متوقف شده است، اما نسخه‌ای به‌روزتر و توسعه‌یافته به نام پیلو[[27]](#footnote-27) به عنوان جایگزین آن مورد استفاده قرار می‌گیرد و به طور گسترده‌ای در پروژه‌های پردازش تصویر استفاده می‌شود.

مشکل اصلی اوپن‌سی‌وی، نمایش کاراکترهای فارسی بر روی عکس است که باعث به هم ریختگی در حرف و ارقام می‌شود. این مشکل با استفاده از کتابخانه پیل برطرف شده و حروف و ارقام به ترتیب درست، بر روی عکس درج می‌شوند.

**2-2-7- کتابخانه نامپای[[28]](#footnote-28)**

نامپای، یک کتابخانه اساسی و محبوب در زبان برنامه‌نویسی پایتون است که برای محاسبات علمی و عددی استفاده می‌شود. نامپای به‌ویژه به دلیل ارائه آرایه‌های چندبعدی[[29]](#footnote-29) با عملکرد بالا شناخته شده است. این آرایه‌ها به توسعه‌دهندگان اجازه می‌دهند تا عملیات‌های پیچیده ریاضی و محاسباتی را به صورت کارآمد و سریع انجام دهند. برخلاف لیست‌های پایتون، آرایه‌های نامپای دارای ابعاد ثابت و نوع داده یکنواخت هستند که این ویژگی‌ها باعث بهبود سرعت و کاهش مصرف حافظه می‌شود. علاوه بر این، نامپیا شامل تعداد زیادی از توابع ریاضی برای انجام محاسبات مختلف مانند جبر خطی، تبدیل فوریه، و تولید اعداد تصادفی است.

نامپای به عنوان یکی از کتابخانه‌های پایه‌ای در عملیات داده‌کاوی[[30]](#footnote-30) و یادگیری ماشین پایتون عمل می‌کند و بسیاری از کتابخانه‌های دیگر به طور مستقیم بر پایه نامپای ساخته شده‌اند. این کتابخانه به دلیل قابلیت‌های گسترده و عملکرد بالا، به‌ویژه در پردازش داده‌های بزرگ و انجام تحلیل‌های عددی در پروژه‌های علمی و مهندسی مورد استفاده قرار می‌گیرد. به عنوان مثال، توسعه‌دهندگان می‌توانند با استفاده از نامپای ماتریس‌ها و بردارها را به سادگی مدیریت کنند و محاسبات پیچیده‌ای مانند ضرب ماتریسی، برون‌یابی، و انتگرال‌گیری عددی را انجام دهند.

**2-2-8- کتابخانه ره[[31]](#footnote-31)**

ره کتابخانه‌ای در پایتون است که برای کار با عبارات منظم[[32]](#footnote-32) طراحی شده است. این کتابخانه امکاناتی را برای جستجو، جایگزینی، و تطبیق الگوهای متنی فراهم می‌کند و به توسعه‌دهندگان اجازه می‌دهد تا با استفاده از الگوهای پیشرفته، به سرعت بخش‌های خاصی از متن‌ها را شناسایی و پردازش کنند. با این کتابخانه، می‌توان عملیات‌های مختلفی مانند بررسی وجود الگو، تقسیم متن بر اساس الگو، و جایگزینی متن‌های مطابق با الگو را به سادگی انجام داد. این ابزار به ویژه در پردازش داده‌ها، تحلیل متن، و اعتبارسنجی ورودی‌های کاربر کاربرد زیادی دارد.

**2-2-9- کتابخانه‌های اصلاح قالب[[33]](#footnote-33)**

عربیک‌ریشیپر[[34]](#footnote-34) و بیدی[[35]](#footnote-35) دو کتابخانه پایتون هستند که برای پردازش و نمایش صحیح متن‌های عربی و دیگر زبان‌های راست به چپ[[36]](#footnote-36) مانند فارسی استفاده می‌شوند. عربیک‌ریشیپر وظیفه تغییر شکل حروف عربی را به گونه‌ای دارد که آن‌ها به درستی به هم متصل شوند، زیرا حروف در این زبان‌ها بسته به موقعیتشان در کلمه، اشکال مختلفی دارند. از سوی دیگر، بیدی برای ترتیب‌گذاری صحیح متن‌های دوزبانه (شامل حروف از راست به چپ و چپ به راست) به کار می‌رود، به‌ویژه وقتی که اعداد یا کلمات لاتین در متن عربی یا فارسی قرار می‌گیرند. این دو کتابخانه معمولاً در کنار هم استفاده می‌شوند تا متن‌های عربی و فارسی به درستی در رابط‌های گرافیکی یا تصاویر نمایش داده شوند. این دو کتابخانه همراه با کتابخانه پیل، برای قرار دادن ارقام و حروف فارسی بر روی عکس، استفاده شده‌اند.

**2-2-10- کتابخانه‌‌ی جیسان**

جیسان کتابخانه‌ای در پایتون است که برای کار با داده‌های فرمت جیسان طراحی شده است. این کتابخانه امکاناتی را برای تبدیل داده‌های پایتون به فرمت جیسان و بالعکس فراهم می‌کند، به طوری که می‌توان به راحتی داده‌ها را بین پایتون و دیگر زبان‌های برنامه‌نویسی یا سیستم‌های مبتنی بر جیسان مبادله کرد. در این پروژه از جیسان برای خواندن داده‌های برچسب‌گذاری شده که در پلتفرم‌های مختلف آماده شده، استفاده می‌شود.

**3-۲- ادبیات مدل**

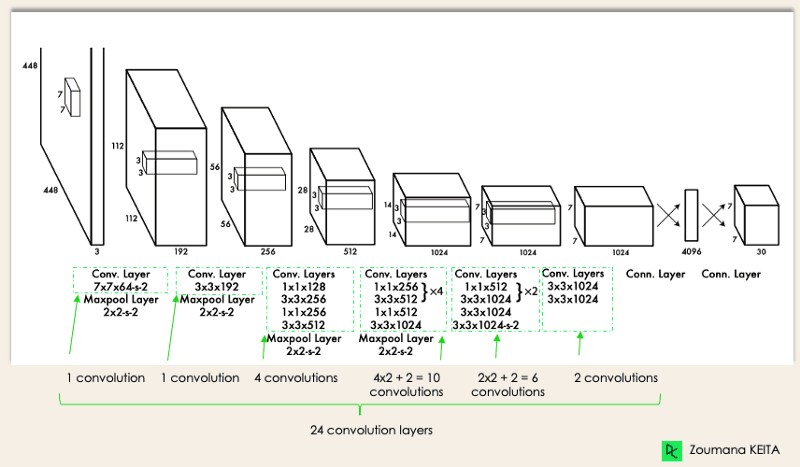
در این پایان‌نامه، برای اینکه بتوانیم حروف، کلمات و اعداد نوشته شده بر روی برچسب را بخوانیم، نیاز بود که مدل‌هایی را با استفاده از آنچه که گفته شد، آموزش دهیم و از آن استفاده کنیم. برای آموزش این مدل‌ها از ابزارها و کتابخانه‌هایی استفاده شد که در این بخش به معرفی آنها می‌پردازیم. همچنین هر آن چیزی که به سرعت بخشیدن و بهبود این فرایند کمک کرد را نیز معرفی خواهیم کرد.

**2-3-1- پلتفرم آلترالیتیکس[[37]](#footnote-37)**

آلترالیتیکسیک کتابخانه پایتون است که به ویژه برای کار با مدل‌های یادگیری عمیق در حوزه بینایی ماشین توسعه یافته است. این کتابخانه به خاطر پیاده‌سازی ساده و کاربردی از مدل‌های پیشرفته‌ی بینایی ماشین مانند یولو[[38]](#footnote-38) مشهور است. این کتابخانه، به توسعه‌دهندگان این امکان را می‌دهد که مدل‌های تشخیص اشیاء و دیگر وظایف بینایی ماشین را با استفاده از پیکربندی‌های ساده و استاندارد به سرعت آموزش دهند و اجرا کنند.

**2-3-2- الگوریتم یولو**

یولو یک الگوریتم پیشرفته برای تشخیص اشیاء در تصاویر و ویدئوها است که به دلیل سرعت و دقت بالای خود شناخته شده است. برخلاف روش‌های سنتی که تصویر را به بخش‌های کوچک‌تر تقسیم کرده و هر بخش را به صورت جداگانه تحلیل می‌کنند، یولو کل تصویر را به یکباره پردازش کرده و به شناسایی و مکان‌یابی اشیاء در آن می‌پردازد. این الگوریتم با استفاده از شبکه‌های عصبی کانولوشنی به سرعت اشیاء مختلف را در تصویر شناسایی کرده و جعبه‌های محدود کننده و برچسب‌های مربوط به هر شیء را پیش‌بینی می‌کند. یولو به دلیل عملکرد بالا و توانایی پردازش در زمان واقعی، در کاربردهایی مانند نظارت ویدئویی، رباتیک، و تحلیل تصاویر پزشکی بسیار مورد استفاده قرار می‌گیرد. ساختار کلی این مدل، در شکل زیر به صورت واضح قابل مشاهده است.



تصویر 8- ساختار کلی مدل یولو

در این پروژه از یولو ورژن 8 برای تشخیص برچسب ارزش مواد غذایی، استفاده شده است. این نسخه، به عنوان نسخه جدید و بهبود یافته الگوریتم یولو، به دلیل مزایای متعددی که ارائه می‌دهد، انتخاب مناسبی برای پروژه‌های بینایی ماشین است. این نسخه با بهینه‌سازی‌های الگوریتمی و معماری پیشرفته‌تر، دقت بالاتری در تشخیص اشیاء و سرعت پردازش سریع‌تری نسبت به نسخه‌های قبلی ارائه می‌دهد. به خاطر توانایی‌های بهبود یافته در شناسایی و مکان‌یابی دقیق اشیاء، قابلیت کار در زمان واقعی، و مقیاس‌پذیری برای پردازش داده‌های بزرگ و متنوع، به ویژه در پروژه‌هایی که نیاز به پردازش سریع و دقیق تصاویر دارند، بسیار مناسب است.

**2-3-3- پلتفرم پایتورچ[[39]](#footnote-39)**

پایتورچ یک کتابخانه متن‌باز و محبوب برای یادگیری ماشین و یادگیری عمیق است که توسط فیس‌بوک[[40]](#footnote-40) توسعه یافته است. این کتابخانه به دلیل سهولت استفاده و انعطاف‌پذیری بالا، به ویژه در میان پژوهشگران و توسعه‌دهندگان یادگیری عمیق محبوبیت زیادی دارد. پایتورچ از تنسور[[41]](#footnote-41) برای انجام محاسبات عددی استفاده می‌کند و با گراف‌[[42]](#footnote-42) محاسباتی پویا، امکان طراحی و آزمایش مدل‌های پیچیده را فراهم می‌سازد. این ویژگی‌ها به توسعه‌دهندگان کمک می‌کند تا به سرعت مدل‌های خود را توسعه داده و آزمایش کنند.

پایتورچ به دلیل سازگاری با ابزارها و کتابخانه‌های دیگر مانند نامپای[[43]](#footnote-43) و سایکیت‌لرن[[44]](#footnote-44) و همچنین پشتیبانی گسترده از سوی جامعه کاربری، یک انتخاب محبوب برای پروژه‌های یادگیری عمیق در حوزه‌هایی مانند پردازش تصویر و پردازش زبان طبیعی است. این کتابخانه به کاربران در سطوح مختلف اجازه می‌دهد تا به راحتی مدل‌های یادگیری ماشین را پیاده‌سازی و آموزش دهند، و در نتیجه در پروژه‌های تحقیقاتی و تجاری بسیار مورد استفاده قرار می‌گیرد.

در اين پايان‌نامه نيز براي پياده‌سازي مدل‌ها و استفاده از آن‌ها، از اين کتابخانه استفاده‌ شده است و مدل‌هایی که در این پروژه نام برده خواهند شد، در این فرمت ذخیره شده اند.

**2-3-4- پلتفرم محاسبات موازی کودا[[45]](#footnote-45)**

کودا یک پلتفرم محاسباتی موازی و رابط برنامه‌نویسی کاربردی است که توسط انویدیا[[46]](#footnote-46) توسعه یافته است. این فناوری به توسعه‌دهندگان اجازه می‌دهد تا از قدرت پردازشی بالای پردازنده‌های گرافیکی[[47]](#footnote-47) برای انجام محاسبات موازی و تسریع پردازش‌ها استفاده کنند. با بهره‌گیری از هزاران هسته پردازشی موجود در کارت گرافیک، کودا می‌تواند زمان محاسبات پیچیده را به طور قابل توجهی کاهش دهد، و این امر به ویژه در زمینه‌هایی مانند یادگیری ماشین و شبیه‌سازی‌های علمی کاربرد دارد.

کودا با زبان‌های برنامه‌نویسی مانند سی[[48]](#footnote-48)، سی‌پلاس‌پلاس[[49]](#footnote-49) و پایتون یکپارچه شده است و بسیاری از کتابخانه‌های یادگیری عمیق مانند تنسورفلو[[50]](#footnote-50) و پایتورچ از آن برای تسریع آموزش مدل‌ها استفاده می‌کنند. این تکنولوژی به توسعه‌دهندگان امکان می‌دهد تا از کارت گرافیک برای انجام محاسبات پیچیده و زمان‌بر به صورت کارآمدتر استفاده کنند، که در نهایت به بهبود عملکرد در کاربردهای مختلفی مانند تحلیل داده‌های بزرگ، رندرینگ[[51]](#footnote-51) گرافیکی و یادگیری عمیق منجر می‌شود.

**2-3-5- پلتفرم کگل[[52]](#footnote-52)**

کگل یک پلتفرم جامع برای تحلیل داده‌ها و مسابقات یادگیری ماشین است که به محققان، دانشمندان داده و توسعه‌دهندگان این امکان را می‌دهد که به صورت آنلاین به مجموعه‌های داده، ابزارهای تحلیلی، و جوامع فعال دسترسی پیدا کنند. کاربران کگل می‌توانند در مسابقات داده‌ای شرکت کنند، مدل‌های یادگیری ماشین را پیاده‌سازی کرده و با دیگران به اشتراک بگذارند. این پلتفرم به ویژه برای آموزش و آزمون مدل‌های مختلف مفید است و از طریق ارائه داده‌های معتبر و ابزارهای قدرتمند، به پیشبرد تحقیق و توسعه در زمینه‌های مختلف کمک می‌کند.

علاوه بر مسابقات، کگل یک محیط تعاملی برای جستجو و تحلیل داده‌ها ارائه می‌دهد که شامل نوت‌بوک[[53]](#footnote-53)‌ها و یک پایگاه داده بزرگ از مجموعه‌های داده عمومی است. کاربران می‌توانند با استفاده از زبان‌های برنامه‌نویسی مختلف مانند پایتون[[54]](#footnote-54) و آر[[55]](#footnote-55) به تحلیل داده‌ها بپردازند و کد‌ها و مدل‌های خود را بر روی سخت‌افزار‌‌های مختلف از جمله کارت گرافیک‌ها و پردازنده‌ها اجرا کنند. همچنین می‌توانند نتایج خود را با جامعه کگل به اشتراک بگذارند. این ویژگی‌ها به محققان و توسعه‌دهندگان این امکان را می‌دهد که به سرعت به داده‌های واقعی دسترسی پیدا کنند، مدل‌های خود را ارزیابی کنند و از بازخورد جامعه برای بهبود عملکرد مدل‌های خود بهره‌برداری کنند[5].

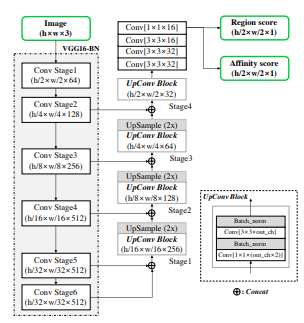
**2-3-6- پلتفرم ایزی‌او‌سی‌آر[[56]](#footnote-56)**

ایزی‌او‌سی‌آر یک کتابخانه منبع باز برای تشخیص و شناسایی متن از تصاویر است که با هدف ارائه یک راه‌حل ساده و سریع برای پردازش متن تصویری طراحی شده است. این کتابخانه به ویژه برای پروژه‌های بینایی ماشین که نیاز به استخراج متن از تصاویر دارند، مناسب است. ایزی‌او‌سی‌آر از مدل‌های پیش‌آموزش‌دیده استفاده می‌کند و به کاربران این امکان را می‌دهد که بدون نیاز به آموزش مدل‌های جدید، به راحتی متن‌ها را از تصاویر مختلف استخراج کنند. با پشتیبانی از زبان‌های مختلف و رابط کاربری ساده، این ابزار به توسعه‌دهندگان و محققان کمک می‌کند تا به سرعت و با دقت بالا متن‌های تصویری را شناسایی کنند.

این پلتفرم شامل دو مدل اصلی است. مدل تشخیص[[57]](#footnote-57) و مدل شناسایی[[58]](#footnote-58). مدل تشخیص مسئول شناسایی نواحی متنی در تصویر است و به تعیین مکان‌های دقیق متن‌ها کمک می‌کند. این مدل قادر است نواحی مختلف متن را به درستی تشخیص دهد، حتی در شرایط پیچیده مانند تغییرات در اندازه، فونت و زاویه متن. مدل شناسایی به دنبال آن، وظیفه شناسایی و استخراج محتوای متنی از نواحی مشخص‌شده را بر عهده دارد. این مدل متن‌های شناسایی شده را به صورت خوانا و قابل استفاده استخراج می‌کند، و به این ترتیب، مراحل پردازش تصویر و استخراج متن به طور کامل و دقیق انجام می‌شود. در ادامه به معرفی دو مدلی که در این پلتفرم استفاده شده می‌پردازیم[6].

1. **کرفت[[59]](#footnote-59)**

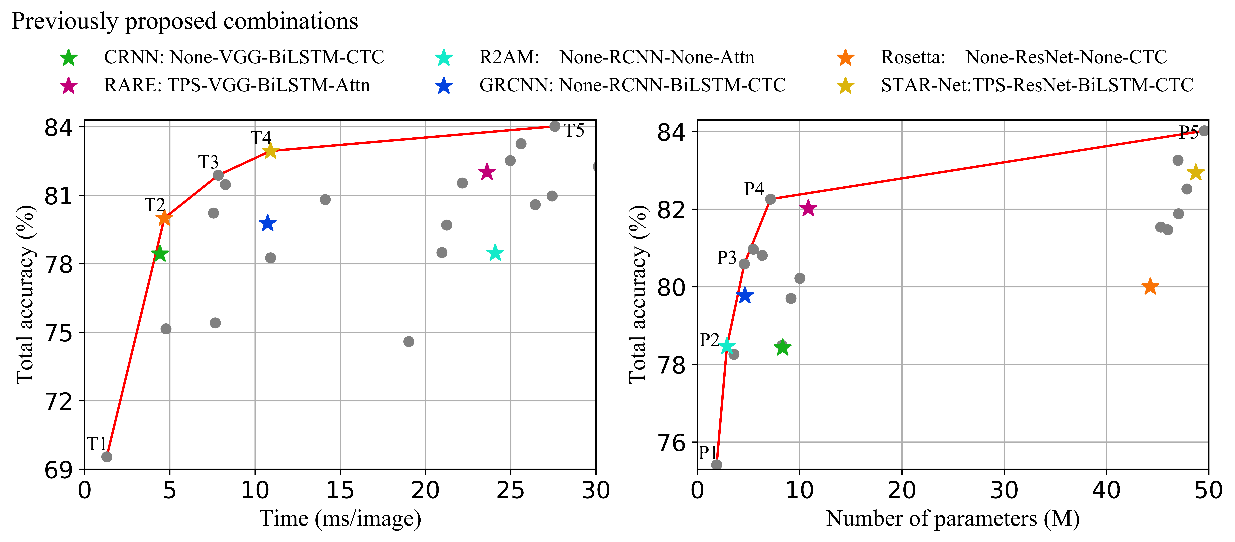
مدل تشخیص متن در ایزی‌او‌سی‌آر کرفت است. کرفتیک مدل پیشرفته برای شناسایی مناطق متنی در تصاویر است که به ویژه برای تشخیص متن در تصاویری با پس‌زمینه پیچیده و نویزهای زیاد طراحی شده است. این مدل با استفاده از تکنیک‌های یادگیری عمیق، نواحی متنی را با دقت بالا شناسایی می‌کند و قادر است متن‌های کوچک و نازک را که توسط مدل‌های دیگر ممکن است نادیده گرفته شوند، شناسایی کند[7].



تصویر 9- ساختار کرفت

1. **شناسایی عمیق متن[[60]](#footnote-60)**

یک مجموعه از مدل‌ها و ابزارها برای شناسایی و خواندن متن از تصاویر است که به منظور مقایسه عملکرد الگوریتم‌های مختلف شناسایی متن طراحی شده است. این شناساگر (TPS-ResNet-BiLSTM-Attn) نام دارد و یکی از مدل‌های مناسب جهت خواندن نوشته‌ها می‌باشد. عملکرد این مدل با سایر مدل‌ها در تصویر زیر مقایسه شده است[8].



تصویر 10- مقایسه سرعت و تعداد پارامترهای مدل خوانش متن

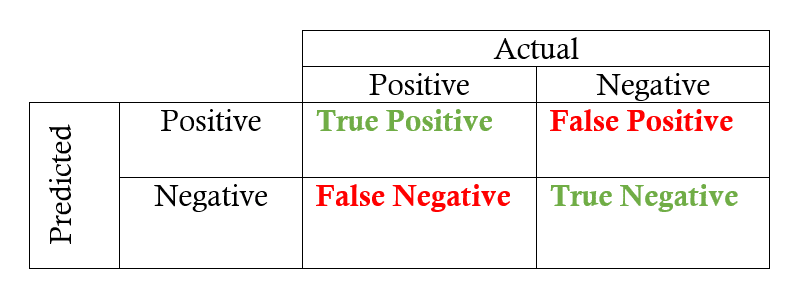
## 4-2- ادبیات ارزیابی

پس از اتمام آموزش مدل با داده‌های آموزشی، مرحله ارزیابی مدل با استفاده از معیارهای مختلف آغاز می‌شود. در این بخش، ابتدا به توضیح هر یک از این معیارها پرداخته می‌شود و سپس ابزارهایی که برای تهیه گزارش از عملکرد مدل به کار گرفته شده‌اند، معرفی خواهند شد.

**2-4-1- ماتریس درهم‌ریختگی[[61]](#footnote-61)**

این ماتریس، یکی از ابزارهای اصلی برای ارزیابی عملکرد مدل‌های طبقه‌بندی است. ماتریس درهم‌ریختگی چهار ترکیب مختلف از پیش‌بینی‌های مدل و مقادیر واقعی را در نظر می‌گیرد که در زیر توضیح داده شده و در ادامه مورد استفاده قرار می‌گیرد.

جدول 1- ماتریس درهم‌ریختگی



1. **True Positive (TP)** : تعداد نمونه‌هایی که واقعاً مثبت هستند و مدل به درستی آن‌ها را مثبت پیش‌بینی کرده است. این سلول در تصویر با رنگ سبز مشخص شده است.
2. **False Positive (FP)** : تعداد نمونه‌هایی که واقعاً منفی هستند اما مدل به اشتباه آن‌ها را مثبت پیش‌بینی کرده است. این وضعیت به عنوان **خطای نوع اول** نیز شناخته می‌شود و در تصویر با رنگ قرمز نمایش داده شده است.
3. **False Negative (FN)** : تعداد نمونه‌هایی که واقعاً مثبت هستند اما مدل به اشتباه آن‌ها را منفی پیش‌بینی کرده است. این وضعیت به عنوان **خطای نوع دوم** نیز شناخته می‌شود و در تصویر با رنگ قرمز نمایش داده شده است.
4. **True Negative (TN)** : تعداد نمونه‌هایی که واقعاً منفی هستند و مدل به درستی آن‌ها را منفی پیش‌بینی کرده است. این سلول نیز با رنگ سبز مشخص شده است.

**2-4-2- معیار دقت[[62]](#footnote-62)**

دقت کلی نشان می‌دهد که مدل به چه میزان در کل پیش‌بینی‌هایش درست عمل کرده است. این معیار نسبت تعداد پیش‌بینی‌های صحیح (چه مثبت و چه منفی) را به کل پیش‌بینی‌ها محاسبه می‌کند. دقت کلی معیار مناسبی زمانی است که داده‌ها دارای توازن خوبی بین کلاس‌ها هستند. این معیار در فرمول زیر نشان داده شده.

**2-4-3- معیار صحت[[63]](#footnote-63)**

صحت نشان می‌دهد که از میان تمام نمونه‌هایی که به عنوان مثبت پیش‌بینی شده‌اند، چه درصدی واقعاً مثبت بوده‌اند. این معیار زمانی مهم است که هزینه‌ی بالایی برای FP وجود دارد. این معیار در فرمول زیر نشان داده شده.

**2-4-4- معیار بازخوانی یا حساسیت[[64]](#footnote-64)**

بازخوانی نشان می‌دهد که مدل از میان تمامی نمونه‌های مثبت واقعی، چه تعداد را به درستی شناسایی کرده است. این معیار زمانی اهمیت دارد که هزینه‌ی بالایی برای از دست دادن نمونه‌های مثبت وجود دارد. فرمول ریاضیاتی این معیار در زیر آمده است.

**2-4-5- معیار F1**

F1 به عنوان میانگین هارمونیک[[65]](#footnote-65) صحت و بازخوانی تعریف می‌شود. این معیار تلاش می‌کند تا یک توازن بین صحت و دقت برقرار کند. اگر دقت و بازخوانی هر دو بالا باشند، F1 نیز بالا خواهد بود، و اگر یکی از این دو پایین باشد، F1 نیز پایین خواهد بود.

به عبارت دیگر، F1 برای مواقعی مناسب است که می‌خواهیم یک معیار واحد داشته باشیم که هم به درستی پیش‌بینی‌های مثبت و هم به میزان پوشش صحیح نمونه‌های مثبت اهمیت دهد.

**2-4-6- میانگین خطای نرمالیزه شده[[66]](#footnote-66)**

میانگین خطا، معیاری است که برای تعیین کمیت تفاوت دو دنباله (مانند رشته ها) از یکدیگر با شمارش حداقل تعداد عملیات مورد نیاز برای تبدیل یک رشته به رشته دیگر استفاده می شود. این عملیات معمولاً شامل درج، حذف و جایگزینی است. با نرمال کردن این فرایند، این معیار مقداری بین 0 و 1 پیدا می‌کند. این نرمال‌سازی کمک می کند که مقایسه بین معیارها راحت‌تر باشد.

**2-4-7- کتابخانه مت‌پلات‌لیب[[67]](#footnote-67)**

متپلات‌لیب یک کتابخانه قدرتمند در زبان برنامه‌نویسی پایتون است که برای مصورسازی داده‌ها و ایجاد نمودارهای متنوع به کار می‌رود. این کتابخانه به برنامه‌نویسان این امکان را می‌دهد که به‌راحتی داده‌ها را تحلیل کنند، الگوها و تغییرات را مشاهده نمایند و اطلاعات را به صورت بصری و قابل فهم نمایش دهند. با استفاده از متپلات‌لیب، می‌توان انواع نمودارهای خطی[[68]](#footnote-68)، دایره‌ای[[69]](#footnote-69)، میله‌ای[[70]](#footnote-70)، نقطه‌ای[[71]](#footnote-71) و سایر نمودارها را ایجاد کرد و آنها را با استفاده از تنظیمات ظاهری و قابلیت‌های تعاملی مختلف بهینه‌سازی نمود. این کتابخانه به دلیل کاربرد گسترده‌ای که در علوم داده، مهندسی، آمار و بسیاری از حوزه‌های دیگر دارد، به یک ابزار ضروری برای تجزیه و تحلیل داده‌ها و ارائه نتایج به صورت گرافیکی تبدیل شده است.

در این پایان‌نامه نیز از متپلات‌لیب در ارزیابی مدل‌ها استفاده شده است. برای بسیاری از تصمیم‌گیری‌ها نیاز به رسم نمودارهایی با استفاده از این کتابخانه بود و تمامی نمودارهایی که در فصل‌های بعدی به نمایش درآمده‌اند، با استفاده از متپلات‌لیب ایجاد شده‌اند.

## 5-2- تجزیه و تحلیل اطلاعات

پس از جمع آوری داده‌های درج شده بر روی برچسب ارزش غذایی محصولات، نوبت به تجزه و تحلیل محصول می‌رسد. در این بخش، ابتدا مفهوم اصلی در این بخش را توضیح داده و سپس ابزارهایی که در این پروسه به ما کمک کردند، معرفی خواهند شد.

**2-5-1- محاسبه کالری مصرفی**

مهم ترین مفهوم در این بخش، معادل متابولیک[[72]](#footnote-72) می‌باشد. معادل متابولیک، یک معیار استاندارد است که برای اندازه‌گیری شدت فعالیت‌های فیزیکی به کار می‌رود. یک MET به میزان مصرف انرژی در حالت استراحت بدن اشاره دارد. این معیار به شما کمک می‌کند تا بفهمید که یک فعالیت چقدر انرژی نسبت به حالت استراحت مصرف می‌کند[9, 10].

**مفهوم MET**

* **یکMET:** برابر با مصرف انرژی در حالت استراحت است که به طور تقریبی معادل 1 کیلوکالری (کالری) در هر کیلوگرم از وزن بدن در هر ساعت است.
* **دو METs:** به این معنا است که شما در حال انجام فعالیتی هستید که دو برابر انرژی حالت استراحت مصرف می‌کند.

به عنوان مثال، راه رفتن با سرعت متوسط ممکن است 4 MET داشته باشد، به این معنا که چهار برابر انرژی حالت استراحت مصرف می‌کند.

مقدار کالری مصرف شده در یک دقیقه فعالیت، با توجه به معدله زیر به دست می‌آید.

**2-5-2- کتابخانه جینجا[[73]](#footnote-73)**

جینجا یک موتور قالب‌سازی قدرتمند و محبوب برای پایتون است که به توسعه‌دهندگان امکان ایجاد قالب‌های پویا برای تولید اچ‌تی‌ام‌ال[[74]](#footnote-74)، ایکس‌ام‌ال[[75]](#footnote-75) و دیگر فرمت‌های متنی را می‌دهد. با استفاده از جینجا، می‌توان داده‌ها را به‌سادگی درون قالب‌ها جاگذاری کرد و صفحات وب پویا ایجاد نمود. این کتابخانه با داشتن سینتکسی[[76]](#footnote-76) شبیه به پایتون، فیلترها و توابع از پیش تعریف‌شده، و پشتیبانی از دستورات کنترلی مانند حلقه‌ها و شرط‌ها، انعطاف‌پذیری بالایی در طراحی قالب‌ها فراهم می‌کند.

جینجا اغلب در فریم‌ورک‌های وب مانند فلسک[[77]](#footnote-77) برای رندر کردن صفحات وب استفاده می‌شود. ویژگی‌های امنیتی آن، مانند خودکارسازی فرار برای جلوگیری از حملات XSS، آن را به گزینه‌ای امن برای توسعه صفحات وب تبدیل می‌کند. به‌طور کلی، جینجا با سادگی و قدرتی که ارائه می‌دهد، یک ابزار اساسی برای توسعه‌دهندگان پایتون در پروژه‌های وب محسوب می‌شود.

**2-5-3- کتابخانه سلنیوم[[78]](#footnote-78)**

سلنیوم یک ابزار قدرتمند برای خودکارسازی مرورگرهای وب است که به‌طور گسترده توسط توسعه‌دهندگان و تست‌کنندگان نرم‌افزار استفاده می‌شود. این کتابخانه به شما امکان می‌دهد تعاملات مختلف با مرورگرها را شبیه‌سازی کنید، مانند کلیک کردن بر روی دکمه‌ها، پر کردن فرم‌ها، جابه‌جا شدن بین صفحات، و استخراج داده‌ها از وب‌سایت‌ها. سلنیوم از مرورگرهای مختلفی مانند کروم، فایر‌فاکس، سافاری و اج پشتیبانی می‌کند و می‌تواند با استفاده از زبان‌های برنامه‌نویسی مختلف از جمله پایتون، جاوا، سی‌شارپ و روبی کنترل شود.

سلنیوم به‌ویژه برای تست خودکار وب‌سایت‌ها بسیار مفید است. با استفاده از این ابزار، تست‌کنندگان می‌توانند سناریوهای پیچیده‌ای را که کاربران واقعی ممکن است تجربه کنند، شبیه‌سازی کرده و صحت عملکرد سایت را بررسی کنند. به‌طور کلی، سلنیوم ابزاری اساسی برای هر کسی است که نیاز به خودکارسازی تعاملات با مرورگرهای وب دارد.

## 6-2- جمع بندی:

همان‌طور که پیش‌تر بیان شد، این پایان‌نامه به چهار بخش مجزا با عناوین داده، مدل، ارزیابی و تجزیه و تحلیل اطلاعات تقسیم شده است. در این فصل، برخی از مفاهیم مرتبط با این بخش‌ها معرفی و بررسی شدند. در فصل‌های بعد، با استفاده از این مبانی به تشریح فرآیند انجام کار در این پایان‌نامه خواهیم پرداخت.

# فصل سوم پیشینه پژوهش

## ۱-3- مقدمه

این فصل به بررسی تحقیقات و پروژه‌های پیشین در زمینه پردازش تصویرو تشخیص نوری حروف[[79]](#footnote-79) می‌پردازد. در این زمینه، تحقیقات زیادی صورت نگرفته است، از دلایل آن می‌توان به تفاوت بین چاپ برچسب ارزش مواد غذایی بر روی محصولات مختلف و استفاده از رنگ‌ها با کد‌های رنگی مختلف نام برد. چند نمونه تحقیق در این زمینه بر روی برچسب محصولات آمریکایی[[80]](#footnote-80) انجام شده و نمونه این پویشگر بر روی محصولات ایرانی، وجود ندارد.

این فصل، چارچوبی را فراهم می‌کند تا نشان دهد چگونه پروژه فعلی می‌تواند به پیشرفت دانش موجود در این زمینه و گسترش آن به زبان فارسی کمک کند و راهکارهایی نوآورانه برای حل مشکلات و بهبود تجربه کاربری ارائه دهد.

## ۲-3- تاریخچه و توسعه OCR

تشخیص نوری حروف یکی از تکنولوژی‌های کلیدی در پردازش تصویر است که برای استخراج متون از تصاویر استفاده می‌شود. تکنولوژی OCR از دهه 1950 توسعه یافته و به مرور زمان با بهبودهای مداوم در دقت و کارایی روبرو شده است. از جمله پیشرفت‌های اخیر می‌توان به استفاده از الگوریتم‌های یادگیری عمیق و شبکه‌های عصبی مصنوعی اشاره کرد که دقت تشخیص حروف را به طور قابل توجهی افزایش داده‌اند. به عنوان مثال، الگوریتم‌هایی نظیر Tesseract که به صورت متن باز توسعه یافته‌اند، قادر به تشخیص متون در چندین زبان و با دقت بالایی هستند.

مدل‌های یادگیری عمیق، مانند شبکه‌های عصبی کانولوشنی[[81]](#footnote-81)و شبکه‌های بازگشتی[[82]](#footnote-82)، به دلیل دقت بالا در تشخیص و طبقه‌بندی تصاویر، به طور گسترده‌ای در OCR مورد استفاده قرار گرفته‌اند. مدل‌های مدرنی نظیر MaskOCR و TransOCR از جدیدترین الگوریتم‌ها در این حوزه هستند که دقت‌های بسیار بالایی در تشخیص متن از تصاویر دارند. این مدل‌ها با استفاده از مجموعه داده‌های بزرگ و متنوع، قادر به تشخیص دقیق حروف در شرایط مختلف، نظیر نویز بالا و کیفیت پایین تصاویر، هستند

## 3-3-مروری بر مطالعات مرتبط

مطالعات و کارهای نسبتاً کمی در زمینه پردازش تصویر و بینایی ماشین بر روی برچسب ارزش محصولات غذایی انجام گرفته است که همه این مطالعات بر روی برچسب‌هایی غیر از برچسبی فارسی پویش می‌شود بوده. بعضی از این محصولات به صورت نرم افزار اندرویدی[[83]](#footnote-83) بوده[11]، و برخی دیگر مقالات علمی[12-14].

این برنامه‌ها راه‌های مختلفی برای به دست آوردن ورودی کاربر از اطلاعات تغذیه‌ای دارند. یکی از راه حل‌های این برنامه‌ها، استفاده از اطلاعاتی است که از قبل توسط مجموعه ای از کاربران جمع آوری شده است. مصرف‌کننده می‌توانند محصول خود را با نام جستجو کند تا آمار مربوطه را بیابد، اما داده های جمع‌آوری‌شده ممکن است کمبود داشته باشند.

اطلاعات، ممکن است برای هر محصول تکراری باشد و این اطلاعات بایکدیگر تطابق نداشته باشند که باعث گیج شدن کاربر می‌شود. همچنین اطلاعات ممکن است کاملاً نادرست باشد زیرا کاربران به صورت دستی آنها را وارد کرده و هیچ تعهدی نسبت به وارد کردن ‌آن‌ها ندارند. راه‌حل‌های دیگری که پیاده‌سازی شده، حول ساخت یک پایگاه‌داده از محصولات شناخته‌شده می‌باشد.

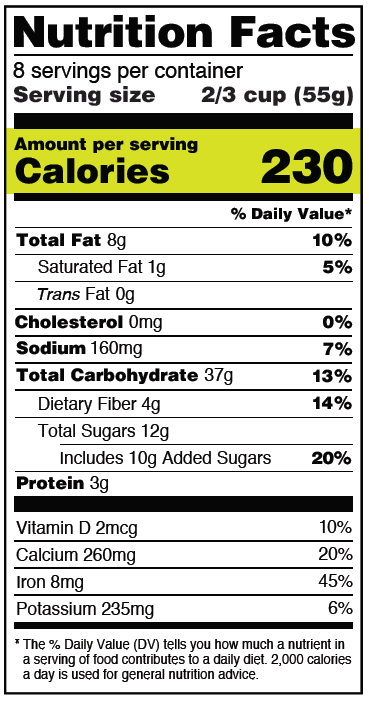
داده‌های تغذیه‌ای پایگاه‌داده محور، مستلزم جمع آوری تمام داده‌ها در یک مکان واحد است. داده‌های در حال تغییر باید دائما به روز شوند. این رویکرد به حجم زیادی از زیرساخت، همکاری و نگهداری نیاز دارد. اگر داده ها حفظ نشوند یا محصولات جدیدی به آن اضافه نشود، داده‌هایی که کاربر به دنبال آن است ممکن است وجود نداشته باشد یا قدیمی باشد.

آن دسته از مقالاتی که تمرکز بر توسعه برنامه‌ای بر پایه استخراج اطلاعات از برچسب ارزش مواد غذایی داشتند، روندی مانند روند نوشته شده در زیر را دنبال کردند.

1. الگوریتم[[84]](#footnote-84)، یک عکس را به عنوان ورودی می‌گیرد.
2. برچسب ارزش مواد غذایی در عکس یافت شده و بریده می‌شود.
3. عکس به بخش‌های مختلفی تقسیم می‌شود که هر بخش نشانگر یک کلمه می‌باشد.
4. بخش‌های مختلف که عکس به آنها شکسته شد، خوانده شده تا عدد یا کلمه از آن استخراج شود.
5. خروجی به دست می‌آید و به صورت لیستی از اطلاعات نمایش داده می‌شود.

داده‌های استفاده شده در این مقالات، بسیار محدود تر از داده‌‌های موجود در الگوریتم معرفی شده در این مقاله می‌باشد. یکی از مقالات[12] به جای استفاده از OCR برای خواندن کلمات، از کلاس‌بند[[85]](#footnote-85) استفاده کرده به این دلیل که کلمات محدود بوده و نیاز به خواندن ندارند. ولی این کار باعث می‌شود که اگر یک کلاس‌بندی به اشتباه انجام شد، آن ماده مغذی به اشتباه مقدار بگیرد اما با استفاده از OCR، اگر مشکلی در خواندن پیش بیاید، یک یا دو حرف عوض شده و می‌توان از سایر حروف، کلمه را حدث زد. البته اگر تعداد عکس‌های مجموعه داده زیاد باشد، مشکل OCR تا حدودی رفع خواهد شد.

این تحقیقات عمدتاً برو روی برچسب محصولات آمریکایی انجام شده است که به شکل زیر است.



تصویر 11- برچسب ارزش مواد غذایی آمریکایی

این برچسب شامل اطلاعات بیشتری نسبت به اطلاعات بر روی محصولات غذایی ایرانی دارد. این اطلاعات شامل، کالری هر سهم از محصول، تعداد وعده ها به ازای هر محصول، انواع چربی‌ها، کلسترول[[86]](#footnote-86)، سدیم[[87]](#footnote-87)، کربوهیدرات کل[[88]](#footnote-88)، پروتئین[[89]](#footnote-89)، ویتامین د[[90]](#footnote-90)، کلسیم[[91]](#footnote-91)، آهن[[92]](#footnote-92)و پتاسیم[[93]](#footnote-93)، می‌باشد. همچنین مقادیر این مواد مغذی هم به میزان وزنی و هم به درصد نوشته شده است.

یکی از تفاوت های اصلی برچسب‌های آمریکایی و ایرانی در رنگی بودن پس‌زمینه مواد مغذی می‌باشد. این رنگی بودن، کار خواندن را کمی دشوار تر از برچسب‌های آمریکایی کرده به این دلیل که کد رنگی[[94]](#footnote-94) هر رنگ ممکن است با کد رنگی متن نوشته شده به سختی قابل تمایز بخشیدن باشد و مدل[[95]](#footnote-95) باید داده‌های زیادی را ببیند تا بتواند بین رنگ‌ها و کدهای رنگی تمایز قائل شود.

در برخی مطالعات انجام شده[14]، تمرکز بر روی خواندن برچسب با چولگی تا 30-45 درجه می‌باشد. همچنین قابلیت خواندن بارکد نیز با همین چولگی نیز میسر است. اما در این پروژه، تمرکز فقط بر روی خود برچسب بدون چولگی بیش از حد می‌باشد.

## 4-3- جمع بندی

این فصل به بررسی تحقیقات پیشین در زمینه پردازش تصویر و تشخیص نوری حروف پرداخته و چالش‌های مرتبط با برچسب‌های ارزش غذایی محصولات را مورد بررسی قرار می‌دهد. با توجه به کمبود تحقیقات در این زمینه، به ویژه برای محصولات با برچسب‌های فارسی، تلاش‌های انجام شده در این پروژه می‌تواند به پیشرفت دانش موجود و گسترش کاربردهای اوسی‌آر در زبان فارسی کمک کند. همچنین، مرور تحقیقات مرتبط نشان می‌دهد که سیستم‌های فعلی دارای محدودیت‌هایی هستند، از جمله عدم تطابق داده‌ها و نیاز به نگهداری پایگاه داده‌ها. این پروژه با استفاده از الگوریتم‌های جدید و مجموعه داده‌های متنوع، به دنبال ارائه راهکارهایی نوآورانه برای بهبود دقت و کارایی سیستم‌های اوسی‌آر در تشخیص برچسب‌های ارزش غذایی است.

# فصل چهارم ساختار و آموزش مدل‌ها

## ۱-4- مقدمه

در این فصل، به بررسی ساختار و معماری مدل‌های مورد استفاده در این پروژه می‌پردازیم و نحوه انتخاب و تنظیم پارامترهای این مدل‌ها را شرح می‌دهیم. سپس، به چگونگی آماده‌سازی داده‌ها برای آموزش مدل‌ها اشاره خواهیم کرد. در ادامه، فرآیند آموزش مدل‌ها، شامل تنظیمات اولیه، مراحل آموزش، و ارزیابی عملکرد مدل‌ها با استفاده از داده‌های آزمایشی توضیح داده خواهد شد. همچنین، چالش‌های احتمالی در طول آموزش و راهکارهای به‌کاررفته برای بهینه‌سازی عملکرد مدل‌ها مورد بحث قرار می‌گیرد.

## 2-4- مدل تشخیص برچسب

همانطور که گفته شد، برای تشخیص برچسب ارزش مواد غذایی، از مدلی به نام یولو ورژن 8 استفاده شد. در این قسمت به نحوه آموزش و همچنین آماده سازی داده برای این مدل می‌پردازیم.

برای مدل اولیه، از Yolov8s استفاده شد. چندین نوع مدل وجود دارد که برای آموزش باید از یکی از این مدل‌ها استفاده کرد. این مدل ها به شرح زیر می‌باشند.

جدول 2- انواع مدل‌های یولو

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model** | **size (pixels)** | **mAPval 50-95** | **Speed CPU ONNX (ms)** | **Speed A100 TensorRT (ms)** | **params (M)** | **FLOPs (B)** |
| [YOLOv8n](https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8n.pt) | 640 | 37.3 | 80.4 | 0.99 | 3.2 | 8.7 |
| [YOLOv8s](https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8s.pt) | 640 | 44.9 | 128.4 | 1.20 | 11.2 | 28.6 |
| [YOLOv8m](https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8m.pt) | 640 | 50.2 | 234.7 | 1.83 | 25.9 | 78.9 |
| [YOLOv8l](https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8l.pt) | 640 | 52.9 | 375.2 | 2.39 | 43.7 | 165.2 |
| [YOLOv8x](https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8x.pt) | 640 | 53.9 | 479.1 | 3.53 | 68.2 | 257.8 |

این جدولی است که در گیت‌هاب[[96]](#footnote-96) یولو قرار دارد[15]. به دلیل راحتی کار، و وجود فقط یک کلاس در هر عکس (برچسب مواد غذایی) کار مدل راحت است، پس نیاز به مدل‌های سنگین نیست و در این پروژه مدل s انتخاب شده که ترکیبی از دقت مناسب و سرعت قابل قبول را دارد.

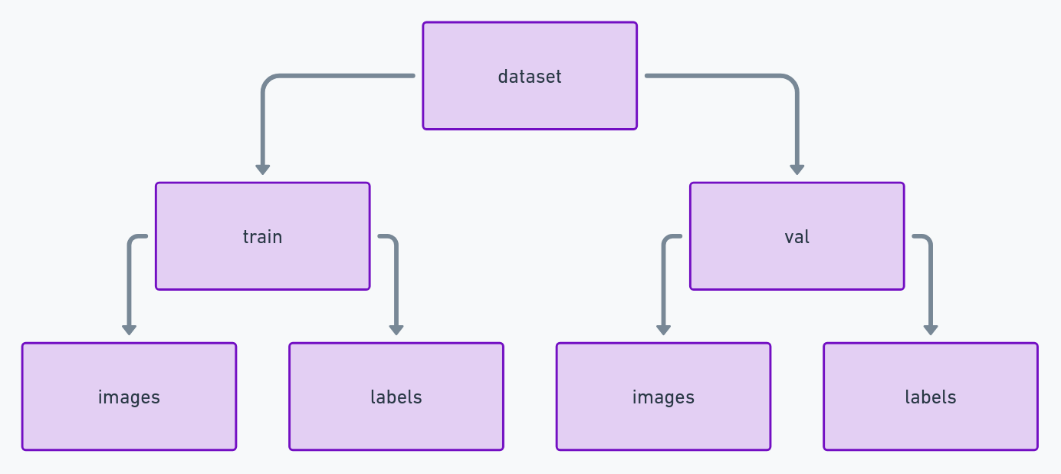
حال برای اینکه سرعت و دقت بیشتری در فرایند آموزش به مدل بدهیم، یک مدل از قبل آموزش دیده بر روی مجموعه داده کوکو[[97]](#footnote-97) استفاده می‌کنیم. مجموعه داده کوکو یک مجموعه داده با 80 کلاس است. هیچ کدام از این کلاس‌ها، مربوط به برچسب نیست، اما وزن‌های مدل، در این حالت نزدیک تر به وزن‌های بعد از آموزش هستند، از این رو آموزشی سریع و با دقت خواهیم داشت.

آرگومان‌هایی[[98]](#footnote-98) که در آموزش مدل استفاده شد به این صورت هستند.

* ایپاک[[99]](#footnote-99): تعداد کل دورهای آموزشی را نشان می‌دهد که هر دوره نشان دهنده یک عبور کامل از کل مجموعه داده است. تنظیم این مقدار می تواند بر مدت زمان تمرین و عملکرد مدل تاثیر بگذارد. در اینجا تعداد کل دورها، 20 تنظیم شده است.
* اندازه دسته[[100]](#footnote-100): با سه حالت تنظیم به عنوان یک عدد صحیح (به عنوان مثال، دسته=16)، حالت خودکار برای 60٪ استفاده از حافظه کارت گرافیک (دسته=1-)، یا حالت خودکار با کسر استفاده مشخص (دسته=7/0). در این پروژه، اندازه دسته 32 درنظر گرفته شده است.
* نرخ یادگیری[[101]](#footnote-101): تنظیم این مقدار برای فرآیند بهینه‌سازی بسیار مهم است و بر سرعت به‌روزرسانی وزن‌های مدل تأثیر می‌گذارد. در اینجا نرخ یادگیری 01/0 تنظیم شده است.

آرگومان‌هایی که یولو می‌گیرد، بیش از 30 تا است، این سه آرگومان دستی تنظیم شدند و مابقی به صورت پیش‌فرض باقی ماند.

هر مدل هوش مصنوعی، قالب دریافت مجموعه داده خود را دارد. یولو نیز از این قضیه مستثنی نیست. قالبی که یولو برای مجموعه داده دریافت می‌کند، به صورت درختی، به شکل زیر است.



تصویر 12- قالب قابل قبول یولو

و برچسب‌ها باید به صورت یک فایل تکست[[102]](#footnote-102) برای عکس متناظر در پوشه عکس باشند. در ادامه مثالی برای درک بهتر خواهیم زد.



تصویر 13- عملیات تشخیص برچسب

همان‌طور که دیده می‌شود، شماره کلاس در گوشه سمت چپ و بالای تصویر نوشته شده است. همچنین برای مشخص کردن قاب، از 4 عدد استفاده می‌شود. طول و عرض نقطه بالا و چپ قاب و طول و عرض نقطه پایین سمت راست قاب. حال محتوای فایل برچسب را در ادامه مشاهده خواهید کرد.



تصویر 14- یک ردیف از برچسب هر عکس

در هر ردیف، 5 عدد وجود دارد. عدد اول شماره کلاس، که در اینجا چون فقط یک کلاس موجود است (کلاس برچسب ارزش محصول) پس کلاس صفر، نشان دهنده آن است. چهار عددی همه که برای نشان دادن قاب می‌باشد به ترتیبی که ذکر شد، در اینجا نوشته می‌شود.

فایل data.yaml فایل است که در داخل پوشه Dataset قرار می‌گیرد و آدرس پوشه‌های train، test و validation داخل آن قرار می‌گیرد.

**3-4- مدل تشخیص کلمات**

همانطور که در فصل‌های گذشته اشاره شد، برای تشخیص کلمات از مدل کرفت استفاده شده است. تشخیص کلمات داخل برچسب از این رو مهم است که ما نیاز نداریم تمامی کلمات و جملات داخل برچسب را بخوانیم و فقط آن بخشی که برای ما مهم است، خوانده می‌شود.

برای مثال، شکل زیر عملکرد مدل کرفت قبل از آموزش بر روی مجموعه داده جدید را نشان می‌دهد.



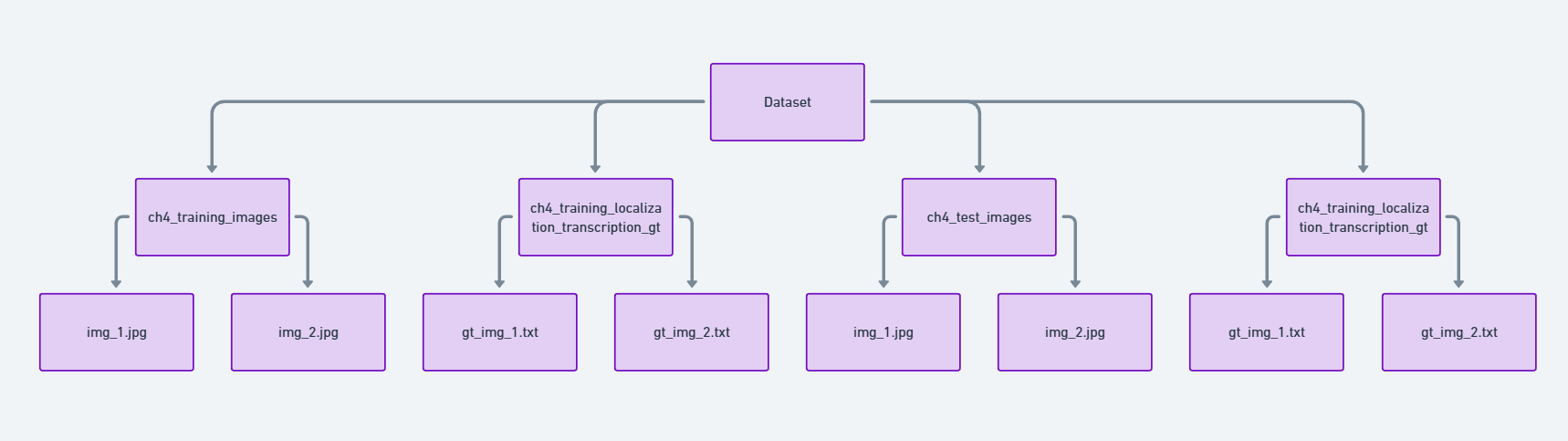
تصویر 15- مدل کرفت، قبل از آموزش

همان‌طور که دیده می‌شود، جملات کم اهمیت مانند «متوسط»، «زیاد» و «IFDA» که به آن‌ها نیازی نداریم خوانده شده اند اما کلمات و اعداد کلیدی مانند «قند» و «نمک» و یکسری از اعداد که رکن اساسی این پروژه محسوب می‎‌شوند، خوانده نشده است. از این رو نیاز داریم که این مدل را به آن صورت که مد‌نظر خودمان هست، آموزش دهیم.

فایل data.yaml این مدل، حاوی آرگومان‌های ورودی هنگام آموزش می‌باشد که چند مورد از این آرگومان‌هایی که در تنظیم مدل‌مهم‌تر هستند، توضیح داده می‌شود.

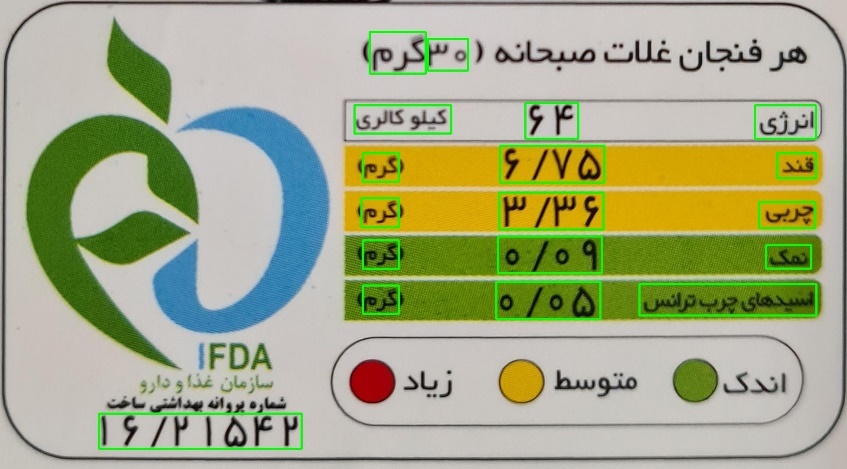
* مدل پیش‌زمینه[[103]](#footnote-103) : این پارامتر مشخص می‌کند که کدام مدل پیش‌زمینه برای استخراج ویژگی‌ها استفاده می‌شود. در اینجا، از معماری وی‌جی‌جی[[104]](#footnote-104) استفاده شده است که یک انتخاب رایج و کارآمد برای مدل‌های تشخیص متن مانند کرفت است.
* مسیر فایل مدل پیش‌آموزش‌دیده[[105]](#footnote-105): استفاده از مدل پیش‌آموزش‌دیده بر روی داده‌های قبلی، می‌تواند باعث تسریع فرایند آموزش و افزایش دقت مدل نهایی شود. در اینجا مدل‌ پیش‌فرضی که در گیت‌هاب کرفت بود[[106]](#footnote-106)، به عنوان مدل پیش‌آموزش‌دیده انتخاب گردید.
* نرخ یادگیری: همان‌طور که گفته شد، یکی از حیاتی‌ترین پارامترها در فرآیند آموزش مدل‌های یادگیری عمیق است. این پارامتر سرعت به‌روزرسانی وزن‌های شبکه در طول آموزش را تعیین می‌کند و تأثیر مستقیم بر روی همگرایی و عملکرد نهایی مدل دارد و در اینجا 001/0 قرار گرفت که پیش‌فرض خود مدل بود.
* اندازه دسته: این پارامتر نیز توضیح داده شد و نشان‌دهنده تعداد نمونه‌هایی است که در هر تکرار از آموزش پردازش می‌شوند. این پارامتر در اینجا 5 در نظر گرفته شد.
* تعداد دور: تعداد دورهای آموزشی که برای این مدل در نظر گرفته شده، 10 هزار دور است.

قالب مجموعه داده ای که در این مدل استفاده شد، با مدل قبلی کمی متفاوت است. ابتدا تصویری برای درک بهتر مجموعه داده در زیر نشان داده شده است.

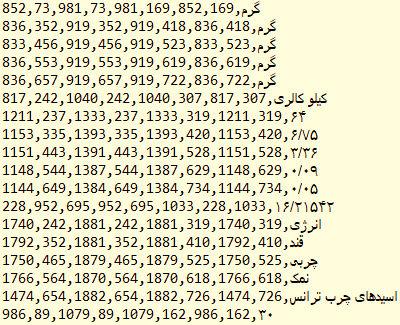


تصویر 16- قالب قابل قبول کرفت

در این مجموعه داده چهار پوشه وجود دارد. دو پوشه برای آموزش و دو پوشه برای ارزیابی مدل. یکی از دو پوشه، حاوی تصاویر می‌باشد و دیگری شامل فایل‌های برچسب که به فرمت تکست هستند. در ادامه یک نمونه از تصاویر و برچسب‌ها نشان داده شده است.



تصویر 17- تشخیص کلمات و ارقام به وصیله کرفت

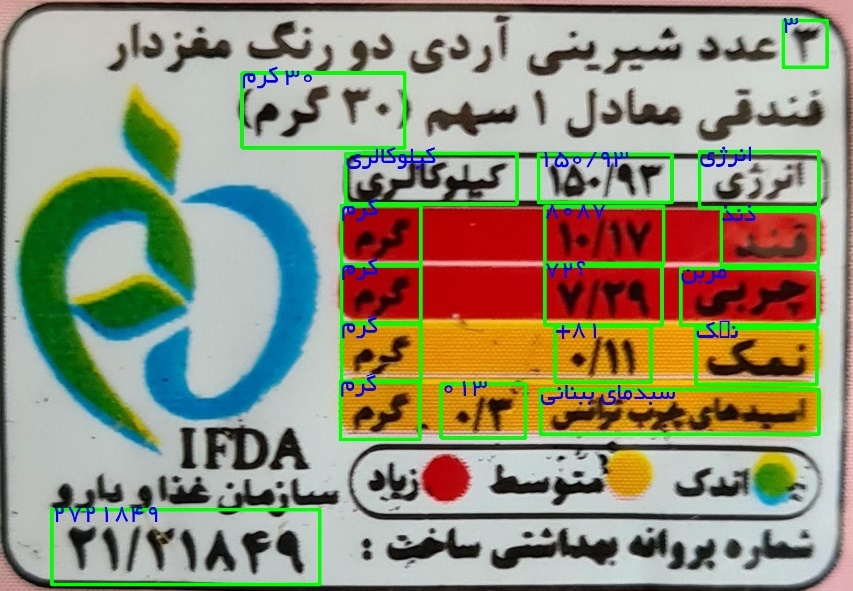


تصویر 18- برچسب‌های تصویر قبل

در تصویر اول 18 عدد کادر سبزرنگ وجود دارد که هرکدام نشان دهنده یک ردیف از فایل تکستی است که در تصویر بعدی نشان داده شده است.، می‌باشد. هر ردیف از برچسب، 9 ویژگی دارد. 8 ویژگی اول به ترتیب دو به دو، طول و عرض گوشه های سمت چپ بالا، سمت راست بالا، سمت راست پایین و سمت چپ پایین می‌باشند. ویژگی آخر نشان دهنده نوشته‌ی داخل هر کادر است. این نوشته به تشخیص کلمه کمک می‌کند.

**4-4- مدل خواندن کلمات**

گام آخر در استخراج اطلاعات برچسب ارزش مواد غذایی، خواندن اطلاعاتیست که مدل استخراج کلمات به دست آورده. این اطلاعات با استفاده از یک مدل او‌سی‌آر خوانده می‌شود که قبلاً به آن اشاره شد. این مدل ازپیش آموزش دیده، عملکرد ضعیفی بر روی داده‌های مجموعه داده این پروژه دارد. در اینجا عملکرد این مدل، قبل از آموزش را مشاهده می‌کنیم.

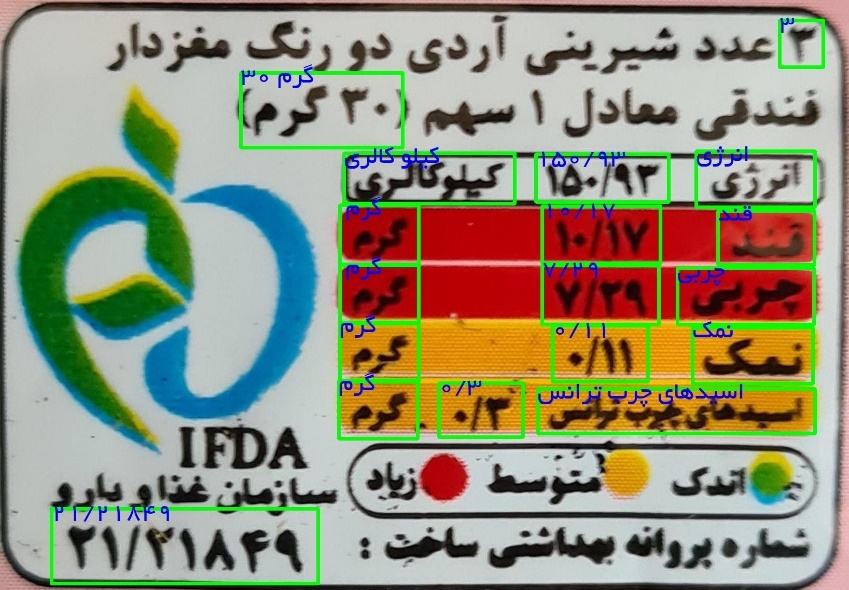


تصویر 19- خروجی مدل اوسی‌آر قبل از آموزش

همانطور که در این تصویر مشاهده می‌کنید، عملکرد مدل ازپیش آماده ضعیف است، دلایل مختلفی می‌تواند داشته باشد. مهم‌ترین دلایل به شرح زیر است.

* رنگ پس‌زمینه: اعداد و ارقام با رنگ پس‌زمینه به خوبی تشخیص داده نمی‌شوند زیرا تشخیص کد رنگی این رنگ‌ها دشوارتر است و مدل از قبل بر روی این پس‌زمینه‌ها آموزش ندیده.
* فونت مختلف: با اینکه در مصوبه‌ی درج این برچسب، فونت و اندازه نوشتار تایین شده، اما برخی از شرکت‌ها با فونت‌هایی غیر از فونت ذکر شده، اطلاعات را درج می‌کنند.
* داشتن نویز یا مات بودن: بعضی از برچسب‌ها دارای نویز هستند یا برخی اوقات عکاس، با لغزش دست عکسی را گرفته که موجب مات شدن تصویر شده است. در تصویر بالا نویز را روی دو کلمه‌ی «چرب ترانس». همچنین در این تصویر، برخی از حروف کمی مات هستند.

حال تصویر زیر که خروجی مدل، پس از آموزش است را مشاهده می‌کنیم.



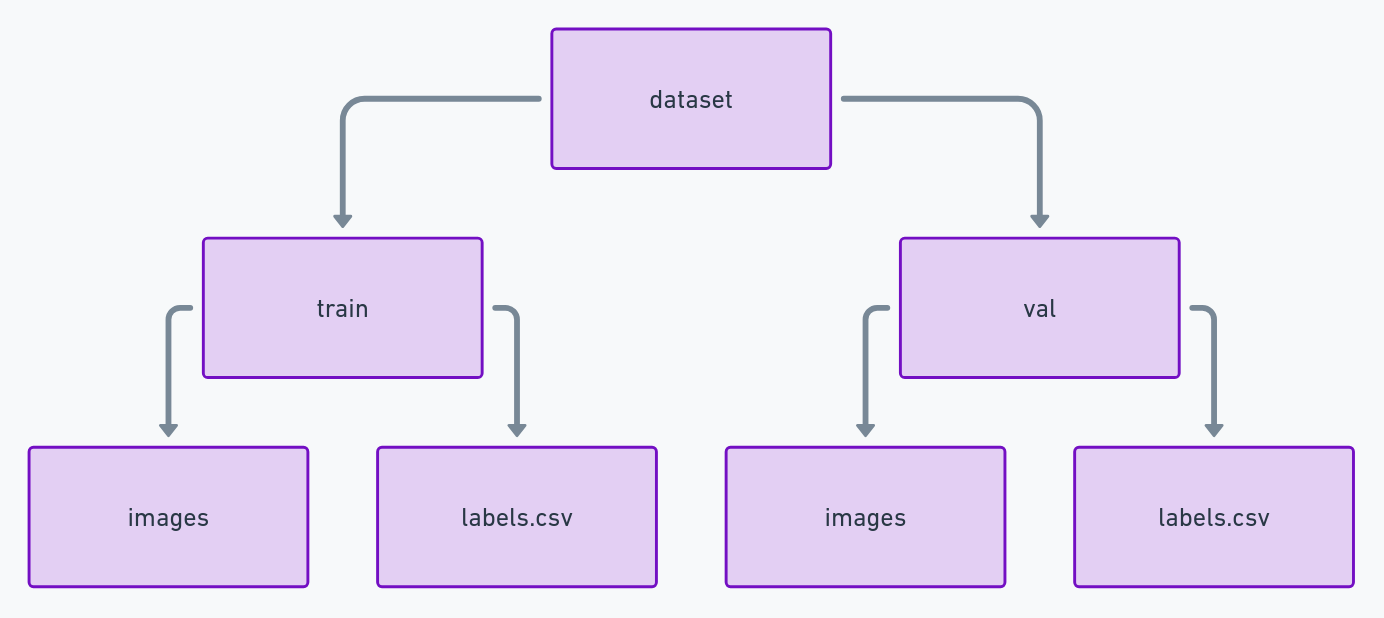
تصویر 20- خروجی مدل اوسی‌آر بعد از آموزش

فایل data.yaml این مدل، تقریباً ساختاری مشابه مدل قبل دارد. آرگومان‌های مهمی که در این مدل وجود دارند و قبلا ذکر نشند عبارت اند از:

* دور آموزشی: تعداد دور آموزشی‌ای که برای این مدل در نظر گرفته شده است، 6 هزار دور می‌باشد.
* مجموعه اعداد موجود: این آرگومان، یک رشته از تمامی اعدادی است که قرار است در برچسب ارزش مواد غذایی خوانده شوند. این رشته به این صورت است: «0123456789۰۱۲۳۴۵۶۷۸۹»
* مجموعه علائم موجود: این آرگومان، یک رشته از تمامی علائمی است که قرار است در برچسب ارزش مواد غذایی خوانده شوند. این رشته به این صورت است: «/.\_-»
* مجموعه حروف موجود: این آرگومان، یک رشته از تمامی حروفی است که قرار است در برچسب ارزش مواد غذایی خوانده شوند. این رشته به این صورت است: «‌grkKcal ملی‌صفر‌ک‌اسیظ‌هاچب‌تس‌قدگم‌کوی بنژ»

در این مجموعه همانطور که مشاهده می‌شود، تمام حروف وجود ندارند و به ترتیب خاصی کنار یکدیگر قرار گرفته اند. این رشته به نحوی تنظیم شده است که تمام کاراکترهای مورد نیاز برای پردازش متون موجود در برچسب‌های ارزش غذایی را به خوبی پوشش دهد. بعضی از حروف دوبار تکرار شده‌اند، زیرا هر حرف ممکن است شکل‌های خاصی در کلمات به خود بگیرد.

تصویر زیر، نشان دهنده قالب مجموعه داده مناسب، برای آموزش مدل او‌سی‌آر است.



تصویر 21- قالب قابل قبول مدل اوسی‌آر

در این مجموعه داده دو پوشه آموزش و ارزیابی و در هر پوشه، تصاویر و یک فایل سی‌اس‌وی[[107]](#footnote-107) وجود دارند. تصاویر مربوط به کلماتی است که در برچسب‌های ارزش مواد غذایی وجود دارند و از تصاویر استخراج شده اند. فایل سی‌اس‌وی شامل دو ستون است. ستون آدرس و کلمه. آدرس هرکدام از این عکس‌ها در ستون آدرس و نوشته داخل هر عکس، داخل ستون کلمه نوشته می‌شود.

## 5-4- جمع بندی

در این فصل، فرآیند استفاده از مدل‌های استفاده شده در این پروژه برای تشخیص و خواندن برچسب‌های ارزش مواد غذایی توضیح داده شده است. مدل تشخیص برچسب، برای تشخیص برچسب‌ها به‌کار گرفته شد و با استفاده از یک مدل پیش‌آموزش‌دیده، دقت و سرعت آن بهینه شد. سپس، مدل یافتن کلمات، برای تشخیص کلمات کلیدی داخل برچسب‌ها مورد استفاده قرار گرفت و عملکرد آن قبل و بعد از آموزش بررسی شد. در نهایت، مدل خوانش متن، برای خواندن متون استخراج‌شده به‌کار رفت و چالش‌های مربوط به تشخیص حروف و اعداد بررسی و نتایج بهبود یافته‌ی مدل پس از آموزش ارائه شد.

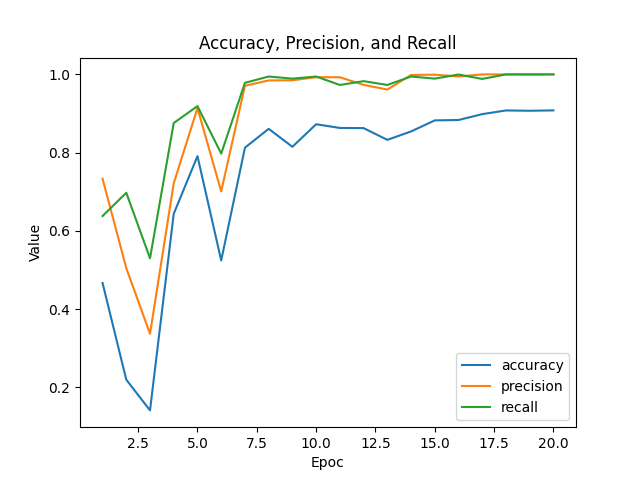
# فصل پنجم ارزیابی مدل‌ها و استفاده از اطلاعات

## ۱-5- مقدمه

این فصل به بررسی نتایج به‌دست‌آمده و معیارهای ارزیابی این پروژه اختصاص دارد. در این بخش، به‌منظور ارزیابی دقیق عملکرد مدل، از چندین معیار کلیدی استفاده شده است که شامل دقت ، صحت ، و بازخوانی می‌باشند. این معیارها به‌طور خاص برای سنجش توانایی مدل در پردازش و تحلیل داده‌ها مورد استفاده قرار گرفته‌اند. لازم به ذکر است که تمامی دقت‌های گزارش‌شده در این فصل، مرتبط با داده‌های آزمون هستند که 20 درصد از کل داده‌های اصلی پروژه را تشکیل می‌دهند. ارزیابی دقیق مدل با استفاده از این داده‌ها، امکان تحلیل جامع‌تری از عملکرد مدل در شرایط واقعی را فراهم می‌سازد و به تعیین نقاط قوت و ضعف آن کمک می‌کند. همچنین، نتایج این ارزیابی‌ها می‌تواند راه‌گشای بهینه‌سازی‌های آتی و بهبود مدل‌های مشابه در پروژه‌های آینده باشد.

## ۲-5- ارزیابی مدل تشخیص برچسب

نمودار زیر، سه معیار دقت، صحت و بازخوانی را برای مدل یولو ورژن 8 که وضیفه جدا کردن برچسب ارزش محصولات غذایی از عکس داشت، را نشان می‌دهد.



تصویر 22- نمودار سه معیار دقت، صحت و بازخوانی برای مدل یولو

این نمودار نشان می‌دهد که مدل پس از چند دوره آموزشی، عملکرد بسیار خوبی را از نظر دقت، صحت و بازخوانی به دست آورده است. اگرچه در دوره‌های اولیه نوساناتی دیده می‌شود، اما با ادامه آموزش، مدل به سرعت یاد می‌گیرد و این نوسانات کاهش می‌یابند. در نهایت، مدل به یک عملکرد پایدار و بالا در تمامی این معیارها می‌رسد، که نشان‌دهنده‌ی آموزش موفقیت‌آمیز مدل است.

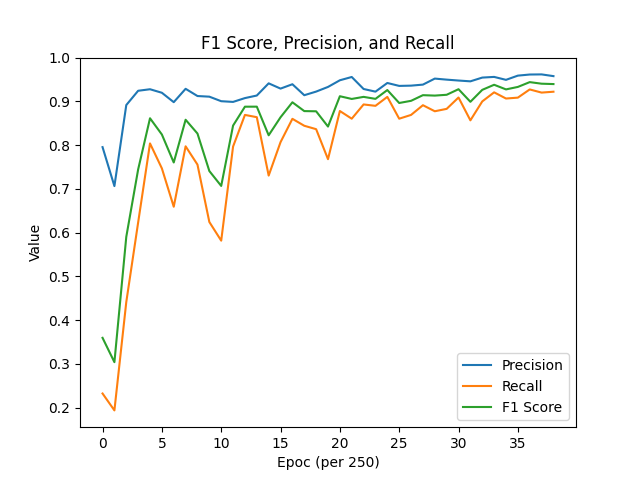
اعداد دقیق اولین و آخرین دوره آموزشی این مدل، در جدول زیر نمایان است.

جدول 3- مقایسه آمار اولین و آخرین دوره آموزشی در یولو

|  |  |  |
| --- | --- | --- |
| آخرین دوره | اولین دوره |  |
| 7%/90 | 6%/46 | دقت |
| 8%/97 | 3%/73 | صحت |
| 1%/98 | 7%/63 | بازخوانی |

## 3-5- ارزیابی مدل تشخیص کلمات

نمودار زیر، سه معیار دقت، صحت و بازخوانی را برای مدل کرفت که وضیفه خواندن کلمات مدنظر بر روی برچسب ارزش محصولات غذایی را داشت، را نشان می‌دهد.



تصویر 23- نمودار سه معیار دقت، صحت و میانگین هارمونیک برای مدل کرفت

نمودار نشان‌دهنده تغییرات صحت، بازخوانی و میانگین هارمونیک در طول 10 هزار دوره آموزش مدل است. در ابتدای آموزش، هر سه معیار به سرعت بهبود یافته و به مقادیر بالاتری دست می‌یابند. صحت (خط آبی) معمولاً بالاتر از بازخوانی (خط نارنجی) است و در اکثر دوره‌ها به بیش از 9/0 می‌رسد. در مقابل، بازخوانی در ابتدای آموزش نوسانات بیشتری دارد اما به تدریج به ثبات رسیده و نزدیک به 9/0 می‌شود.F1 score (خط سبز) که میانگین هارمونیک این دو معیار است، به طور پیوسته بهبود می‌یابد و بین صحت و بازخوانی قرار دارد. به طور کلی، این نمودار نشان می‌دهد که مدل با گذشت زمان به دقت و قابلیت بازیابی مناسبی دست یافته است.

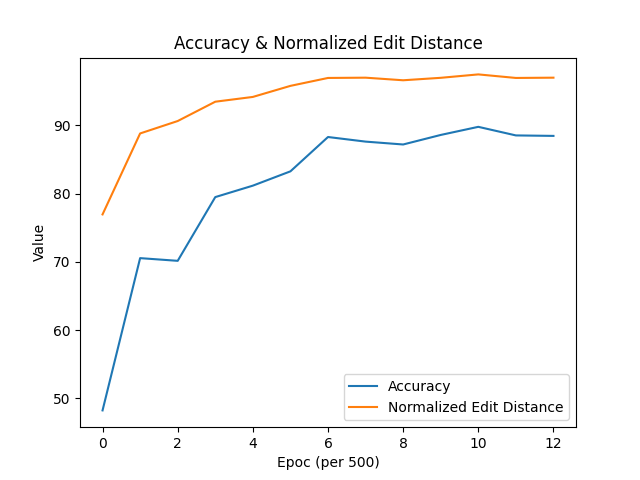
اعداد دقیق اولین و آخرین دوره آموزشی این مدل، در جدول زیر نمایان است.

جدول 4- مقایسه آمار اولین و آخرین دوره آموزشی در کرفت

|  |  |  |
| --- | --- | --- |
| آخرین دوره | اولین دوره |  |
| 2%/92 | 9%/35 | میانگین هارمونیک |
| 7%/95 | 5%/79 | صحت |
| 9%/93 | 2%/23 | بازخوانی |

## 4-5- ارزیابی مدل خوانش حروف و ارقام

نمودار زیر، رابطه بین دو معیار عملکرد مدل یعنی دقت و میانگین خطای نرمالیزه شده را در طول تعداد دور‌های مختلف نمایش می‌دهد.



تصویر 24- نمودار معیارهای دقت و میانگین خطای نرمالیزه شده در OCR

* **میانگین خطای نرمالیزه‌شده:** این مقدار با افزایش تعداد ایپاک‌ها ابتدا به سرعت افزایش می‌یابد و سپس به یک حد ثابتی نزدیک به ۹6 درصد می‌رسد. این نشان می‌دهد که مدل به تدریج در حال بهبود است و خطاهای نرمالیزه شده کمتر می‌شود، اما بعد از چند دوره به یک ثبات نسبی می‌رسد.
* **دقت:** دقت نیز به تدریج افزایش می‌یابد و از حدود 48 درصد در ابتدا به بیش از ۸8 درصد می‌رسد. پس از یک سری دوره‌ها، دقت نیز به یک مقدار نسبتا ثابت می‌رسد.

اما دلیل اینکه دقت از میانگین خطای نرمالیزه‌شده کمتر است، چیست؟

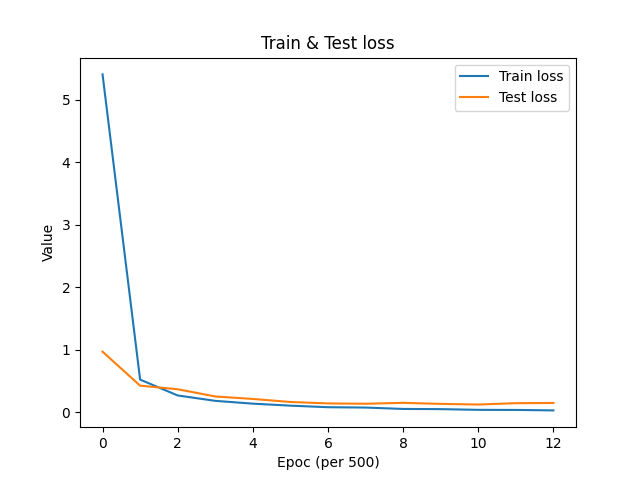
* **دقت** یک معیار بسیار سخت‌گیرانه است. اگر حتی یک کاراکتر در پیش‌بینی مدل اشتباه باشد، کل پیش‌بینی به عنوان اشتباه در نظر گرفته می‌شود و در نتیجه دقت کاهش می‌یابد.
* **میانگین خطای نرمالیزه‌شده** انعطاف‌پذیرتر است. این معیار با محاسبه فاصله ویرایشی، میزان نزدیکی پیش‌بینی به جواب درست را اندازه‌گیری می‌کند. بنابراین، اگر پیش‌بینی به جواب درست نزدیک باشد، حتی اگر کاملاً صحیح نباشد، همچنان مقدار نسبتاً بالایی برای این معیار به دست می‌آید.

اعداد دقیق اولین و آخرین دوره آموزشی این مدل، در جدول زیر نمایان است.

جدول 5- مقایسه آمار دقت و میانگین خطای نرمالیزه‌شده در اولین و آخرین دوره

|  |  |  |
| --- | --- | --- |
| آخرین دوره | اولین دوره |  |
| 4%/88 | 2%/48 | دقت |
| 9%/96 | 9%/76 | میانگین خطای نرمالیزه‌شده |

تابع زیان این مدل برا داده‌های آموزش و آزمایش نیز در ادامه به نمایش درآمده اند.



تصویر 25- تابع زیان داده‌های آموزش و آزمایش برای مدل OCR

هر دو خط مربوط به توابع زیان، روند کاهشی را نشان می‌دهند که این نشان دهنده مناسب بود آموزش و کمتر شدن خطای به وجود آمده در مدل است. مشاهده می‌شود که پس از یک تعداد دور، مقدار تابع زیان برای داده‌ها تغییری نکرده است و این درست جاییست که دیگر نیاز به آموزش نداریم چون با آموزش بیشتر، مدل بیش‌برازش می‌شود و نمی‌تواند داده‌های جدید را درست پیش‌بینی کند.

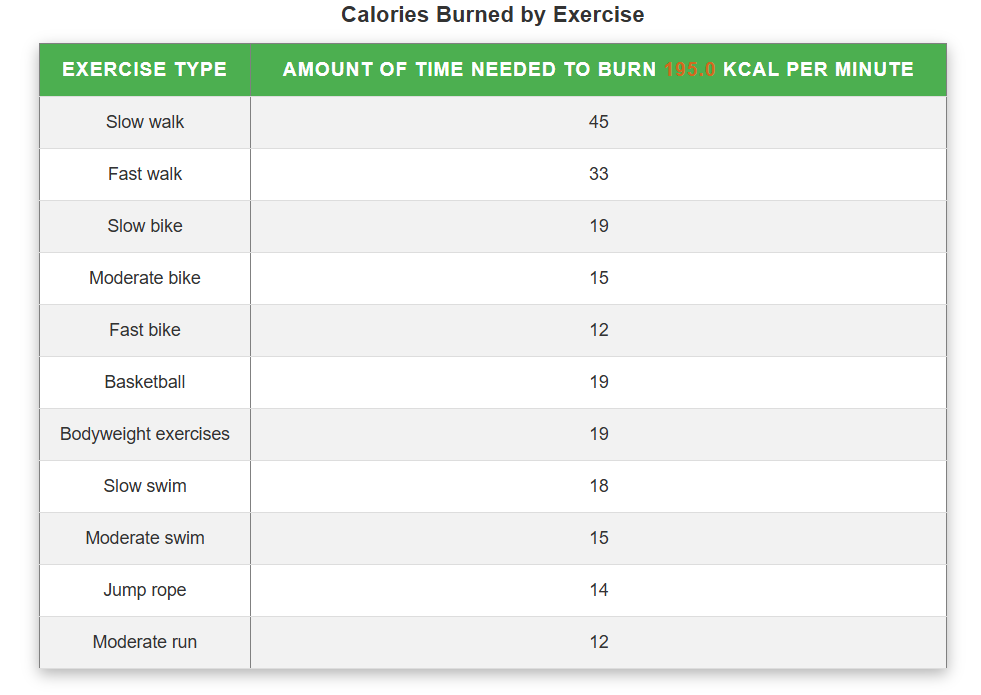
جدول 6- مقایسه آمار مقادیر توابع زیان داده‌های آموزش و آزمایش در اولین و آخرین دوره

|  |  |  |
| --- | --- | --- |
| آخرین دوره | اولین دوره |  |
| 04/0 | 40/5 | آموزش |
| 14/0 | 97/0 | آزمایش |

## 5-5- تجزیه و تحلیل انرژی محصول

در این پروژه، بعد از اینکه مقادیر و کلمات نوشته شده بر روی برچسب ارزش غذایی خوانده شدند، حال با توجه به موقعیت کلمه «انرژی» در تصویر پیدا شد، عدد نوشته شده در سطر این کلمه، به عنوان انرژی هر سهم از محصول پیدا می‌شود. حال با استفاده از این عدد و مقدارMET مخصوص هر فعالیت بدنی و با فرض وزن 75 کیلوگرم، در یک جدول اچ‌تی‌ام‌ال، اطلاعات مربوط به زمان مورد نیاز هر فعالیت بدنی برای سوزاندن آن سهم از محصول، ذکر شده است.

برای نمونه به جدول زیر دقت کنید.

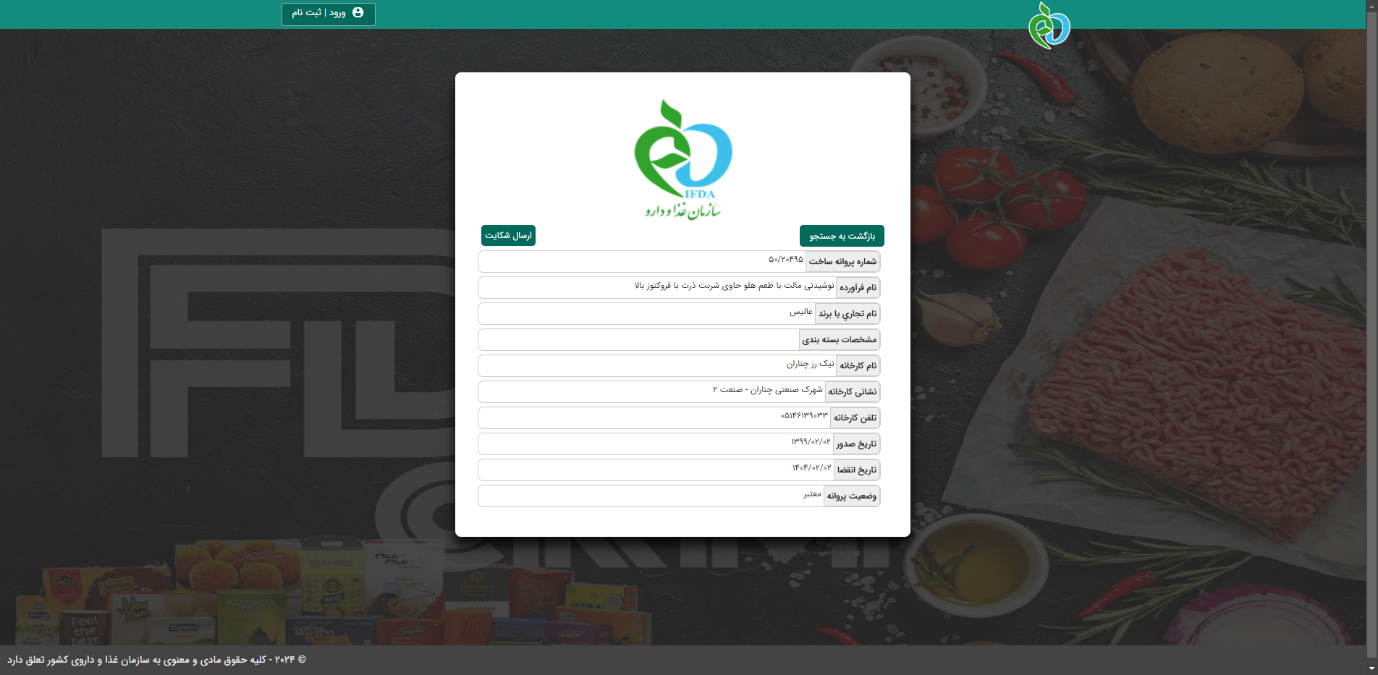


تصویر 26- خروجی برنامه بعد از تجزیه و تحلیل انرژی هر سهم محصول

## ۲-3- اطلاعات فنی محصول

با استفاده از سلنیوم، لینک سایت استعلام سیب سلامت همراه با پروانه بهداشتی محصول به عنوان کوئری[[108]](#footnote-108) به این سایت فرستاده شد و پس از دریافت اطلاعات فنی محصول، تصویری از این صفحه به عنوان خروجی در کنار سایر خروجی‌ها (تصویر محصول به همراه کلمات خوانده شده و جدول سوزاندن کالری دریافتی برحسب فعالیت بدنی) با همان نام ذخیره می‌شود.

تصویر زیر، نمای کلی وب سایت پس از استعلام محصول می‌باشد.



تصویر 27- اطلاعات فنی و نمای وب‌سایت پس از استعلام محصول

## 6-5- جمع بندی

در فصل پنجم، ارزیابی و تحلیل عملکرد مدل‌های مختلف مورد استفاده در پروژه به‌طور جامع مورد بررسی قرار گرفت. معیارهای کلیدی مانند دقت، صحت، و بازخوانی به عنوان شاخص‌های اصلی برای سنجش کارایی مدل‌ها در نظر گرفته شدند. این فصل نشان می‌دهد که مدل‌ها پس از چندین دوره آموزشی به عملکرد مطلوبی دست یافته‌اند و نوسانات اولیه به تدریج کاهش یافتند. نتایج به‌دست‌آمده از این ارزیابی‌ها، نقاط قوت و ضعف مدل‌ها را روشن کرده و زمینه‌ساز بهبود و بهینه‌سازی‌های آینده شده است. همچنین، در این فصل به تحلیل دقیق انرژی محصولات و اطلاعات فنی مربوط به آن‌ها نیز پرداخته شد.

# فصل ششم جمع بندی و پیشنهاد‌هایی برای بهبود پژوهش

## 1-6- جمع بندی کلی

این پروژه با هدف توسعه یک سیستم جامع برای تحلیل برچسب ارزش غذایی محصولات و ارزیابی آن‌ها از جنبه‌های مختلف طراحی شده است. مراحل اصلی پروژه شامل پردازش تصویر، تشخیص برچسب‌ها، خوانش کلمات و ارقام، و در نهایت تحلیل اطلاعات مربوط به انرژی محصولات غذایی بوده است.

ابتدا، از مدل‌های پیشرفته‌ی یادگیری عمیق مانند یولو برای تشخیص برچسب‌های غذایی و مدل کرفت برای خواندن کلمات و ارقام استفاده شده است. این مدل‌ها پس از چندین دوره آموزشی، عملکردی بالا در دقت، صحت و بازخوانی از خود نشان داده‌اند. ارزیابی‌های انجام‌شده بر روی این مدل‌ها نقاط قوت و ضعف آن‌ها را مشخص کرده و منجر به بهینه‌سازی‌های مؤثر در روند پروژه شده است.

در ادامه، اطلاعات مربوط به انرژی محصولات از طریق تحلیل متنی استخراج و با استفاده از فرمول‌های محاسباتی، زمان مورد نیاز برای سوزاندن کالری دریافتی از هر محصول با فعالیت‌های بدنی مختلف محاسبه و در قالب یک جدول ارائه شده است. همچنین، با استفاده از ابزارهایی نظیر سلنیوم، اطلاعات فنی محصولات از وب‌سایت‌های مرتبط جمع‌آوری و به عنوان بخشی از گزارش نهایی ارائه شده است.

به‌طور کلی، این پروژه نشان‌دهنده یکپارچگی در استفاده از تکنیک‌های یادگیری ماشین، پردازش تصویر، و استخراج اطلاعات وب بوده و قابلیت‌های خود را در تحلیل و ارائه اطلاعات کاربردی در حوزه سلامت و تغذیه به‌خوبی به نمایش گذاشته است.

## 2-6- بهبود‌های قابل اعمال

با توجه به گستردگی زیاد پروژه، نقاط و ظرفیت بالایی برای بهبود وجود دارد و اینکه که پروژه به طور کمال و تماماً بهینه پیاده سازی نشده و جای کار زیادی باقیست. در ادامه، لیستی از بهبود‌های ممکن مشاهده می‌شود.

* **مجموعه داده:** با اینکه تا کنون مجموعه داده عظیمی جمع آوری شده که حدودا 1000 برچسب ارزش غذایی محصول دارد اما در زمینه یادگیری ماشین و یادگیری عمیق، چیزی به نام داده کافی وجود ندارد و هرچه تعداد داده ها بیشتر باشد، دقت مدل بالاتر خواهد رفت. همچنین در این پروژه تمرکز بر روی برچسب ارزش غذایی استاندارد بود و برچسب‌های غیر استاندارد به درستی خوانده نمی‌شوند. به همین دلیل، اظافه کردن این امکان به این پروژه، باعث کامل‌تر شدن مجموعه داده می‌شود.
* **بهبود مدل یولو:** در این پروژه از یولو ورژن 8 استفاده شد. یکی از این دلایل، بهتر بودن پشتیبانی و مستندات[[109]](#footnote-109) خوب این مدل بود. مدل‌های بهتر چه در این زمان و چه در آینده معرفی خواهند شد که با استفاده از آنها، می‌توان دقت بخش اول این پروژه را از مقدار فعلی بهبود بخشید.
* **پیاده سازی تکنیک‌های داده افزایی[[110]](#footnote-110):** به‌کارگیری تکنیک‌های داده‌افزایی می‌تواند باعث افزایش تنوع داده‌های ورودی و بهبود عملکرد مدل‌ها شود. مثلاً تغییراتی در رنگ، اندازه، چرخش و غیره بر روی تصاویر برچسب‌ها اعمال شود تا مدل‌ها برای داده‌های واقعی‌تر آموزش ببینند.
* **خواندن برچسب‌های کج و زاویه دار:** در این پروژه، تصویری که کاربر باید وارد کند، تصویری بدون زاویه و چرخش است. می‌توان با استفاده از الگوریتم‌های مختلف، این مشکل را نیز برطرف و زاویه و چرخش تصاویر را مدیریت کرد.
* **بهینه‌سازی پردازش متن و استخراج اطلاعات:** می‌توان از الگوریتم‌های پیشرفته‌تر پردازش زبان طبیعی[[111]](#footnote-111) برای تحلیل دقیق‌تر متن‌های موجود بر روی برچسب‌ها استفاده کرد. این امر می‌تواند به بهبود دقت در تشخیص و تفسیر اعداد و کلمات کمک کند و خروجی معنی‌دارتری به کاربر نشان دهد.
* **استفاده از روش‌های یادگیری تجمعی[[112]](#footnote-112):** ترکیب چند مدل مختلف و استفاده از روش‌های یادگیری تجمعی می‌تواند به بهبود دقت نهایی پروژه منجر شود. مثلاً ترکیب خروجی چندین مدل مختلف برای تشخیص برچسب‌ها و کلمات می‌تواند نتیجه نهایی دقیق‌تری را ارائه دهد.
* **افزایش سرعت و کارایی پردازش:** بهینه‌سازی کد و استفاده از سخت‌افزارهای قوی‌تر مانند ‌GPU‌ها یا TPU‌ها می‌تواند به افزایش سرعت پردازش و کاهش زمان اجرای کل سیستم کمک کند. این به‌ویژه در مواقعی که نیاز به پردازش تعداد زیادی تصویر یا متن داریم، بسیار مفید است.
* **بهبود رابط کاربری و تجربه کاربر**: بیشتر وقت این پروژه، صرف برچسب زدن مجموعه داده و آموزش مدل‌ها شد در نتیجه شاهد رابط کاربری حرفه‌ای نیستیم. طراحی یک رابط کاربری بهتر و ساده‌تر برای کاربران نهایی می‌تواند استفاده از سیستم را راحت‌تر و جذاب‌تر کند. به‌علاوه، ارائه نتایج به‌صورت بصری و تعاملی می‌تواند تجربه کاربر را بهبود بخشد.
* **راه‌‌اندازی برنامه برای موبایل:** با توجه به اینکه تصاویر از طریق دوربین به برنامه داده می‌شوند و این دوربین عموماً دوربین تلفن همراه است، پس اگر بتوان نرم افزاری برای تلفن همراه نوشت که کاربران بتوانند به صورت مستقیم و در نرم افزار تصویر برداری کرده و به صورت آنی، نتایج را مشاهده کنند، تجربه کاربری بسیار بهتری به مصرف کننده انتقال داده می‌شود.
* **تحلیل کامل و جامع تر**: در این پروژه فقط مشاهده شد که با مصرف هرسهم از محصول، چه مدت زمان از هر فعالیت بدنی نیاز است تا آن مقدار انرژی دفع شود. اما این انتهای کار نیست. در این پروژه می‌توان با مقدار اندازه گیری و تحلیل 4 ماده غذایی یعنی قند، نمک، چربی و اسیدهای چرب ترانس، تحلیل جامع و کامل‌تری بر اطلاعات داشت. همچنین با مقایسه انرژی هرسهم و مقدار هرسهم از محصول برحسب گرم، که هربار خوانده می‌شود، می‌توان از سالم یا ناسالم بودن آن محصول، اطمینان حاصل کرد.
* **استفاده از رنگ‌ها**: رنگ‌ها در این پروژه، نقش مزاحمی در خواندن اطلاعات داشتند و این پروسه را سخت‌تر کردند. اما می‌توان به آن‌ها نیز به عنوان داده نگاه کرد. با استخراج رنگ‌ها از برچسب، می‌توان اطلاعت مفیدی از برچسب ارزش مواد غذایی استخراج کرد و تحلیلی بهتر داشت.

# منابع

[1] General Department of Supervision and Evaluation of Food Products and F. a. D. O. cosmetics, "Instruction and guide for inserting nutritional color indicator." [Online]. Available: <https://fdo.skums.ac.ir/Dorsapax/Data/Sub_5/File/neshangar%20rangi.pdf>

[2] "Roboflow." <https://roboflow.com/> (accessed.

[3] "Label studio." <https://labelstud.io/> (accessed.

[4] "Saitak number convertor." <https://www.saitak.com/number> (accessed.

[5] "Kaggle." <https://www.kaggle.com/> (accessed.

[6] "EasyOCR repository." <https://github.com/JaidedAI/EasyOCR> (accessed.

[7] "Craft repository." <https://github.com/clovaai/CRAFT-pytorch> (accessed.

[8] "Deep text recognition benchmark repository." <https://github.com/clovaai/deep-text-recognition-benchmark> (accessed.

[9] "How Do You Calculate Calories Burned During Exercise?" <https://www.medicinenet.com/how_to_calculate_calories_burned_during_exercise/article.htm> (accessed.

[10] "Metabolic Equivalent." <https://www.whyiexercise.com/metabolic-equivalent.html> (accessed.

[11] "Scan Nutrition Labels to Track Foods Faster." <https://macrofactorapp.com/label-scanner-announcement/> (accessed.

[12] J. REIBRING, "Photo OCR for Nutrition Labels," 2017. [Online]. Available: <https://odr.chalmers.se/bitstream/20.500.12380/256649/1/256649.pdf>

[13] R. S. Nate Matsunaga, "Image processing for the extraction of nutritional

information from food labels," Santa Clara University, 2015. [Online]. Available: <https://scholarcommons.scu.edu/cgi/viewcontent.cgi?article=1041&context=cseng_senior>

[14] T. Zaman, "Vision Based Extraction of Nutrition Information from Skewed Nutrition Labels," 2016. [Online]. Available: <https://digitalcommons.usu.edu/etd/4893>

[15] "Yolov8." <https://github.com/ultralytics/ultralytics> (accessed.

1. Image processing [↑](#footnote-ref-1)
2. Computer vision [↑](#footnote-ref-2)
3. Labeling [↑](#footnote-ref-3)
4. Detector [↑](#footnote-ref-4)
5. Models [↑](#footnote-ref-5)
6. Image annotation [↑](#footnote-ref-6)
7. Bounding boxes [↑](#footnote-ref-7)
8. Segmentation masks [↑](#footnote-ref-8)
9. key points [↑](#footnote-ref-9)
10. XML [↑](#footnote-ref-10)
11. JSON [↑](#footnote-ref-11)
12. Platform [↑](#footnote-ref-12)
13. Roboflow [↑](#footnote-ref-13)
14. Label studio [↑](#footnote-ref-14)
15. Open source [↑](#footnote-ref-15)
16. API [↑](#footnote-ref-16)
17. Saitak [↑](#footnote-ref-17)
18. OpenCV [↑](#footnote-ref-18)
19. Intel [↑](#footnote-ref-19)
20. Linux [↑](#footnote-ref-20)
21. MacOS [↑](#footnote-ref-21)
22. PIL (Python Imaging Library) [↑](#footnote-ref-22)
23. GPJ [↑](#footnote-ref-23)
24. PNG [↑](#footnote-ref-24)
25. GIF [↑](#footnote-ref-25)
26. BMP [↑](#footnote-ref-26)
27. Pillow [↑](#footnote-ref-27)
28. NumPy [↑](#footnote-ref-28)
29. Ndarray [↑](#footnote-ref-29)
30. Data mining [↑](#footnote-ref-30)
31. RE [↑](#footnote-ref-31)
32. Regular Expressions [↑](#footnote-ref-32)
33. Format reshaper [↑](#footnote-ref-33)
34. Arabic reshaper [↑](#footnote-ref-34)
35. Bidi (Bidirectional) [↑](#footnote-ref-35)
36. RTL [↑](#footnote-ref-36)
37. Ultralytics [↑](#footnote-ref-37)
38. YOLO (You Only Look Once) [↑](#footnote-ref-38)
39. PyTorch [↑](#footnote-ref-39)
40. Facebook [↑](#footnote-ref-40)
41. Tensor [↑](#footnote-ref-41)
42. Graph [↑](#footnote-ref-42)
43. Numpy [↑](#footnote-ref-43)
44. Scikit-learn [↑](#footnote-ref-44)
45. CUDA (Compute Unified Device Architecture) [↑](#footnote-ref-45)
46. Nvidia [↑](#footnote-ref-46)
47. GPU [↑](#footnote-ref-47)
48. C [↑](#footnote-ref-48)
49. C++ [↑](#footnote-ref-49)
50. Tensorflow [↑](#footnote-ref-50)
51. Rendering [↑](#footnote-ref-51)
52. Kaggle [↑](#footnote-ref-52)
53. Notebook [↑](#footnote-ref-53)
54. Python [↑](#footnote-ref-54)
55. R [↑](#footnote-ref-55)
56. EasyOCR [↑](#footnote-ref-56)
57. Detection [↑](#footnote-ref-57)
58. Recognition [↑](#footnote-ref-58)
59. CRAFT (Character Region Awareness for Text detection) [↑](#footnote-ref-59)
60. Deep text recognition [↑](#footnote-ref-60)
61. Confusion matrix [↑](#footnote-ref-61)
62. Accuracy [↑](#footnote-ref-62)
63. Precision [↑](#footnote-ref-63)
64. Recall [↑](#footnote-ref-64)
65. Harmonic mean [↑](#footnote-ref-65)
66. Normalized Edit Distance [↑](#footnote-ref-66)
67. Matplotlib [↑](#footnote-ref-67)
68. Line chart [↑](#footnote-ref-68)
69. Pie chart [↑](#footnote-ref-69)
70. Bar chart [↑](#footnote-ref-70)
71. Scatter plot [↑](#footnote-ref-71)
72. Metabolic Equivalent [↑](#footnote-ref-72)
73. Jinja2 [↑](#footnote-ref-73)
74. HTML [↑](#footnote-ref-74)
75. XML [↑](#footnote-ref-75)
76. Syntax [↑](#footnote-ref-76)
77. Flask [↑](#footnote-ref-77)
78. Selenium [↑](#footnote-ref-78)
79. Optical character recognition [↑](#footnote-ref-79)
80. USDA nutrition label [↑](#footnote-ref-80)
81. Convoloutional neural network [↑](#footnote-ref-81)
82. Recurrent neural network [↑](#footnote-ref-82)
83. Android [↑](#footnote-ref-83)
84. Algorithm [↑](#footnote-ref-84)
85. Classifier [↑](#footnote-ref-85)
86. Cholesterol [↑](#footnote-ref-86)
87. Sodiom [↑](#footnote-ref-87)
88. Total carbohydrate [↑](#footnote-ref-88)
89. Protein [↑](#footnote-ref-89)
90. Vitamin D [↑](#footnote-ref-90)
91. Calcium [↑](#footnote-ref-91)
92. Iron [↑](#footnote-ref-92)
93. Potassium [↑](#footnote-ref-93)
94. RGB [↑](#footnote-ref-94)
95. Model [↑](#footnote-ref-95)
96. GitHub [↑](#footnote-ref-96)
97. COCO dataset [↑](#footnote-ref-97)
98. Arguments [↑](#footnote-ref-98)
99. Epoc [↑](#footnote-ref-99)
100. Batch size [↑](#footnote-ref-100)
101. Learning rate [↑](#footnote-ref-101)
102. .txt [↑](#footnote-ref-102)
103. Backbone [↑](#footnote-ref-103)
104. VGG [↑](#footnote-ref-104)
105. Pretrained (checkpoint) [↑](#footnote-ref-105)
106. CRAFT\_clr\_amp\_14000.pth [↑](#footnote-ref-106)
107. CSV [↑](#footnote-ref-107)
108. Query [↑](#footnote-ref-108)
109. Documentation [↑](#footnote-ref-109)
110. Augmentation [↑](#footnote-ref-110)
111. Natural language processing [↑](#footnote-ref-111)
112. Ensemble learning [↑](#footnote-ref-112)