



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

پایان نامه کارشناسی ارشد

گرایش هوش مصنوعی

تولید خودکار شرح بر تصاویر با استفاده از شبکه‌های عصبی
کانولوشنی عمیق و بازگشته

نگارش

احمد اسدی

استاد راهنما

دکتر رضا صفابخش

بهمن ماه ۱۳۹۶

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تایید و تصویب پایان نامه موسوم به فرم کمیته دفاع- موجود در پرونده آموزشی- را قرار دهید.

نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



به نام خدا

تاریخ: بهمن ماه ۱۳۹۶

تعهدنامه اصالت اثر

دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)

اینجانب احمد اسدی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی استادی دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مأخذ ذکر گردیده است. این پایان‌نامه قبل از احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مأخذ بلامنع است.

احمد اسدی

امضا

نویسنده پایان نامه، در صورت تایل میتواند برای سپاسگزاری پایان نامه خود را به شخص
یا اشخاص و یا ارگان خاصی تقدیم نماید.

پاسکزاری

نویسنده پایان نامه می تواند مراتب امتحان خود را نسبت به استاد راهنمای و استاد مشاور و یا دیگر افرادی که طی انجام پایان نامه به نحوی او را یاری و یا با او همکاری نموده اند ابراز دارد.

احمد اسدی
بهمن ماه ۱۳۹۶

چکیده

در این قسمت چکیده پایان نامه نوشته می‌شود. چکیده باید جامع و بیان‌کننده خلاصه‌ای از اقدامات انجام شده باشد. در چکیده باید از ارجاع به مرجع و ذکر روابط ریاضی، بیان تاریخچه و تعریف مسئله خودداری شود.

واژه‌های کلیدی:

کلیدواژه اول، ...، کلیدواژه پنجم (نوشتن سه تا پنج واژه کلیدی ضروری است)

فهرست مطالب

عنوان

صفحه

۱	۱ مقدمه
۲	۱-۱ بیان مساله
۳	۲-۱ نگاهی بر سیر پژوهش‌های پیشین
۸	۳-۱ نگاهی بر ایده پژوهش
۹	۴-۱ خلاصه فصول بعدی
۱۰	۲ مروری بر مطالعات گذشته
۱۱	۱-۲ مقدمه
۱۱	۲-۲ پژوهش‌های انجام شده در زمینه درک صحنه توسط مغز انسان
۱۴	۳-۲ روش‌های مبتنی بر مدل‌های گرافی-احتمالی
۱۵	۱-۳-۲ استفاده از مدل میدان تصادفی مارکف
۱۸	۲-۳-۲ استفاده از مدل میدان تصادفی شرطی
۲۰	۳-۳-۲ استفاده از سایر مدل‌های گرافی احتمالی
۲۵	۴-۲ روش‌های مبتنی بر شبکه‌های عصبی کانولوشنی عمیق
۲۶	۱-۴-۲ اختصاص معنا به قطعه‌های مختلف تصویر
۲۸	۲-۴-۲ ناحیه‌بندی عمیق تصاویر به منظور نگاشت دوطرفه جملات و تصاویر
۳۲	۳-۴-۲ مدل دوطرفه نگاشت تصاویر و جملات مبتنی بر یادگیری عمیق
۳۳	۴-۴-۲ مدل زبانی مبتنی بر شبکه عصبی بازگشتی
۳۴	۵-۴-۲ مدل دوطرفه نگاشت تصاویر و جملات با استفاده از شبکه عصبی بازگشتی
۳۷	۵-۲ تولید شرح بر تصاویر با استفاده از روش‌های مبتنی بر توجه بصری
۳۷	۱-۵-۲ روش‌های مبتنی بر توجه بصری در حوزه ترجمه ماشینی
۴۲	۲-۵-۲ روش‌های مبتنی بر توجه بصری در حوزه تولید شرح متناظر تصویر
۴۷	۳-۵-۲ فعالیت‌های مشابه دیگر
۴۸	۶-۲ جمع‌بندی
۵۴	۳ روش ارائه شده در پژوهش
۵۵	۱-۳ مقدمه
۵۵	۲-۳ رمزگذار
۵۷	۳-۳ رمزگشا
۵۷	۱-۳-۳ معماری رمزگشا
۶۰	۲-۳-۳ جاسازی کلمات
۶۲	۴-۳ نحوه آموزش و تست شبکه
۶۲	۵-۳ جمع‌بندی

۶۵	۴ پیاده‌سازی، آزمون و ارزیابی
۶۶	۱-۴ مقدمه
۶۶	۲-۴ پیاده‌سازی
۶۶	۱-۲-۴ چارچوب کاری تنسورفلو
۶۷	۳-۴ معیار BLEU [۲۹]
۶۷	۴-۴ معیار METEOR [۱۹]
۶۸	۵-۴ معیار ROUGE-L [۲۱]
۶۸	۶-۴ معیار CIDEr [۳۵]
۶۹	۷-۴ آزمایشات
۶۹	۱-۷-۴ معرفی مجموعه داده
۶۹	۲-۷-۴ نمونه‌هایی از خروجی‌ها
۷۰	۳-۷-۴ مقایسه با روش‌های مشابه
۷۱	۸-۴ بحث درباره نتایج و عملکرد شبکه
۷۱	۹-۴ جمع‌بندی
۷۲	۵ جمع‌بندی، نتیجه‌گیری و پیشنهادات
۷۶	منابع و مراجع
۸۰	پیوست
۹۷	واژه‌نامه‌ی فارسی به انگلیسی
۹۹	واژه‌نامه‌ی انگلیسی به فارسی

فهرست اشکال

صفحه

شكل

۱-۱	نمونه‌ای از تصاویر ورودی و خروجی‌های مورد انتظار	۳
۲-۱	طرحواره‌ای از بستر کاری رمزگذار-رمزگشا	۸
۱-۲	نمونه توصیف‌های افراد برای تصاویر	۱۲
۲-۲	ساختار مطلوب اطلاعات استخراج شده از تصاویر [۶]	۱۳
۳-۲	تصاویر دنیای واقعی مورد استفاده در آزمایشات [۶]	۱۴
۴-۲	نمودار مقایسه‌ای عملکرد مغز در درک صحنه	۱۵
۵-۲	نمونه‌ای از نتایج به دست آمده از آزمایشات [۶]	۱۶
۶-۲	نگاشت تصویر به فضای معنایی	۱۷
۷-۲	مدل میدان تصادفی مارکف در درک صحنه	۱۸
۸-۲	مدل سلسله‌مراتبی میدان تصادفی شرطی در درک صحنه	۱۹
۹-۲	مدل گرافی احتمالی مورد استفاده در پژوهش [۲۰]	۲۱
۱۰-۲	نمونه تصاویر موجود در مجموعه‌داده مورد استفاده [۲۰]	۲۵
۱۱-۲	ماتریس درهم‌ریختگی مدل کامل ارائه شده در [۲۰]	۲۶
۱۲-۲	نتیجه مقایسه مدل‌های مختلف در [۲۰]	۲۶
۱۳-۲	نتایج نهایی به دست آمده از مدل بر روی تصاویر. [۲۰]	۲۷
۱۴-۲	طرحواره عملکرد روش RCNN	۲۹
۱۵-۲	نتایج عملکرد اهداف تعریف شده در روش RCNN برای همترازسازی تصاویر و جملات	۳۱
۱۶-۲	نتایج نهایی روش RCNN	۳۲
۱۷-۲	نتایج حاصل از جستجوی جملات در روش RCNN	۳۳
۱۸-۲	ساختار کلی شبکه ارائه شده برای نگاشت دوطرفه تصاویر و جملات در پژوهش [۲]	۳۵
۱۹-۲	نمونه‌ای از جملات تولید شده برای تصاویر توسط مدل پیشنهاد شده در [۲]	۳۶
۲۰-۲	ساختار کلی چارچوب کاری رمزگذار-رمزگشا	۳۸
۲۱-۲	ساختار رمزگشا مورد استفاده در چارچوب کاری [۱]	۳۹
۲۲-۲	ساختار کلی یک شبکه عصبی بازگشتی دوطرفه	۴۱
۲۳-۲	یک واحد از شبکه حافظه کوتاه‌مدت بلند مورد استفاده در رمزگشا پژوهش [۳۶]	۴۳
۲۴-۲	نحوه عملکرد الگوریتم در تغییر توجه بصری و کلمه تولید شده در هر نقطه.	۴۷
۲۵-۲	چند نمونه از تصاویر که در آن‌ها توجه بصری روی یک جسم منجر به تولید کلمه دقیق متناظر شده است [۳۶].	۴۷
۲۶-۲	نمونه‌هایی از تولید کلمات نامناسب مطابق با نقاط توجه استفاده شده در مدل [۳۶]	۴۸
۲۷-۲	فرایند تولید شرح متناظر تصویر با استفاده از توجه بصری سخت [۳۶]	۴۹
۲۸-۲	فرایند تولید شرح متناظر تصویر با استفاده از توجه بصری نرم [۳۶]	۵۰
۲۹-۲	ساختار پشت‌های ارائه شده در [۲۴]	۵۱
۳۰-۲	ساختار چارچوب کاری ارائه شده در [۳] در حوزه تولید شرح متناظر تصویر	۵۲
۱-۳	شبکه عصبی کوچک جایگزین فیلتر ۵ * ۵	۵۵

۵۶	۲-۳ ساختار کلی رمزگذار مورد استفاده در پژوهش
۵۸	۳-۳ نمایی از یک سلول شبکه LSTM
۵۹	۴-۳ طرحواره‌ای از ساختار LSTM پشته‌ای
۶۰	۵-۳ طرحواره‌ای از مدل Skip-Gram
۶۲	۶-۳ معماری ارائه شده در پژوهش حاضر
۶۷	۱-۴ یک نمونه از گراف‌های محاسباتی در چارچوب کاری تنسورفلو

فهرست جداول

صفحه

جدول

۱-۱ خلاصه‌ای از سیر پژوهش‌های پیشین در حوزه تولید شرح متناظر تصاویر	۴
۱-۲ امتیاز BLEU کسب شده توسط مدل نگاشت دوطرفه ارائه شده در مقایسه با مدل‌های دیگر [۲].	۳۵
۲-۱ جدول نتایج بازیابی تصاویر با استفاده از جملات ورودی در مدل ارائه شده در [۲]	۳۶
۳-۱ نتایج اعمال روش [۳۶] بر روی مجموعه‌داده‌های مختلف در مقایسه با روش‌های مختلف.	۴۶
۱-۳ مقایسه عمل کرد رمزگذار مورد استفاده در پژوهش با مدل‌های دیگر [۳۳]	۵۶
۲-۱ نمونه‌هایی از خروجی‌های صحیح مدل ارائه شده در پژوهش	۷۰
۲-۲ نمونه‌هایی از خروجی‌های غلط مدل ارائه شده در پژوهش	۷۱

فهرست نمادها

نماد	مفهوم
\mathbb{R}^n	فضای اقلیدسی با بعد n
\mathbb{S}^n	کره یکه n -بعدی
M^m	خمینه m -بعدی
$\mathfrak{X}(M)$	جبر میدان‌های برداری هموار روی M
$\mathfrak{X}'(M)$	مجموعه میدان‌های برداری هموار یکه روی (M, g)
$\Omega^p(M)$	مجموعه p -فرمی‌های روی خمینه M
Q	اپراتور ریچی
\mathcal{R}	تانسور انحنای ریمان
ric	تانسور ریچی
L	مشتق لی
Φ	۲-فرم اساسی خمینه تماسی
∇	التصاق لوی-چویتای
Δ	لاپلاسین ناهموار
∇^*	عملگر خودالحاق صوری القا شده از التصاق لوی-چویتای
g_s	متر ساساکی
∇	التصاق لوی-چویتای وابسته به متر ساساکی
Δ	عملگر لاپلاس-بلترامی روی p -فرمها

فصل اول

مقدمه

به دنبال پیشرفت تکنولوژی در ساخت دوربین‌های عکاسی و ورود دوربین‌های نیمه‌خودکار و خودکار به بازار، تعداد زیادی از کاربران سیستم‌های رایانه‌ای به استفاده از این تکنولوژی در ثبت تصاویر مورد علاقه خود جذب شده‌اند. دقیق و کیفیت مطلوب تصویربرداری از یک سو و سهولت استفاده از دوربین از سوی دیگر، باعث شده تعداد تصاویر ثبت شده توسط کاربران به طور روزافرون افزایش یابد؛ به‌طوری‌که امروزه اغلب کاربران، تعداد بی‌شماری از این تصاویر را در گوشی‌های تلفن همراه، تبلت‌ها و رایانه‌های شخصی خود نگهداری می‌کنند.

از جمله مشکلاتی که در اثر ایجاد این حجم وسیع از تصاویر بوجود آمده، مشکل مدیریت این تصاویر و یافتن تصاویر خاص بین مجموعه بزرگی از تصاویر موجود، است. از همین‌رو دست‌یابی به سامانه‌ای که بتواند به طور خودکار تمامی این تصاویر را مدیریت نماید، یک نیاز بر جسته به شمار می‌رود. ارائه سامانه‌ای که قادر به درک تصاویر و توصیف آن‌ها در قالب جملات زبان طبیعی باشد، بستر مناسبی برای مدیریت تصاویر به طور هوشمند را فراهم می‌نماید.

در این فصل، ابتدا موضوع پژوهش را به طور کامل بیان کرده و اهمیت ارائه راهکار مناسبی در این خصوص را ذکر می‌نماییم. سپس نگاهی اجمالی و گذرا بر سیر پژوهش‌های مرتبط با این حوزه اندخته و رویکرد اتخاذ شده برای حل این مساله را بیان می‌کنیم. در انتها خلاصه‌ای از آن‌چه در فصول بعدی این گزارش آمده، مطرح خواهیم نمود.

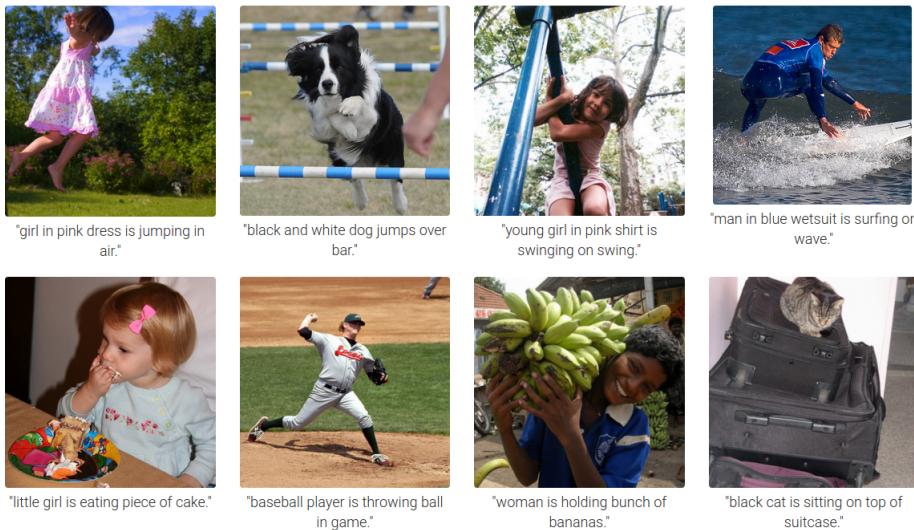
۱-۱ بیان مساله

درک تصاویر و توانایی توصیف آن‌ها، کلیدی‌ترین امکان قابل تصور برای سامانه‌های مدیریت هوشمند تصاویر به شمار می‌روند. با وجود سامانه‌ای که بتواند به طور خودکار و هوشمند، شرحی توصیفی متناظر هر تصویر دلخواه ورودی تولید نماید، می‌توان مساله مدیریت تصاویر را به مساله جستجوی میان توصیف تولید شده از تصاویر، کاهش داد.

سامانه مولد شرح خودکار بر تصاویر، سامانه‌ای است که قادر باشد هر تصویر ورودی، با هر اندازه و از هر منظره و موضوع دلخواه را در قالب جملات زبان طبیعی توصیف نماید. شرح تولید شده برای هر تصویر توسط این سامانه، باید به لحاظ زبانی درست بوده و در معنا، توصیف‌کننده تصویر باشد. این جملات علاوه بر این که باید از لحاظ دستور زبان و قواعد نحوی، صحیح باشند، باید به لحاظ معنایی نیز حامل معنای کامل و مرتبط با تصویر ورودی باشند.

برای دست‌یابی به سامانه‌ای که قادر به توصیف تصاویر از موضوعات مختلف باشد، ابتدا باید صحنه به نمایش درآمده در تصویر را به درستی درک کرد. درک صحیح از صحنه، عبارت است از بیان تصویر به نحوی که اطلاعات کلی موجود و هدف اصلی تصویر، واضح و مشخص باشد. علاوه بر این، محتوای تصاویر باید به گونه‌ای بیان گردد که جستجوی تصاویر، بر اساس توصیف تولید شده، به سهولت قابل انجام باشد.

شکل ۱-۱ نمونه‌هایی از تصاویر ورودی چنین سامانه‌ای را به همراه خروجی تولید شده توسط آن نمایش می‌دهد. همان‌طور که در این نمونه‌ها مشخص است، تصویر ورودی می‌تواند از هر منظره و موضوع دلخواهی باشد و شرح تولید شده باید شامل جملات صحیح زبانی با معنای متناظر تصویر باشند.



شکل ۱-۱: نمونه‌ای از تصاویر ورودی و خروجی‌های مورد انتظار

۲-۱ نگاهی بر سیر پژوهش‌های پیشین

شرح خودکار تصاویر، توجه پژوهش‌گران بسیار زیادی را به خود جلب کرده است و فعالیت‌های متنوع و متعددی در این راستا انجام شده‌اند. نکته قابل تأمل در این رابطه، این است که گستره وسیعی از رویکردها و روش‌ها در بین این پژوهش‌ها به چشم می‌خورد. در این بخش از گزارش، نگاهی بر سیر این پژوهش‌ها اندادخته و روند پیش‌رفت پژوهش در این حوزه را بیان می‌نماییم. همان‌طور که بیان شد، ارائه سامانه‌ای که قادر به توصیف تصاویر باشد، نیازمند ارائه روش‌هایی برای حل دو چالش اساسی زیر است.

۱. چالش درک صحنه (بازنمایی تصاویر با استفاده از بردار ویژگی)

توصیف صحنه باید دقیق باشد، به این معنی که اجسام موجود در صحنه باید به طور دقیق از هم تفکیک شده و دسته‌بندی شوند. تصویر توصیف شده باید در قالب مناسبی بازنمایی شود که بتوان به راحتی از آن برای تولید جمله استفاده نمود. استخراج یک یا چند بردار ویژگی مناسب، که اطلاعات مختلف را در خود جای داده باشند، یک بازنمایی مناسب برای تصاویر است.

۲. چالش تولید جمله (تبديل بردار ویژگی به دست آمده به جملات صحیح زبانی)

جملات تولید شده برای شرح تصویر باید به لحاظ دستور زبان، املا و معنا صحیح بوده و با تصویر مرتبط خود سازگار باشند و آن را به درستی و دقیق شرح دهند.

در سال‌های قبل از ۲۰۱۴، عموم روش‌های ارائه شده، این دو چالش را به طور مجزا از یکدیگر بررسی کرده و ایده یا روش جدیدی برای بهبود یکی از این چالش‌ها ارائه می‌دادند. اما از اواسط سال ۲۰۱۴ به بعد، عموم پژوهش‌گران با اقتباس از روش‌های موجود در حوزه ترجمه ماشینی، با استفاده از یک بستر کاری رمزگذار-رمزگشای^۱ و با بهره‌گیری از شبکه‌های عصبی کانولوشنی و بازگشتی عمیق، اقدام به ارائه مدل‌های یکپارچه برای تولید شرح متناظر تصویر می‌نمایند.

جدول ۱-۱، سیر پژوهش‌های مرتبط با حوزه تولید شرح متناظر تصویر را به طور خلاصه نمایش می‌دهد. همان‌طور

^۱Encoder-Decoder Framework

که در این جدول ملاحظه می‌شود، تا حدود سال ۲۰۰۷ میلادی، پژوهش‌گران در این حوزه، بیشتر روی چالش درک صحنه و استخراج اطلاعات مفید از تصویر تمرکز داشته‌اند. از حدود سال ۲۰۰۷ به بعد، به مرور و با پیشرفت مدل‌های استخراج ویژگی و اطلاعات از تصویر، بخش قابل توجهی از پژوهش‌ها به سمت کار بر روی تولید جملات گرایش پیدا کردند. در سال ۲۰۱۱ با رفع محدودیت یادگیری شبکه‌های عصبی بازگشتی (که در فصل بعد به طور تفصیلی به آن خواهیم پرداخت)، استفاده از روش‌های دیگر برای تولید جمله به کلی کنار گذاشته شد. از حدود سال ۲۰۱۴ میلادی، شبکه‌های عصبی کانولوشنی نیز جایگزین مدل‌های گرافی احتمالی در بخش درک صحنه شدند. از آن پس، عموم پژوهش‌های انجام شده، در بستر کاری رمزگذار-رمزگشا و با استفاده از ترکیب شبکه‌های عصبی کانولوشنی و بازگشتی، اقدام به ارائه مدل‌ها و معماری‌های جدید برای تولید خودکار شرح متناظر تصویر می‌نمایند.

جدول ۱-۱: خلاصه‌ای از سیر پژوهش‌های پیشین در حوزه تولید شرح متناظر تصاویر

روش کلی پژوهش در درک صحنه	روش کلی پژوهش در تولید جمله	باشه زمانی
دسته‌بندی تصاویر	تولید زبان طبیعی	قبل از سال ۲۰۰۰
مدل‌های گرافی احتمالی	بازیابی نزدیک‌ترین جمله	۲۰۰۰ تا ۲۰۰۷
مدل‌های گرافی احتمالی	استفاده از قالب زبانی	۲۰۱۱ تا ۲۰۰۷
مدل‌های گرافی احتمالی	شبکه عصبی بازگشتی	۲۰۱۱ تا کنون
شبکه عصبی کانولوشنی عمیق	شبکه عصبی بازگشتی	۲۰۱۴ تا کنون

در ادامه این بخش، نگاهی اجمالی بر سیر پژوهش‌ها می‌اندازیم.

• چالش درک صحنه

درک صحنه یکی از چالش‌های اساسی در زمینه بینایی ماشین است که روش‌های مختلفی برای دستیابی به آن ارائه شده است. با وجود تعدد پژوهش‌های موجود در این مورد، ارائه تعریف جامع و شامل برای این مفهوم دشوار است. عموماً این مفهوم، بسته به مورد کاربرد و هدف پژوهش، به استخراج مجموعه مشخصی از اطلاعات در مورد صحنه که برای پژوهش، کافی و مفید باشد محدود می‌شود. به همین دلیل، مجموعه اطلاعات مطلوب از تصویر که باید استخراج شود در هر پژوهش به طور خاص تعریف می‌شود.

درک صحنه در زمینه تولید خودکار شرح بر تصاویر، به طور عام شامل موارد زیر می‌شود:

۱. تشخیص اجسام موجود در صحنه و دسته‌بندی آن‌ها (مانند توپ، تلویزیون)

۲. تشخیص ارتباط مکانی بین اجسام موجود در صحنه (مانند پشت، بالا)

۳. دسته‌بندی محیط (مانند جنگل، دریا)

۴. دسته‌بندی فعالیت به تصویر کشیده شده (مانند راه‌رفتن، خوابیدن)

فعالیت‌های متعددی برای تشخیص هر یک از موارد بالا انجام شده است. به طور عام می‌توان روش‌های مورد استفاده در استخراج اطلاعات مطلوب صحنه را در زمینه تولید خودکار شرح بر تصاویر به سه دسته عمده زیر تقسیم‌بندی نمود:

۱. دسته‌بندی تصاویر

دسته‌بندی تصاویر، از جمله اولین روش‌های ارائه شده، در حوزه درک صحنه به شمار می‌رود. عموماً در این دسته از روش‌ها، تصاویر با بررسی ویژگی‌های مختلف تصویر مانند رنگ، بافت، توصیف‌کننده‌های مختلف و موارد مشابه دیگر، در دسته‌های مختلف دسته‌بندی می‌شوند. این دسته‌ها می‌توانند بر

اساس منظره موجود در تصویر (مانند خیابان، جنگل و آتاق)، فعالیت در حال انجام (مانند ورزش کردن یا پرواز کردن)، اجسام موجود در تصویر (مانند توب، تلویزیون و ماشین) یا هر موضوع دیگر که به توصیف تصاویر کمک کند، تعریف شوند. این دسته از روش‌ها در سال‌های قبل از ۲۰۰۰ برای استخراج اطلاعات از تصویر مورد استفاده قرار می‌گرفتند.

بهره‌گیری از این نوع از الگوریتم‌ها، علی‌رغم مزایای زیاد مانند سادگی و تفسیرپذیری، در دسرهای بزرگی به همراه دارد. یکی از مهم‌ترین این مشکلات، می‌توان به خاص‌منظره بودن روش‌های موجود در این دسته اشاره کرد. با توجه به این موضوع که ساخت چنین مدل‌هایی، نیازمند تعریف دقیق تمام دسته‌ها از پیش است، به کارگیری این روش‌ها در مواردی که تصاویر، قید محدود کننده‌ای ندارند، امکان‌پذیر نیست.

۲. استفاده از مدل‌های گرافی احتمالی^۱

در این دسته از روش‌ها، با استفاده از مدل‌های گرافی احتمالی می‌توان در مورد حضور یا عدم حضور اجسام مختلف در صحنه و رابطه بین اجسام موجود استنتاج نمود. همین‌طور فرایندهایی مانند قطعه‌بندی تصویر^۲ در این روش‌ها با استفاده از مدل‌های گرافی احتمالی انجام می‌شوند. یکی از ویژگی‌های این دسته از روش‌ها این است که می‌توانند به طور همزمان، هر دو چالش درک صحنه و تولید جمله را مرتفع نمایند.

با وجود قدرت بالای این مدل‌ها در حل مسائل مختلف و تفسیرپذیری به مراتب بالاتر آن‌ها نسبت به شبکه‌های عصبی، پیچیدگی تحلیل و طراحی آن‌ها بسیار زیاد است. این امر باعث می‌شود، پیش‌برد این مدل‌ها در مساله تولید شرح متناظر تصاویر، که دارای پیچیدگی‌های بسیار زیادی است، با مشکلات جدی روپرتو شود.

۳. استفاده از شبکه‌های عصبی کانولوشنی عمیق

در این دسته از روش‌ها، با استفاده از شبکه‌های عصبی کانولوشنی عمیق، پس از قطعه‌بندی تصاویر، اقدام به تفکیک اجسام مختلف در صحنه و برچسب‌گذاری هر جسم، بسته به یادگیری انجام شده، می‌شود. با بهبود روزافرون این مدل‌ها و ارائه شبکه‌های از پیش آموزش دیده و قابل استفاده، به مرور، روش‌های مبتنی بر مدل‌های گرافی احتمالی، جای خود را به شبکه‌های عصبی کانولوشنی عمیق در حل چالش درک صحنه دادند.

استفاده از این شبکه‌ها، با وجود این که تفسیرپذیری کمی دارند، علاوه بر سادگی پیاده‌سازی و طراحی، منجر به حصول نتایج بهتر و دقیق‌تر نسبت به روش‌های قبلی شده است. در حال حاضر عموم پژوهش‌گران در این حوزه، از شبکه‌های عصبی کانولوشنی به عنوان مدل استخراج ویژگی و اطلاعات از تصاویر استفاده می‌نمایند.

• چالش تولید جمله

چالش تولید جمله، متوجه ساخت جملاتی به زبان طبیعی است، به طوری که از لحاظ دستور زبان، املا و معنا صحیح باشند. از طرفی با توجه به هدف اصلی ما که تولید شرح بر تصاویر است، جملات تولید شده باید علاوه بر این که شرط صحت مذکور را ارضا می‌کنند، با تصور ورودی، صحنه توصیف شده در تصویر و رخداد به نمایش کشیده شده، هم‌خوانی داشته باشند. تضمین این هم‌خوانی از جمله معضلات دیگری است

^۱Probabilistic Graphical Models (PGMs)

^۲Image Segmentation

که باید برای آن چاره‌ای اندیشید.

به طور کلی، چهار روش مختلف زیر برای تولید جمله وجود دارد.

۱. روش تولید زبان طبیعی

مساله تولید خودکار جملات زبان طبیعی، یکی از مسائلی است که از دیرباز در هوش مصنوعی مطرح بوده و دارای کاربردهای فراوانی است. در این بخش به دنبال مدلی هستیم که بتواند با استفاده از داده‌های غیر قابل تفسیر برای انسان، جملاتی به زبان طبیعی و مناسب با شرایطی که قبل ذکر شد، تولید نماید. داده‌های اولیه که جملات با استفاده از آن‌ها تولید می‌شوند، می‌توانند شامل انواع داده‌های غیر متنی از جمله نمودارها، تصاویر، اعداد و مواردی از این دست باشند. این دسته از روش‌ها شدیداً خاص منظوره هستند. جملات تولید شده توسط این مدل‌ها، معمولاً جملاتی هستند که برای کاربردهای بسیار خاص و ویژه تولید شده‌اند. به عنوان مثال، تولید جملاتی مبني بر ورود، خروج و یا تاخیر قطارها در یک ترمینال می‌تواند توسط این مدل‌ها انجام شود. با توجه به محدود نبودن تصاویر در حوزه تولید خودکار شرح بر تصاویر، این خاص منظوره بودن، باعث کاهش چشم‌گیر عمل کرد این دسته از روش‌ها می‌شود. به همین دلیل، این روش‌ها، معمولاً در سال‌های قبل از ۲۰۰۰ در این حوزه مورد استفاده قرار می‌گرفتند.

۲. بازیابی شبیه‌ترین جمله

در این دسته از روش‌ها، ابتدا مدلی برای استخراج ویژگی از جملات ارائه می‌شود. استخراج ویژگی به نحوی انجام می‌شود که بردار ویژگی جملات قابل مقایسه با بردار ویژگی تصاویر باشد. سپس یک معیار شباخت بین این دو بردار ویژگی ارائه می‌شود که با استفاده از آن می‌توان میزان شباهت جملات و تصاویر را با هم بررسی نمود. برای تصاویر ورودی جدید، بردار ویژگی، استخراج شده و با بردار ویژگی تمام جملات موجود در مجموعه داده بررسی می‌شود. شبیه‌ترین جمله به تصویر ورودی به عنوان شرح بر تصویر، بازیابی می‌شود.

روش‌های موجود در این دسته، با وجود سادگی قابل توجهی که دارند، مشکلاتی را به وجود می‌آورند. از جمله مهم‌ترین این مشکلات این است که شبیه‌ترین جمله بازیابی شده از مجموعه داده، معمولاً به طور کامل نمی‌تواند تصویر ورودی جدید را توصیف نماید و برای توصیف کامل نیاز به ایجاد تغییراتی دارند.

۳. استفاده از کلیشه‌های آماده زبانی

استفاده از کلیشه زبانی، برای تولید جملات منطبق با تصاویر ورودی جدید، با توجه به مشکلی که روش‌های مبتنی بر بازیابی شبیه‌ترین جمله دارند، پیشنهاد شد. در این دسته از روش‌ها، ابتدا یک کلیشه زبانی برای جملات خروجی تولید می‌شود. این کلیشه معمولاً در قالب یک جمله توصیفی است. سپس با استخراج کلیدوازه‌های مناسب بسته به اطلاعات استخراج شده از تصویر، این کلیشه کامل شده و به عنوان شرح تولید شده بر تصویر، ارائه می‌شود.

استفاده از کلیشه‌های زبانی، جملات بهتری را نسبت به روش‌های مبتنی بر بازیابی شبیه‌ترین جمله، تولید می‌نماید. با این حال، ثابت بودن کلیشه زبانی باعث تولید جملات با ساختار یکسان برای تصاویر

متفاوت می‌شود.

۴. استفاده از شبکه‌های عصبی بازگشتی

شبکه‌های عصبی بازگشتی، مدل‌هایی هستند که برای پیش‌بینی و تولید دنباله‌های زمانی مورد استفاده قرار می‌گیرند. بهره‌گیری از این مدل‌ها برای تولید جملات زبان طبیعی از سال ۲۰۱۱ با حل مشکل ناپایداری آموزش شبکه، به طور عملی امکان‌پذیر شد. در حال حاضر، با توجه به این نکته که عمل کرد این شبکه‌ها به مراتب از عمل کرد روش‌های دیگر بهتر بوده، استفاده از این مدل، جایگزین تمامی روش‌های قبلی شده است.

روش‌های موجود در این دسته، برای یادگیری بهینه، نیاز به وجود مجموعه‌داده زیادی دارند. پیچیدگی پیاده‌سازی این شبکه‌ها از یک سو و کارآمدی آن‌ها در حل مسائل گوناگون از سوی دیگر سبب به وجود آمدن بسترها کاری مختلفی برای استفاده از آن‌ها شده است. این امر، بهره‌گیری از شبکه‌ها را در حل مسائل، به طور چشم‌گیری سهولت بخشیده است.

پس از سال ۲۰۱۴، تقریباً تمام پژوهش‌های انجام شده در حوزه تولید شرح بر تصاویر، از شبکه‌های عصبی برای رفع هر دو چالش استفاده می‌نمایند.

• بستر کاری رمزگذار-رمزگشا

همان‌طور که ذکر شد، از سال ۲۰۱۴ به بعد، استفاده از شبکه‌های عصبی کانولوشنی و بازگشتی عمیق، در حل مسائل هر دو چالش، جایگزین روش‌های دیگر شد. از طرفی در این سال‌ها، نگاه جدیدی به مساله تولید شرح برای تصاویر در بین پژوهش‌گران این حوزه شکل گرفت. این نگاه، مساله تولید شرح بر تصاویر را با استفاده از روش‌های موجود در حل مساله ترجمه ماشینی، تحلیل و بررسی می‌کرد که باعث ایجاد روش‌های جدیدی با اقتباس از روش‌های ترجمه ماشینی، برای تولید شرح تصاویر شد.

در ترجمه ماشینی، مساله مورد بررسی، تولید یک جمله به زبان مقصد با داشتن جمله‌ای از زبان مبدا است. برای این کار، دوتابع نگاشت تعریف می‌شود. تابع نگاشت اول، جمله ورودی از زبان مبدا را به یک فضای معنایی نگاشت می‌کند و برای هر جمله ورودی، یک بردار ویژگی در فضای معنایی تولید می‌نماید. به این تابع، رمزگذار^۴ می‌گویند. تابع نگاشت دوم، یک بردار از فضای معنایی را دریافت کرده و آن را به یک نقطه در فضای جملات زبان مقصد نگاشت می‌کند. به این تابع، رمزگشا^۵ می‌گویند. بستر کاری متشكل از یک رمزگذار و یک رمزگشا را، بستر کاری رمزگذار-رمزگشا می‌نامند.

استفاده از بستر کاری رمزگذار-رمزگشا در حوزه پژوهشی تولید شرح بر تصاویر، به سادگی انجام می‌شود. کافیست تابع رمزگذار را به نحوی تغییر دهیم که با گرفتن یک تصویر، آن را به فضای معنایی نگاشت کند. سپس با استفاده از تابع رمزگشا با تعریف مشابه در ترجمه ماشینی برای معنای هر تصویر، جمله‌ای را در فضای جملات زبان طبیعی مورد نظر، تولید می‌نماییم. به این ترتیب، گویی تصاویر را به زبان طبیعی ترجمه نموده‌ایم و ترجمه نهایی، همان شرح تولید شده برای تصویر است.

در سال ۲۰۱۵، استفاده از شبکه‌های عصبی کانولوشنی و بازگشتی عمیق در فضای پژوهشی تولید شرح متناظر تصاویر، شرایط را برای ورود ایده بستر کاری رمزگذار-رمزگشا به این حوزه، به خوبی فراهم کرده بود. یک شبکه عصبی کانولوشنی عمیق و یک شبکه عصبی بازگشتی عمیق می‌توانند به ترتیب گزینه‌های

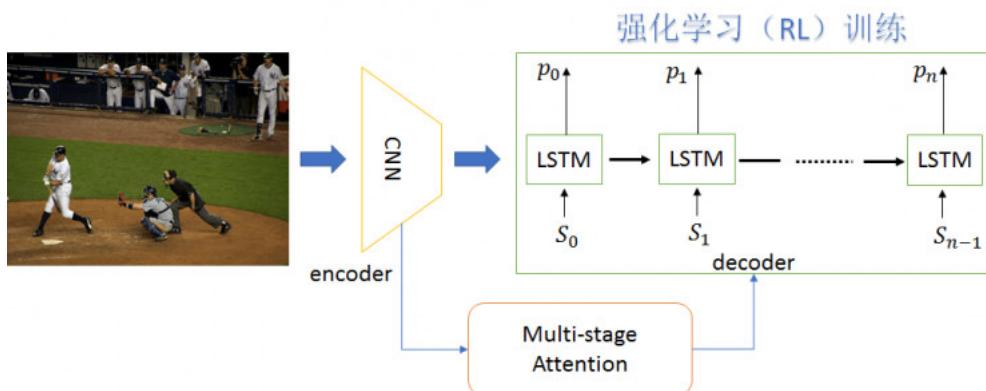
⁴Encoder

⁵Decoder

مناسبی برای توابع رمزگذار و رمزگشا به حساب بیایند. از این سال تا کنون، پژوهش‌گران این حوزه، علاوه بر بررسی فعالیت‌های اخیر انجام شده در حوزه تولید شرح متناظر تصاویر، حوزه پژوهشی ترجمه ماشینی را نیز کم و بیش رصد کرده و از پیشرفت‌های اخیر آن در حوزه تولید شرح بر تصاویر استفاده می‌نمایند.

۱-۳ نگاهی بر ایده پژوهش

در پژوهش پیش رو، از بستر کاری رمزگذار-رمزگشا برای ارائه مدل تولید شرح متناظر تصاویر استفاده شده است. شکل ۲-۱ طرح‌واره‌ای از ساختار بستر کاری رمزگذار-رمزگشا را نمایش می‌دهد.



شکل ۱-۲: طرح‌واره‌ای از بستر کاری رمزگذار-رمزگشا

همان‌طور که قبلاً ذکر شد، در این مدل، رمزگذار با دریافت تصویر، یک بردار ویژگی از تصویر استخراج می‌نماید و آن را به رمزگشا منتقل می‌نماید. رمزگشا که یک شبکه عصبی بازگشتی است، وظیفه تولید جمله متناظر تصویر را بر عهده دارد. به طور کلی این شبکه، در هر مرحله، احتمال رخداد کلمه بعدی با توجه به بردار ویژگی تصویر و کلمات تولید شده قبلی را تولید می‌نماید. رابطه (۱-۱)، ساده‌ترین ارائه ریاضیاتی از این مدل را بیان می‌نماید که در آن، Φ_{t+1} خروجی شبکه بازگشتی در مرحله $t + 1$ ، Θ بردار خروجی شبکه عصبی کانولوشنی و y_t تا y_{t+1} به ترتیب، کلمات تولید شده توسط شبکه عصبی بازگشتی در مراحل t تا $t + 1$ را نمایش می‌دهند.

$$\Phi_{t+1} = p(y_{t+1}|y_0, y_1, \dots, y_t, \Theta) \quad (1-1)$$

همان‌طور که مطابق با رابطه (۱-۱) مشخص است، خروجی شبکه، یک بردار به طول تعداد کلمات موجود در دیکشنری است که در هر مولفه آن، احتمال رخداد کلمه متناظر در دیکشنری وجود دارد. کلمه خروجی در مرحله $t + 1$ به شکل تصادفی و با توجه به این توزیع احتمال، از بین کلمات موجود در دیکشنری انتخاب می‌شود. فرایند آموزش این شبکه، به دلیل تنک بودن مقادیر خروجی مورد انتظار، تعداد پارامترهای زیاد و عدم استفاده از معنای لغات، فرایند سختی است. در این پژوهش ما از جاسازی کلمات^۶ استفاده کرده و مدل شبکه را از پیش‌بینی احتمال وقوع کلمات به انتخاب کلمه با معنای مورد انتظار تغییر دادیم. این کار باعث کاهش تعداد پارامترهای قابل آموزش شبکه، حذف برچسب‌های تنک و همین‌طور استفاده از معنای لغات و کلمات هم‌معنی در شرح تولیدی می‌شود. لازم به ذکر است، استفاده از جاسازی کلمات به جای پیش‌بینی احتمال رخداد آن‌ها، نتایج بهتری را از خود نشان داده است.

^۶Word Embedding

۱-۴ خلاصه فصول بعدی

در این فصل سعی بر ارائه مقدمات اولیه پژوهش، بیان موضوع مساله، تاکید بر ضرورت حل مساله و بیان مختصری از سیر پژوهش‌های انجام شده در این حوزه داشتیم. در فصل دوم از این گزارش، پژوهش‌های پیشین انجام شده در حوزه تولید شرح متناظر تصاویر را به طور مفصل مورد بررسی قرار می‌دهیم. در این فصل، علاوه بر ذکر مواردی از روش‌های قدیمی، بر روی بررسی پژوهش‌های انجام شده بعد از سال ۲۰۱۴ تمرکز بیشتری نموده و به دلیل ارتباط با ساختار استفاده شده در این پژوهش، ایده‌های مختلف ارائه شده در این سال‌ها را با جزئیات بیشتری بررسی می‌نماییم.

در فصل سوم، ایده اصلی پژوهش، ساختار و معماری شبکه‌های مورد استفاده و نحوه آموزش شبکه را مورد بررسی قرار خواهیم داد. فرایند آزمون و ارزیابی ساختار پیشنهادی را در فصل چهارم به طور کامل بررسی کرده و ضمن توضیح معیارهای ارزیابی مورد استفاده، نتایج عمل کرد روش پیشنهادی را با روش‌های دیگر ارائه شده، بیان می‌نماییم. لازم به ذکر است، بخش‌های اصلی کد پروژه، در پیوست اول آورده شده است.

فصل دوم

مروری بر مطالعات گذشته

۱-۲ مقدمه

تولید خودکار شرح بر تصاویر، یکی از چالشی‌ترین مسائل روز در حوزه هوش مصنوعی به شمار می‌رود. در فصل گذشته، سیر پژوهش‌های انجام شده در این حوزه را به طور اجمالی مورد بررسی قرار دادیم. در این فصل، نگاه موشکافانه‌تری بر پژوهش‌های پیشین انداخته و چالش‌های مطرح شده و روند رفع هر یک از آن‌ها را به طور دقیق تر و با جزئیات بیشتر مورد بررسی قرار می‌دهیم.

همانند بسیاری دیگر از مسائل مطرح در زمینه هوش مصنوعی، ایده‌های اولیه برای حل مساله تولید خودکار شرح بر تصاویر نیز با بررسی عمل کرد مغز انسان، ایجاد شد. در بخش اول از این فصل، یکی از پژوهش‌هایی را که با هدف بررسی روند درک صحنه و توصیف تصویر توسط مغز انسان انجام شده را مورد بررسی قرار می‌دهیم تا با ویژگی‌های اصلی مورد انتظار برای چنین سامانه‌ای آشنا شویم.

همان‌طور که گفته شد، فرایندهای تولید خودکار شرح بر تصاویر، در سال‌های قبل از ۲۰۱۴، عموماً با هدف حل یکی از چالش‌های درک صحنه یا تولید جمله انجام می‌شدند. در ادامه بحث و بررسی پیرامون مطالعات گذشته، ابتدا نمونه‌هایی از روش‌های ارائه شده برای حل چالش درک صحنه و سپس نمونه‌هایی از پژوهش‌هایی از انجام شده برای حل چالش تولید جمله را مورد بررسی قرار می‌دهیم.

استفاده از شبکه‌های عصبی و یادگیری عمیق، که از سال‌های ۲۰۱۴ به بعد توجه تعداد زیادی از پژوهش‌گران را به خود جلب کرد، در ادامه این فصل مورد بررسی قرار می‌گیرد. روش‌های ارائه شده بر بستر کاری رمزگذار-رمزگشا و روند پژوهش با استفاده از این بستر کاری، به طور دقیق‌تر و با ارائه جزئیات بیشتر مورد مطالعه قرار خواهد گرفت. از آن‌جا که پژوهش حاضر، از این بستر کاری به عنوان چارچوب اصلی استفاده کرده است، تمرکز بحث در این قسمت، بیش از سایر قسمت‌ها خواهد بود.

۲-۱ پژوهش‌های انجام شده در زمینه درک صحنه توسط مغز انسان

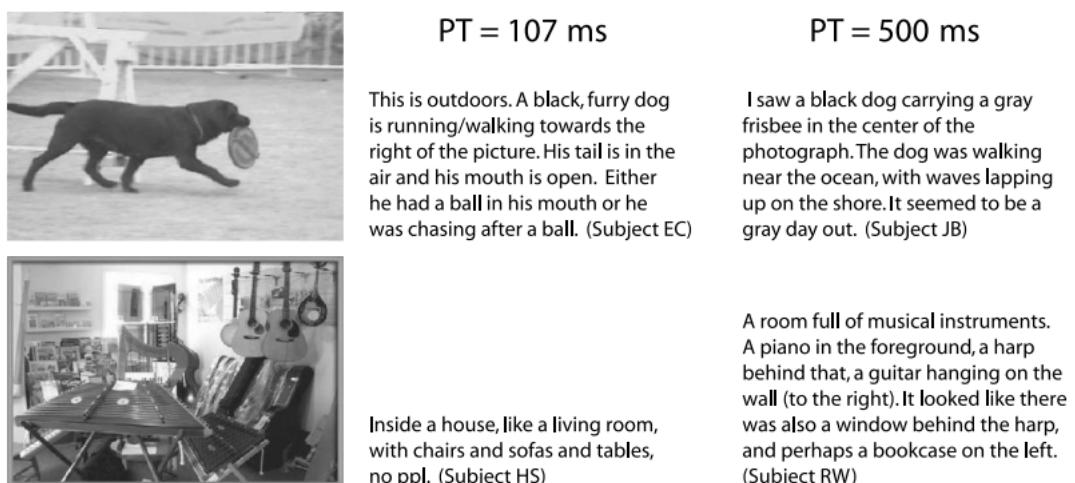
مساله درک صحنه، مانند بیشتر مسائل موجود در زمینه بینایی ماشین، الهام گرفته از نحوه رفتار انسان‌ها است. اغلب انسان‌ها با دیدن یک تصویر قادرند توصیف کامل و دقیقی از آن تصویر ارائه دهند که شامل تمام نکات لازم و ضروری نهفته در تصویر باشد. در بیشتر موارد، زمان مورد نیاز برای مغز انسان به منظور پردازش یک تصویر و توصیف آن، زمان بسیار کم و ناچیزی است. این واقعیت، این ایده را در ذهن تداعی می‌کند که بخش قابل توجهی از اطلاعات مورد نیاز از هر تصویر، در اولین لحظاتی که تصویر به مغز می‌رسد (در نگاه اول) قابل استخراج است. بنابراین سامانه‌های رایانه‌ای باید قادر باشند با الگو گرفتن از مغز انسان، در کوتاه‌ترین زمان ممکن، اطلاعات کافی و مفید نهفته در تصویر را استخراج کرده و صحنه به نمایش کشیده شده در تصویر را توصیف کنند.

این فرض که مغز انسان می‌تواند در کوتاه‌ترین زمان ممکن، بیشترین حجم اطلاعات تصویر را به درستی استخراج نماید، توسط پژوهش‌گران متعددی مورد ارزیابی قرار گرفته است. از جمله اولین پژوهش‌هایی که به بررسی این فرض پرداخته‌اند می‌توان به پژوهش‌های [۳۰] و [۳۱] اشاره کرد. در این پژوهش‌ها، با نشان دادن تصاویر به صورت دنباله‌ای^۱ به مجموعه‌ای از افراد، از آن‌ها خواسته شده تا بهترین و دقیق‌ترین توصیفی را که می‌توانند، برای تصاویری که دیده‌اند، بازگو کنند. نتایج حاصل از این دو پژوهش نشان می‌دهند که انسان می‌تواند یک تصویر معمولی را در بازه زمانی کمتر از ۲۰۰ میلی‌ثانیه، تشخیص داده و آن را توصیف کند. اگرچه این زمان برای تشخیص و توصیف یک تصویر کافیست، زمان مورد نیاز برای به خاطر سپاری تصویر بسیار بیشتر از این مقدار است. در پژوهش [۶] آزمایش دیگری انجام شده که از اهمیت بسیاری برخوردار است. در پژوهش‌های قبلی، افرادی که

^۱Image Series

تصاویر را توصیف می‌کردند، درباره موضوع کلی تصاویر اطلاعاتی داشتند. اما در این آزمایش، تصاویر مختلفی از دنیای واقعی که محدود به شرایط خاصی نبوده‌اند، بدون ارائه پیش‌فرض درباره موضوع، به افراد نمایش داده شده و از آن‌ها خواسته شده که تصویر را به بهترین شکل توصیف کنند. آزمایشات در این پژوهش، در دو مرحله انجام شده‌اند.

۱. توسط یک رایانه، تصاویر متعددی در بازه‌های زمانی متفاوت به افراد نمایش داده می‌شوند و پس از اتمام زمان نمایش هر تصویر، یک ماسک بصری، تصویر را می‌پوشاند. در این حالت از افراد خواسته شده است که بهترین توصیف ممکن از تصویر را تایپ کنند. شرایط محیطی آزمایشات مطابق با استانداردها رعایت شده است. هر تصویر به طور تصادفی بین ۲۷ الی ۵۰۰ میلی ثانیه روی نمایش گر نمایش داده شده و سپس یک ماسک روی تصویرقرار گرفته و افراد فرصت دارند تا توصیف خود را از تصویر، بنویسنند.

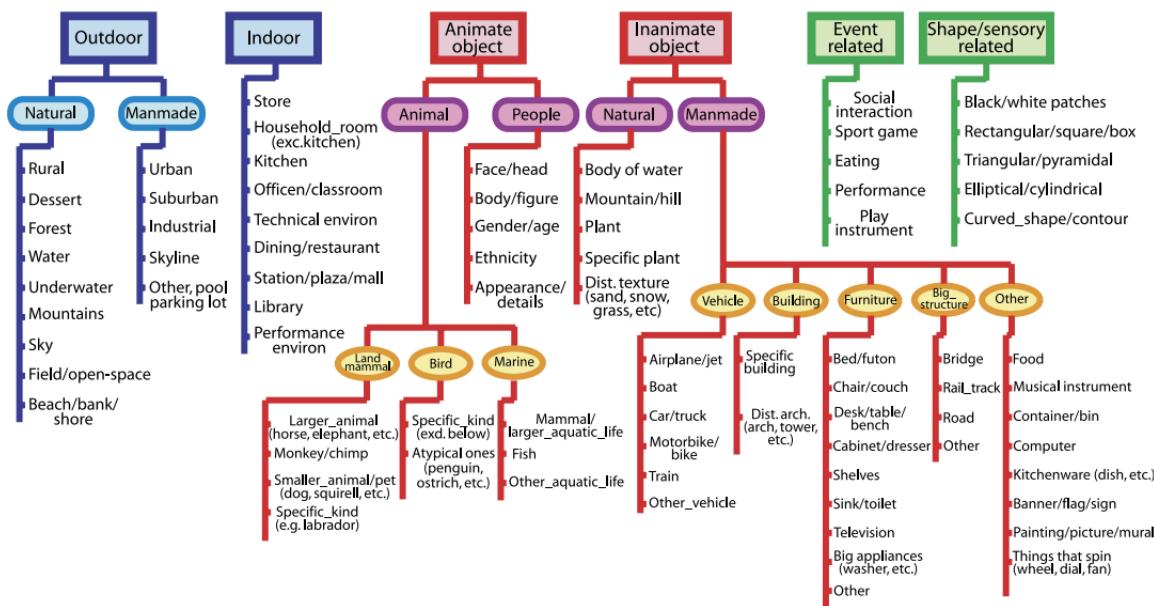


شکل ۲: نمونه توصیف‌های افراد برای تصاویر [۶]

۲. در این مرحله، آزمایش روی افراد متفاوتی انجام شده‌است. این گروه افراد موظفند پس از دیدن تصاویر، به بهترین شکل ممکن آن‌ها را دسته‌بندی کنند. برخلاف افراد شرکت‌کننده در آزمایش قبلی که می‌توانستند به هر شکلی اطلاعات استخراج شده را بنویسند، به افراد حاضر در این گروه یک فرم مشخص از دسته اطلاعات مطلوب داده شده است که افراد موظفند آن را براساس محتوای تصویری که دیده‌اند، پر کنند. شکل ۲-۲ ساختار مطلوب پاسخ افراد را در این آزمایش نمایش می‌دهد.

این ساختار با تحلیل پاسخ‌های جمع‌آوری شده از آزمایش اول استخراج شده است و شامل انواع مختلفی از اطلاعات است که افراد در آزمایش اول به آن اشاره کرده‌اند.

شکل ۳-۲ چند نمونه از تصاویر مورد استفاده در آزمایشات این پژوهش را نمایش می‌دهد. این تصاویر از اینترنت استخراج شده‌اند. برای استخراج این تصاویر از فضای اینترنت، از یک گروه افراد شامل ۱۰ نفر که با موضوع پژوهش آشنا نبوده‌اند خواسته شده تا هر یک، نام ۵ دسته صحنه مختلف را به طور تصادفی بنویسند. پس از حذف نام‌های تکراری، ۳۰ الی ۲۵ نام منحصر به فرد باقی مانده‌است. سپس تصاویر مربوط به هریک از این نام‌ها توسط موتور جستجوی گوگل استخراج شده و ۳ الی ۶ تصویر از صفحات اولیه نتایج به عنوان تصاویر نمونه انتخاب شده‌اند.



شکل ۲-۲: ساختار مطلوب اطلاعات استخراج شده از تصاویر [۶]

ارزشمندترین نکته درباره پژوهش انجام شده، یافته های آن است. این پژوهش نکاتی را در مورد توانایی مغز انسان در توصیف صحنه روشن می کند که حائز اهمیت هستند. در ادامه این نتایج را بررسی خواهیم کرد.

نتایج به دست آمده از آزمایشات

۱. حداقل زمان لازم برای مغز انسان به منظور درک صحنه، برابر با ۵۰۰ میلی ثانیه است.

۲. این مدت زمان، برای صحنه های ساده و بدون پیچیدگی، به حدود ۱۰۰ میلی ثانیه می رسد. به عنوان نمونه در شکل ۱-۲ تصویر اول که دارای پیچیدگی های کمتری نسبت به تصویر دوم است در مدت زمان ۱۰۷ میلی ثانیه، به طور کامل توصیف شده است در صورتی که تصویر دوم که به نسبت، پیچیده تر است، مدت زمان بیشتری برای توصیف نیاز داشته است.

۳. با استفاده از ساختار مندسازی پاسخ های افراد در آزمایش دوم و اطلاعات جمع آوری شده در درخت پاسخ ها (که در شکل ۲-۲ نمایش داده شده است) و میانگین گیری روی تمام تصاویر، نمودار های مقایسه ای برای مدت زمان ۱۰۷ میلی ثانیه و ۵۰۰ میلی ثانیه ایجاد شده است. شکل ۴-۲ نمودار های مقایسه ای را نمایش می دهد. در این نمودارها، میله های قرمز نشان دهنده نتایج برای زمان ۱۰۷ میلی ثانیه و میله های آبی نمایش دهنده نتایج برای حالت ۱۰۷ میلی ثانیه هستند. در دو نمودار اول (نمودار های بالا سمت راست و بالا سمت چپ) تشخیص و استخراج اطلاعات مربوط به اجسام مختلف بسته به متحرک بودن^۲ یا متحرک نبودن^۳ آن ها، در نمودار سوم (نمودار پایین سمت چپ) تشخیص و استخراج اطلاعات مربوط به صحنه موجود در تصویر و در نمودار چهارم (نمودار پایین سمت راست) تشخیص و استخراج اطلاعات مربوط به رخداد موجود در تصویر، موردن بررسی قرار گرفته اند.

همان طور که مشخص است، مدت زمان ۱۰۷ میلی ثانیه برای مغز انسان، زمان بهینه برای توصیف صحنه

² Animated

³ Inanimated



شکل ۲-۳: تصاویر دنیای واقعی مورد استفاده در آزمایشات [۶]

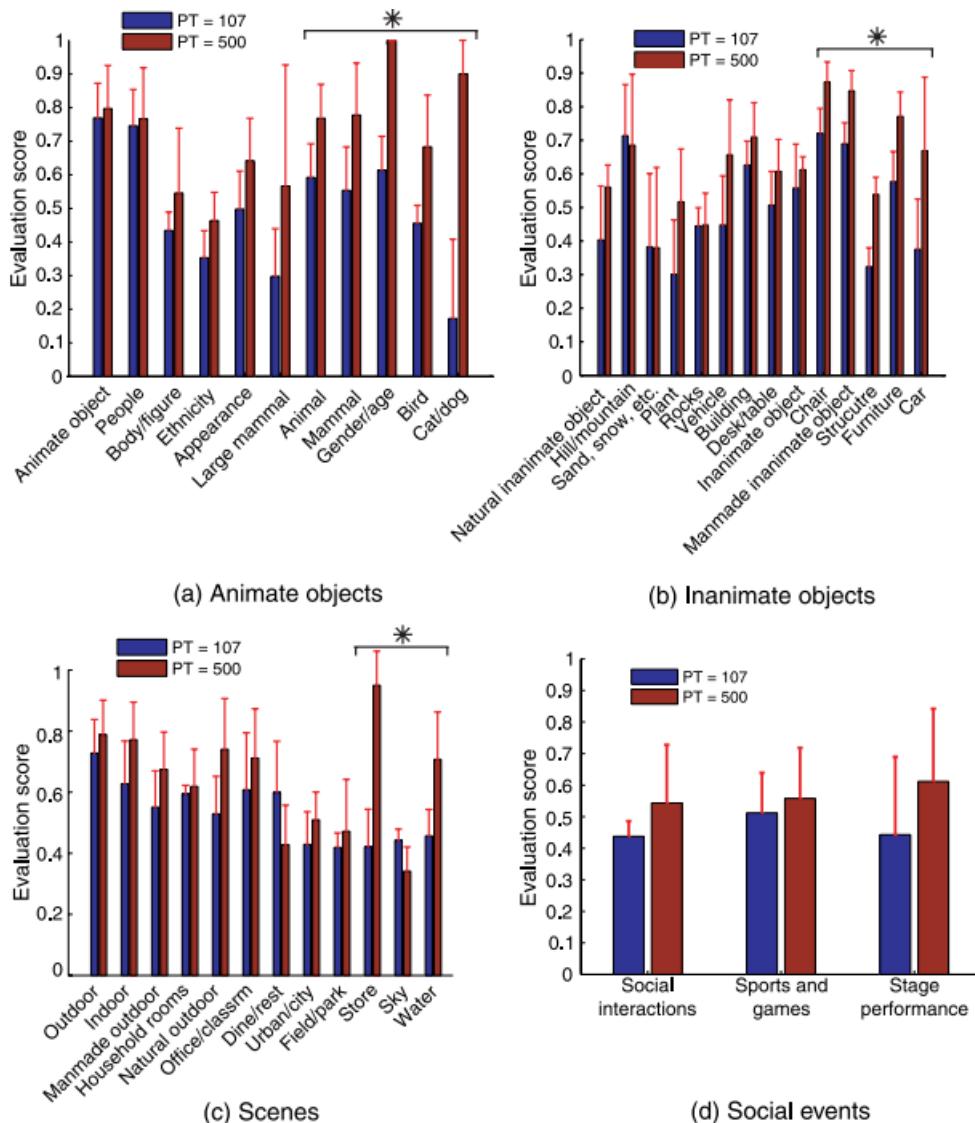
است. تفاوت‌های بین نتایج در اکثر موارد، جزئی و در مقابل تفاوت زمانی موجود، بسیار کوچک هستند. به علاوه، در تمام مواردی که نیاز به اطلاعات کلی از تصویر وجود دارد، تفاوت بین دو بازه زمانی چندان چشمگیر نیست، اما در مواردی که برای تشخیص نیاز به دانستن جزئیات بیشتر از تصویر وجود دارد (مانند سن، جنسیت و نوع حیوان)، تفاوت بین دو زمان، قابل ملاحظه است.

همین‌طور با مقایسه تفاوت عملکرد بین حالات متحرک بودن و متحرک نبودن اجسام، فواصل موجود در نمودارها قابل ملاحظه می‌شود. در حالت کلی، تفاوت بین عملکرد مغز در دو بازه، در حالتی که اجسام ساکن در تصویر وجود دارند به مراتب کمتر از حالتی است که اجسام موجود در تصویر، متحرک باشند.

شکل ۲-۵ نمونه دیگری از نتایج به دست آمده از آزمایشات را در مدت‌زمان‌های مختلف نمایش می‌دهد. در این شکل، سه تصویر از مجموعه تصاویر انتخاب شده و با مدت‌زمان‌های مختلف، به افراد نمایش داده شده است. در این شکل، نمونه‌هایی از توصیفات تولید شده توسط انسان، برای هر یک از تصاویر و در زمان‌های مختلف قابل مشاهده هستند.

۲-۳ روش‌های مبتنی بر مدل‌های گرافی-احتمالی

همان‌طور که قبلاً ذکر شد، روش‌های مبتنی بر استفاده از مدل‌های گرافی احتمالی، از جمله پرکاربردترین روش‌ها در مرحله درک صحنه در زمینه تولید خودکار شرح بر تصاویر هستند. این روش‌ها با استفاده از نظریه گراف، آمار و احتمالات اقدام به ارائه یک توزیع احتمالی برای متغیر تصادفی مورد بررسی، با توجه به داده‌های موجود در



شکل ۲-۴: نمودارهای مقایسه‌ای عملکرد مغز انسان در درک صحنه در بازه‌های زمانی ۱۰۷ و ۵۰۰ میلی‌ثانیه [۶]

مجموعه آموزشی می‌کنند. تا کنون، مدل‌های استاندارد گرافی-احتمالی مختلفی برای حل مسائل مختلف، ارائه شده‌اند که تقریباً از تمام این مدل‌ها می‌توان در بخش‌های مختلف تولید شرح خودکار بر تصاویر، استفاده نمود. یک نمونه از کاربرد مدل میدان تصادفی مارکف، یک نمونه از کاربر مدل میدان تصادفی شرطی و یک نمونه از کاربرد یک مدل مولد خاص منظوره، که توسط خانم لی و همکارانش طراحی شده است، در مساله تولید خودکار شرح بر تصاویر را مورد بررسی قرار می‌دهیم.

۲-۳-۱ استفاده از مدل میدان تصادفی مارکف^۴ [۵]

پژوهش [۵] که توسط آقای فرهادی و همکارانش در سال ۲۰۱۰ انجام شده است، با استفاده از یک مدل ساده میدان تصادفی مارکف، به حل مساله درک صحنه می‌پردازد. مدل ارائه شده در این پژوهش، علاوه بر حل چالش درک صحنه، تا حدودی می‌تواند برای تولید جملات توصیف‌گر تصویر نیز مورد استفاده قرار بگیرد.

^۴Markov Random Field (MRF)

 <p>PT 27 ms</p> <p>There was a range of dark splotches in the middle of the picture, running from most of the way on the left side, to all the way on the right side. This was surrounded primarily by a white or light gray color. (Subject: KM)</p>	 <p>PT 40 ms</p> <p>I saw a very bright object, shaped in a pyramidal shape. There was something black in the front, but I couldn't tell what it was. (Subject: JB)</p>	 <p>PT 67 ms</p> <p>Possibly outdoors, maybe a few ducks, or geese. Water in the background. (Subject: JL)</p>	<p>PT 500 ms</p> <p>It was definitely on a coast by the ocean with a large rock in the foreground and at least three birds sitting on the rock. (Subject: CC)</p>	<p>Couldn't see much; it was mostly dark w/ some square things, maybe furniture. (Subject: AM)</p>	<p>This looked like an indoor shot. Saw what looked like a large framed object (a painting?) on a white background (i.e., the wall). (Subject: RW)</p>	<p>I saw the interior of a room in a house. There was a picture to the right, that was black, and possibly a table in the center. It seemed like a formal dining room. (Subject: JB)</p>	<p>Some fancy 1800s living room with ornate single seaters and some portraits on the wall. (Subject: WC)</p>	<p>Looked like something black in the center with four straight lines coming out of it against a white background. (Subject: AM)</p>	<p>The first thing I could recognize was a dark splotch in the middle. It may have been rectangular-shaped, with a curved top...but, that's just a guess. (Subject: KM)</p>	<p>A person, I think, sitting down or crouching. Facing the left side of the picture. We see their profile mostly. They were at a table or were some object was in front of them (to their left side in the picture). (Subject: EC)</p>	<p>This looks like a father or somebody helping a little boy. The man had something in his hands, like a LCD screen or laptop. They looked like they were standing in a cubicle. (Subject: WC)</p>
---	--	--	---	--	--	--	--	--	---	---	--

شکل ۲-۵: نمونه‌ای از نتایج به دست آمده از آزمایشات [۶]

همان‌طور که قبلاً ذکر شد، اطلاعات قابل استخراج در فرایند درک صحنه، در هر پژوهش بسته به کاربرد و علاقه پژوهش‌گران، قابل تعریف است. در پژوهش [۵] اطلاعات قابل استخراج از تصویر شامل موارد زیر می‌شود:

۱. اجسام موجود

۲. فعالیت به تصویر کشیده شده

۳. صحنه موجود

بنابر تعریف فوق، در فرایند درک صحنه در این پژوهش، به ازای هر تصویر، یک سه‌تایی «جسم، فعالیت، صحنه»^۴ ایجاد می‌شود که بیان کننده اطلاعات مطلوب موجود در تصویر است. مولفه^۵ «جسم» در این سه‌تایی، مشخص کننده برچسب حاصل از دسته‌بندی اجسام موجود در صحنه، مولفه «فعالیت»، مشخص کننده اطلاعات مربوط به فعالیت در حال انجام و مولفه «صحنه»، مشخص کننده اطلاعات مربوط به محیط تصویر هستند. از آنجایی که هر سه‌تایی

مربوط به یک تصویر، محتوای تصویر را توصیف می‌نماید، به فضای این سه‌تایی‌ها، فضای معنا^۶ می‌گویند.

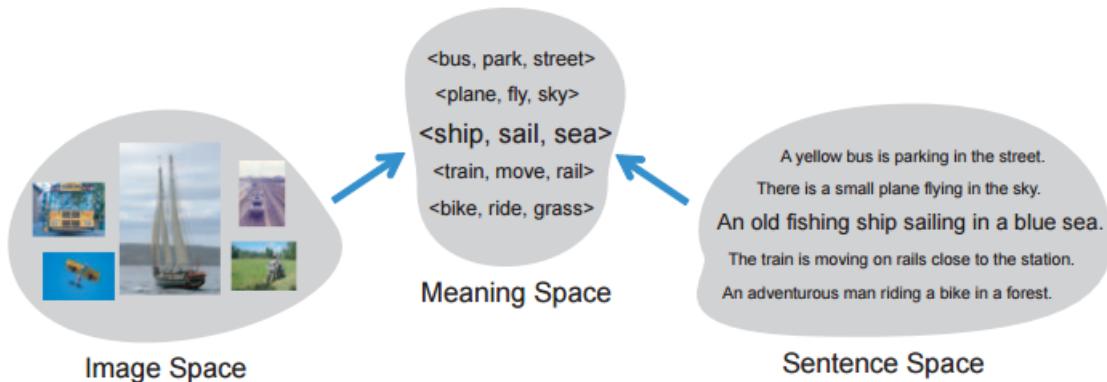
شکل ۶-۲ نمایی از نگاشت اطلاعات از فضای تصاویر و جملات به فضای معنایی، نمایش می‌دهد. همان‌طور که در شکل مشخص است، به ازای هر تصویر، یک سه‌تایی معنایی ایجاد می‌شود. همین‌طور به ازای هر جمله در فضای جملات، یک سه‌تایی ایجاد می‌شود به‌طوری که جملات و تصاویر متناظر شان، به یک سه‌تایی یکسان، نگاشت شوند. همان‌طور که مشخص است، با داشتن نگاشتهایی که خواص مذکور را داشته باشند، می‌توان با استفاده از سه‌تایی‌های فضای معنا، تصاویر را مدیریت کرد.

مدل میدان تصادفی مارکف مورد استفاده در این پژوهش، یک مدل کوچک و ساده، شامل ۳ گره است. شکل

^۴<Object, Activity, Scene>

^۵Field

^۶Meaning Space



شکل ۲-۶: نگاشت تصویر به فضای معنایی. فضای معنایی شامل اطلاعات مطلوب برای استخراج در فرایند درک صحنه است. به ازای هر تصویر، یک سه‌تایی ایجاد می‌شود [۶].

۷-۲ طرح‌واره‌ای از مدل میدان تصادفی مارکف مورد استفاده در این پژوهش را نمایش می‌دهد. همان‌طور که در شکل مشخص است، به ازای هر کدام از مولفه‌های تعریف شده در فضای معنایی، یک گره در این مدل وجود دارد. مقادیر مختلف در هر گره، برابر است با مقادیر مختلف موجود در مولفه متناظر در فضای معنا، که با توجه به داده‌های مجموعه آموزشی مشخص می‌شوند. همین‌طور به ازای هر دو گره موجود در این مدل، یک یال بیان‌کننده ارتباط بین دو میدان در فضای معنایی وجود دارد.

برای استنتاج در این مدل، لازم است ابتدا فاکتورهای مورد استفاده در مدل را شناخته و مقادیر آن‌ها را مشخص نماییم. در مدل پیشنهادی، دو نوع فاکتور تعریف شده است:

۱. فاکتورهای گره

این فاکتورها، برای مشخص کردن میزان شباهت مقادیر مختلف گره با تصویر ورودی، تعریف شده‌اند. ویژگی‌های مورد استفاده برای مقداردهی این فاکتورها، شامل موارد زیر هستند:

(آ) استفاده از آشکارکننده‌های ^۸ فلزنسوالب ^۹، به منظور محاسبه امتیاز اطمینان ^{۱۰}. برای هر دسته از اجسام موجود در مجموعه داده [۷].

پس از محاسبه امتیاز اطمینان همه دسته‌های موجود، دسته‌ای که بیشترین امتیاز را دارد می‌تواند به عنوان دسته منتخب در مولفه متناظر گره، انتخاب شود. در فرایند مقداردهی این ویژگی، قبل از انجام محاسبات، اطمینان حاصل می‌شود که از هر دسته موجود، حداقل یک تصویر در مجموعه داده وجود داشته باشد.

(ب) استفاده از پاسخ دسته‌بندی کننده دیوالا ^{۱۱}، ارائه شده در مقاله [۴]

(ج) استفاده از دسته‌بندی کننده مبتنی بر گیست ^{۱۲}

بر اساس مقادیر محاسبه شده برای ویژگی‌های بالا و با استفاده از الگوریتم ماشین بردار پشتیبان ^{۱۳}، یک دسته‌بندی برای هر گره ارائه می‌شود که بیان‌کننده دسته ویژگی‌های مربوط به مقادیر مختلف گره است. با

⁸Detector

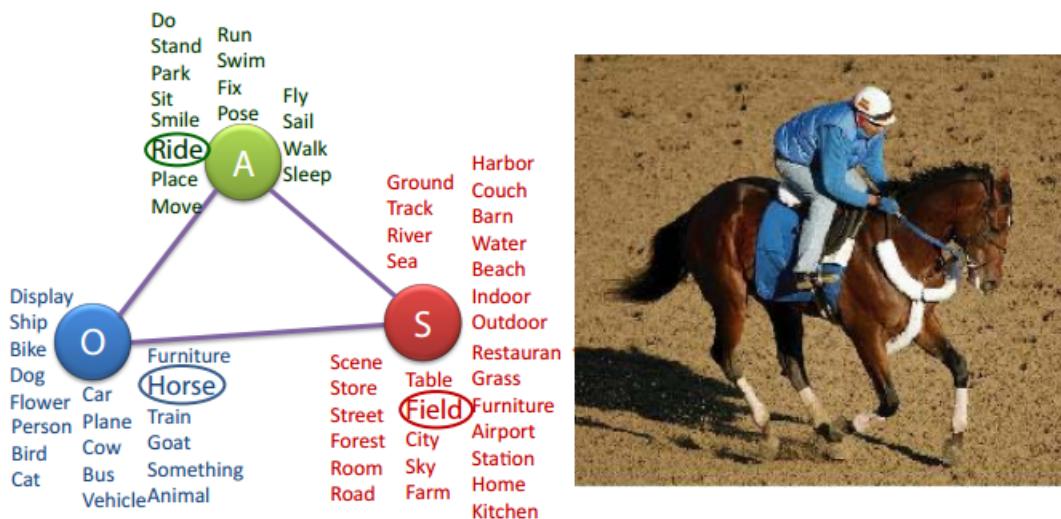
⁹Felzenszwaalb

¹⁰Confidence Score

¹¹divvala

¹²Gist-based classification response

¹³Support Vector Machine (SVM)



شکل ۲-۲: طرح‌واره مدل میدان تصادفی مارکف ارائه شده در پژوهش [۵] که شامل ۳ گره است. در این مدل، به ازای هر میدان از فضای معنا، یک گره وجود دارد و بین هر سه گره، به طور دو به دو، یک یال موجود است [۵].

استفاده از این دسته‌بندی، با ورود هر تصویر، می‌توان برای هر مقدار در هر گره، یک امتیاز شباهت محاسبه نمود. استفاده از الگوریتم یافتن نزدیک‌ترین همسایه‌های موجود برای هر تصویر ورودی، بر اساس امتیاز شباهت محاسبه شده و میانگین‌گیری روی همسایه‌های استخراج شده، معیار خوبی از تخمین مقدار هر گره، به ازای هر تصویر ورودی ایجاد می‌کند. به این ترتیب، با ورود هر تصویر می‌توان برای هر کدام از گره‌های موجود در مدل، یک مقدار محتمل مشخص نمود. سه‌تایی شامل مقادیر محتمل بدست‌آمده در هر گره، سه‌تایی متناظر تصویر ورودی در فضای معنا را مشخص می‌کند.

۲. فاکتور یال

این فاکتور، برای مشخص کردن میزان ارتباط مقادیر مختلف دو گره با یکدیگر در تصویر ورودی مورد استفاده قرار می‌گیرند.

۲-۳-۲ استفاده از مدل میدان تصادفی شرطی^{۱۴}

مدل میدان تصادفی شرطی، یکی از پرکاربردترین مدل‌های گرافی احتمالی در زمینه درک صحنه است که پژوهش‌های متعددی از آن به عنوان مدل اصلی در درک صحنه استفاده کرده‌اند. به عنوان نمونه، در پژوهش‌های [۲۲] و [۱۸] از مدل میدان تصادفی شرطی به منظور توصیف صحنه استفاده شده است.

پژوهش [۲۲] که توسط لین و همکارانش در سال ۲۰۱۳ ارائه شد، سعی در توصیف اجسام سه‌بعدی با استفاده از قطعه‌بندی تصاویر دو بعدی، هندسه سه‌بعدی و روابط بین صحنه و اجسام موجود، دارد. در این پژوهش، پس از استخراج ویژگی‌ها و اطلاعات بدست‌آمده از منابع مختلف، عمل استنتاج توسط یک مدل تصادفی شرطی انجام می‌شود که منجر به نگاشت تصویر ورودی به فضای معنایی می‌شود. همین‌طور در پژوهش [۱۸] که توسط لادیکی و همکارانش در سال ۲۰۱۰ ارائه شد، یک بستر کاری احتمالی برای استنتاج درباره نواحی مختلف تصویر، اجسام موجود و ویژگی‌های مختلف آن‌ها مانند دسته‌بندی، موقعیت مکانی و ابعاد، مبتنی بر مدل میدان تصادفی شرطی، ارائه شده است. با توجه به وسعت و تعدد فعالیت‌های انجام شده، در این بخش چزئیات یکی از روش‌های ارائه

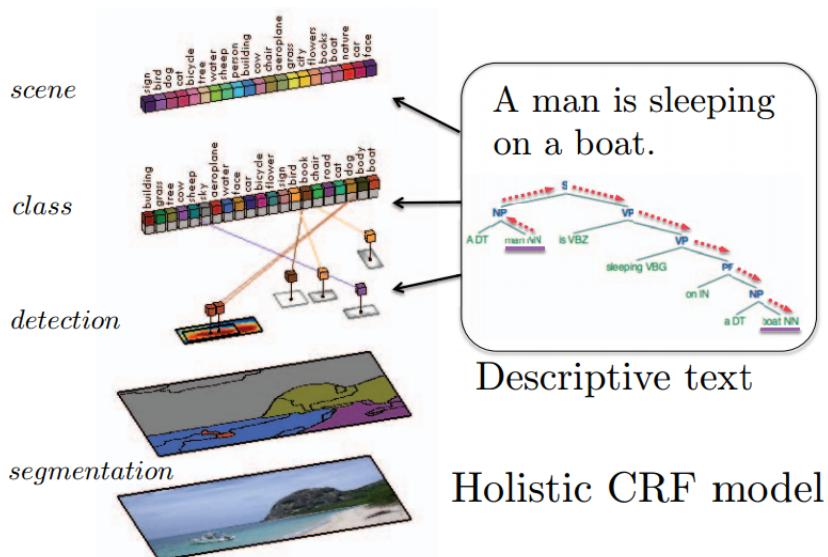
^{۱۴}Conditional Random Field (CRF)

شهر در این دسته‌بندی را مطرح نموده و از بررسی عمیق‌تر پژوهش‌های دیگر صرف‌نظر می‌نماییم.
در پژوهش [۹] که در سال ۲۰۱۳ توسط فیدلر و همکارانش ارائه شد، از مدل میدان تصادفی شرطی برای توصیف صحنه و اجسام موجود در آن استفاده شده است. میدان‌های تصادفی در این مدل، شامل متغیرهای زیر هستند:

۱. متغیرهای تصادفی بیان‌کننده برچسب دسته متناظر قطعات مختلف هر تصویر به شیوه سلسله مراتبی دارای دو سطح

۲. متغیرهای تصادفی بیان‌کننده صحت دسته تشخیص داده شده برای هر جسم

شکل ۸-۲ طرح‌واره مدل سلسله‌مراتبی ارائه شده در پژوهش [۹] را نمایش می‌دهد. همان‌طور که مشاهده می‌شود این مدل از دو سطح انتزاع، یکی برای برچسب قطعات مختلف تصویر و دیگری برای حضور یا عدم حضور هر دسته از اجسام در تصویر، تشکیل شده است.



شکل ۸-۲: طرح‌واره مدل سلسله مراتبی مبتنی بر میدان تصادفی شرطی که بر اساس اطلاعات بصری و اطلاعات جملات توصیف‌کننده شرح محتمل تصویر را تولید می‌نماید [۹].

دو دسته متغیر تصادفی مختلف، که هر یک نماینده متغیرهای تصادفی موجود در یکی از این سطوح انتزاع هستند، تعریف شده‌اند؛ متغیرهای تصادفی $\{X_i \in \{1, \dots, C\}, Y_j \in \{1, \dots, C\}\}$ بیان‌کننده دسته قطعه i از سطح پایین سلسله مراتب و متغیرهای تصادفی $\{Y_\alpha \in \{1, \dots, C\}\}$ بیان‌کننده دسته قطعه j از سطح بالای سلسله مراتب. به علاوه، دو دسته متغیر تصادفی دیگر به نام‌های b_l و z_k به ترتیب برای نمایش حضور یا عدم حضور یک تشخیص کاندید [۱۵] و حضور یا عدم حضور جسم با دسته k در تصویر، تعریف شده‌اند. با توجه به متغیرهای تعریف شده، مدل کلی میدان تصادفی شرطی را می‌توان معادل رابطه (۱-۲) تعریف کرد. در این رابطه $(a_\alpha)_{\alpha} \Psi_\alpha^{type}$ نماینده تابع پتانسیل تعریف شده روی متغیرهای مختلف است. با این تعریف، یافتن تخمین احتمال بیشینه پسین [۱۶]، منجر به یافتن پاسخ مورد نظر می‌شود. در ادامه، توابع پتانسیل مختلف که در این پژوهش تعریف شده‌اند، ارائه خواهد شد. لازم به ذکر است در تمام این موارد، برای سهولت، توابع پتانسیل به شکل لگاریتمی تعریف شده‌اند.

$$P(X, Y, b, z) = \frac{1}{Z} \prod_{type} \prod_{\alpha} \Psi_\alpha^{type}(a_\alpha) \quad (1-2)$$

^{۱۵}Candidate Detection

^{۱۶}MAP Estimation

توابع پتانسیل مختلف تعریف شده در این پژوهش عبارتند از:

۱. پتانسیل قطعه‌بندی یگانی^{۱۷}

پتانسیل قطعه‌بندی یگانی در هر قطعه و هر ابرقطعه^{۱۸} از تصویر، با استفاده از میانگین‌گیری روی امتیاز افزایش تکستون^{۱۹} که در پژوهش^{۲۰} [۱۷] ارائه شده است، انجام می‌شود.

۲. انطباق بین متغیرهای دو سطح انتزاع با یکدیگر

یک مقدار جریمه به ازای دسته‌های مخالف بین دو سطح در نظر گرفته می‌شود تا در حد امکان، دسته‌های منتخب از بین سطوح مختلف، با یکدیگر انطباق داشته باشند. پتانسیل تعریف شده در این بخش معادل رابطه^{۲۱} تعریف می‌شود.

$$\phi_{ij}(X_i, Y_j) = \begin{cases} -\gamma & X_i \neq Y_j \\ 0 & X_i = Y_j \end{cases} \quad (2-2)$$

در رابطه^{۲۲}، پارامتر γ در فرآیند یادگیری که منجر به بهینه‌سازی پارامترهای مختلف مدل می‌شود، به دست می‌آید.

۳. پتانسیل انطباق تصویر و دسته جسم

برای اندازه‌گیری میزان انطباق هر کدام از دسته‌های موجود برای اجسام با تصویر ورودی، از معیار انطباق ارائه شده در پژوهش^{۲۳} [۱۸] توسط فلزنسوالب که به روش دی‌پی‌ام^{۲۰} مشهور است، استفاده شده است. برای کاهش تعداد پارامترها و افزایش کارایی مدل استفاده شده، برای هر تصویر حداقل ۳ دسته جسم، به عنوان دسته‌های منتخب کاندید، در نظر گرفته می‌شوند.

۳-۳-۲ استفاده از سایر مدل‌های گرافی احتمالی

در بین پژوهش‌های موجود در زمینه درک صحنه با استفاده از روش‌های گرافی احتمالی، علاوه بر مدل‌های استاندارد، از مدل‌های مولد دیگر در پژوهش‌های متعددی استفاده شده است. در ادامه این بخش، به بررسی چند نمونه از این مدل‌ها خواهیم پرداخت.

۱. دسته‌بندی تصاویر بر اساس صحنه و اجسام موجود به طور توأم^{۲۰} [۲۰]

مدل استفاده شده در این پژوهش، از تصاویر در سطح صحنه و سطح اجسام استفاده کرده و با یکپارچه‌سازی و تجمیع اطلاعات موجود در این دو سطح، اقدام به دسته‌بندی تصویر می‌نماید. شکل^{۹-۲} مدل استفاده شده در این پژوهش را به منظور یکپارچه‌سازی و تجمیع اطلاعات حاصل از تحلیل صحنه و تشخیص اجسام موجود در آن، ارائه می‌دهد.

یکی از اهدافی که در این پژوهش دنبال می‌شود، برچسب‌گذاری معنایی^{۲۱} تمام پیکسل‌های موجود در تصویر است. به همین منظور، تمام تصاویر مورد استفاده، به نواحی^{۱۰} *^{۱۰} تقسیم شده و مورد استفاده قرار می‌گیرند. برای بررسی بهتر مدل، ابتدا متغیرهای تصادفی مورد استفاده را تعریف کرده و سپس به بررسی

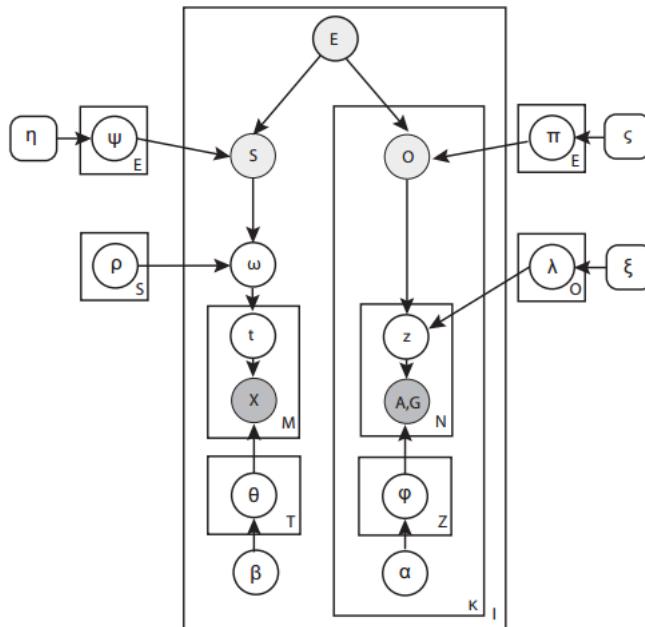
^{۱۷}Unary Segmentation Potential

^{۱۸}Supersegment

^{۱۹}Texton Boost

^{۲۰}DPM

^{۲۱}Semantic Labelling



شکل ۹-۲: مدل استفاده شده به منظور تجمعی اطلاعات صحنه و اجسام موجود در آن به منظور دسته‌بندی تصاویر [۲۰]

روند یادگیری و استنتاج مدل می‌پردازیم.

متغیر تصادفی X که حاوی اطلاعاتی مبتنی بر حضور یا عدم حضور دسته‌های مختلف صحنه است، در بخش تشخیص صحنه به کار می‌رود. اطلاعات این متغیر با استفاده از توصیف‌کننده سیفت^{۲۲} و به ازای هر ناحیه از تصویر، به دست می‌آید. برای بخش تشخیص اجسام موجود در صحنه، از دو منبع اطلاعاتی مختلف استفاده می‌شود. اطلاعات مربوط به حضور یا عدم حضور دسته‌های مختلف اجسام در متغیر تصادفی A و اطلاعات مربوط به شکل کلی آن‌ها در متغیر تصادفی G نمایش داده می‌شود.

هر گره از مدل ارائه شده، نماینده یک متغیر تصادفی است. گره‌هایی که با رنگ تیره مشخص شده‌اند، نماینده متغیرهایی هستند که در فرایند آموزش دیده می‌شوند و بقیه متغیرها، متغیرهای مخفی^{۲۳} هستند. گره‌های خاکستری روش‌تر، متغیرهایی هستند که فقط در فرایند آموزش دیده می‌شوند در حالی که متغیرهای تیره‌تر در هر دو فرایند آموزش و آزمون مشاهده می‌شوند.

متغیر تصادفی E ، نماینده یک دسته از رخداد^{۲۴} های ممکن است. توزیع احتمال اولیه این متغیر تصادفی، یک توزیع یکنواخت فرض شده است که به هر تصویر ورودی، بر اساس همین توزیع، یک مقدار خاص از این متغیر تصادفی اختصاص داده می‌شود. با دانستن دسته رخداد موجود در تصویر، یک تصویر صحنه^{۲۵} متناظر با تصویر ورودی تولید می‌شود. با فرض وجود S دسته صحنه مختلف در مجموعه‌داده، به هر تصویر، تنها یک دسته صحنه اختصاص داده می‌شود. روند اختصاص دسته صحنه به تصویر مطابق زیر است:

* ابتدا یک دسته اولیه مطابق با توزیع احتمال شرطی ($P(S|E, \psi)$ به تصویر اختصاص داده می‌شود.

^{۲۲}SIFT Descriptor

^{۲۳}Latent

^{۲۴}Event

^{۲۵}Scene Image

ψ یک پارامتر چندجمله‌ای^{۲۶} حاکم بر توزیع احتمالاتی S به شرط داشتن E است. به علاوه، ψ یک ماتریس به ابعاد $E * S$ و پارامتر η یک بردار S بعدی در نقش مقدار اولیه دیریکله^{۲۷} برای پارامتر ψ است.

* در قدم بعدی با داشتن مقدار S ، پارامترهای ω را بر اساس احتمال $P(\omega|S, \rho)$ تولید می‌کنیم. آن‌جا که ω پارامتر چندجمله‌ای گره‌های مخفی t هستند، باید مجموع همه آن‌ها برابر با یک باشد. به علاوه، ρ یک ماتریس به ابعاد $T * S$ و مقدار اولیه دیریکله برای پارامتر ω است که در آن T تعداد کل t ‌ها است.

- * برای تولید هر یک از M ناحیه تصویر (مقادیر متغیر تصادفی X) به شکل زیر عملی می‌کنیم:
 - یک مقدار t از توزیع احتمال $Mult(\omega)$ تولید می‌شود که مشخص‌کننده موضوعی^{۲۸} است که این ناحیه از تصویر مطابق با آن تولید شده است.
 - متغیر تصادفی X از توزیع احتمالی $P(X|t, \theta)$ تولید می‌شود. یک ماتریس به ابعاد $V_s * V_s$ است که در آن V_s تعداد کلمات موجود در پایگاه داده مربوط به صحنه s است. به علاوه، θ یک پارامتر چندجمله‌ای برای X است و β مقدار اولیه دیریکله برای θ .

همانند فرایندی که طی آن، تصویر صحنه به تصویر ورودی اختصاص داده می‌شود، فرایندی وجود دارد که طی آن تصویر اجسام^{۲۹} به تصویر ورودی اختصاص داده می‌شود. بر خلاف صحنه، هر تصویر می‌تواند بیش از یک جسم داشته باشد. تعداد کل اجسام موجود در یک تصویر را با K و تعداد کل دسته‌های موجود برای اجسام در مجموعه‌داده را با O نمایش می‌دهیم. فرایند زیر برای هر یک از K جسم موجود در تصویر اجرا می‌شود:

- * ابتدا یک دسته جسم با توزیع احتمالی $P(O|E, \pi)$ به تصویر اختصاص داده می‌شود که در آن، π یک ماتریس به ابعاد $O * O$ و ζ یک بردار به طول O و مقدار اولیه دیریکله پارامتر π است.
- * سپس با داشتن O می‌توان تمام نواحی A و G مرتبط با دسته جسم را تولید نمود. فرایند تولید این نواحی به شکل زیر است:
 - متغیر تصادفی مخفی z که مشخص کننده موضوع است، از توزیع احتمالی $Mult(\lambda, |O|)$ تولید می‌شود. متغیر λ یک ماتریس به ابعاد $O * Z$ است که در آن Z تعداد کل مقادیر مختلف متغیر z است. به علاوه ξ مقدار اولیه دیریکله برای پارامتر λ است.
 - نواحی مطلوب از توزیع احتمال $P(A, G|t, \phi)$ تولید می‌شوند که در آن، ϕ یک ماتریس به ابعاد $Z * V_o$ است. V_o تعداد کل کلمات موجود در مجموعه‌داده، به ازای نواحی A و G است. پارامتر α مقدار اولیه دیریکله برای پارامتر ϕ است.

با توجه به متغیرهای تصادفی توضیح داده شده در بالا، توزیع احتمالی توام کل سیستم را می‌توان مطابق با

^{۲۶}Multinomial

^{۲۷}Dirichlet prior

^{۲۸}Topic

^{۲۹}Object Image

رابطه ۳-۲ تعریف کرد.

$$\begin{aligned}
 P(E, S, O, X, A, G, t, z, \omega | \rho, \phi, \lambda, \psi, \pi\theta) &= P(E) \cdot P(S|E, \psi) \cdot P(\omega|S, \rho) \\
 &\cdot \prod_{m=1}^M P(X_m|t_m, \theta) \cdot P(t_m|\omega) \\
 &\cdot \prod_{k=1}^K P(O_k|E, \pi) \\
 &\cdot \prod_{n=1}^N P(A_n, G_n|z_n, \phi) \cdot P(z_n|\lambda, O_k)
 \end{aligned} \tag{۳-۲}$$

به علاوه، با توجه به توضیحات ارائه شده در بالا، هر کدام از عبارات موجود در رابطه ۳-۲ را می‌توان با عبارات معادل آن‌ها که در روابط ۴-۲ تا ۱۰-۲ آمده، جایگزین نمود.

$$P(S|E, \psi) = Mult(S|E, \psi) \tag{۴-۲}$$

$$P(\omega|S, \rho) = Dir(\omega|\rho_j), S = j \tag{۵-۲}$$

$$P(t_m|\omega) = Mult(t_m|\omega) \tag{۶-۲}$$

$$P(X_m|t_m, \theta) = P(X_m|\theta_j), t_m = j \tag{۷-۲}$$

$$P(O_k|E, \pi) = Mult(O_k|E, \pi) \tag{۸-۲}$$

$$P(z_n|\lambda, O_k) = Mult(z_n|\lambda, O_k) \tag{۹-۲}$$

$$P(A_n, G_n|z_n, \phi) = P(A_n, G_n|\phi_j), z_n = j \tag{۱۰-۲}$$

در ک صحنه در این پژوهش، محدود به استخراج سه دسته اطلاعات زیر از تصویر است:

(آ) رخدادی که در تصویر به نمایش گذاشته شده است.

(ب) صحنه‌ای که تصویر در آن ایجاد شده است.

(ج) اجسامی که در تصویر حضور دارند.

با توجه به این محدودیت و با در نظر گرفتن مدل ارائه شده، استفاده از تخمین بیشینه احتمال^{۳۰}، می‌تواند برای استخراج اطلاعات مطلوب مفید باشد. از همین رو، تخمین بیشینه احتمال، در سه سطح مختلف (هر سطح برای یک دسته از اطلاعات مطلوب) اعمال می‌شود. در سطح اجسام، احتمال رخداد تصویر ورودی به شرط اجسام موجود مطابق با رابطه ۱۱-۲، احتمال رخداد تصویر ورودی به شرط صحنه، مطابق با رابطه ۱۲-۲ و احتمال رخداد تصویر ورودی به شرط دسته رخداد به نمایش گذاشته شده در تصویر، مطابق با رابطه

^{۳۰} Maximum Likelihood

۱۳-۲ محاسبه می‌شوند.

$$P(I|O) = \prod_{n=1}^N \sum_j P(A_n, G_n | z_j, O) P(z_j | O) \quad (11-2)$$

$$P(I|S, \rho, \theta) = \int P(\omega | \rho, S) (\prod_{m=1}^M \sum_{t_m} P(t_m | \omega) P(X_m | t_m, \theta)) d\omega \quad (12-2)$$

$$P(I|E) \propto \sum_j P(I|O_j) P(O_j|E) P(I|S) P(S|E) \quad (13-2)$$

فرایند یادگیری این مدل، شامل یافتن بهترین مقادیر برای پارامترهای $\{\psi, \rho, \pi, \lambda, \theta, \beta\}$ است. این فرایند برای سه پارامتر $\{\psi, \rho, \theta\}$ با استفاده از روش انتقال پیام متغیر^{۳۱} و برای سه پارامتر $\{\pi, \lambda, \beta\}$ با استفاده از نمونه‌برداری گیبس^{۳۲} انجام می‌شود.

آزمایشات انجام شده در این پژوهش، بر روی یک مجموعه‌داده شامل تصاویر از ۸ دسته ورزشی مختلف که در هر دسته، بین ۱۳۷ تا ۲۵۰ تصویر مختلف وجود دارد، انجام شده‌اند. از جمله چالش‌های موجود در این مجموعه‌داده می‌توان به وجود زمینه‌های متنوع و پیچیده در تصاویر، تنوع دسته‌های مختلف اجسام موجود، تنوع اندازه اجسام موجود از یک دسته، تنوع حالت اجسام، تنوع تعداد نمونه‌های یک جسم در یک تصویر و کوچک بودن بیش از اندازه ابعاد اجسام در تصویر اشاره کرد. شکل ۱۰-۲^{۱۰} نمونه‌ای از تصاویر موجود در این مجموعه‌داده را نمایش می‌دهد.

استفاده از مدل کامل ارائه شده در این پژوهش، منجر به تشخیص صحیح ۷۳.۴٪ از تصاویر شده است. شکل ۱۱-۲^{۱۱} ماتریس درهم‌ریختگی^{۳۳} مربوط به این مدل را نمایش می‌دهد. همان‌طور که در این ماتریس مشخص است، کمترین نرخ تشخیص در بین دسته‌های ورزشی موجود در این مدل، ۵۲٪ و بیشترین نرخ تشخیص ۹۲٪ است.

بسته به میزان استفاده از اطلاعات مختلف استخراج شده برای استنتاج، مدل‌های مختلفی به وجود می‌آیند که در شکل ۱۲-۲^{۱۲} نتایج عملکرد هریک از این مدل‌ها با مدل‌های دیگر مقایسه شده است. همان‌طور که در شکل ۱۲-۲^{۱۲} مشخص است، بهترین کارایی مربوط به مدل کامل است. در صورتی که در مدل، فقط از اطلاعات مربوط به صحنه استفاده شود، نتایج بدست آمده اگرچه با نتایج مدل کامل قابل مقایسه نیست، از نتایج مدل مبتنی بر اطلاعات جسم بهتر است.

شکل ۱۳-۲^{۱۳} نتایج نهایی به دست آمده از مدل را نمایش می‌دهد. در این شکل، تصاویر موجود در هر سطر نماینده تصاویر موجود در یکی از دسته‌های ورزشی هستند. ستون اول برچسب به دست آمده از رخداد موجود در تصویر، ستون دوم برچسب‌های تشخیص داده شده مربوط به اجسام موجود، ستون سوم برچسب اختصاص داده شده مربوط به دسته صحنه و ستون چهارم توزیع مرتب شده اجسام به شرط رخداد را به نمایش می‌گذارند. در نمودارهای موجود در ستون چهارم، محور افقی شامل نام اجسام و محور عمودی مقدار توزیع را نمایش می‌دهد.

^{۳۱}Variational Message Passing

^{۳۲}Gibbs Sampling

^{۳۳}Confusion Matrix

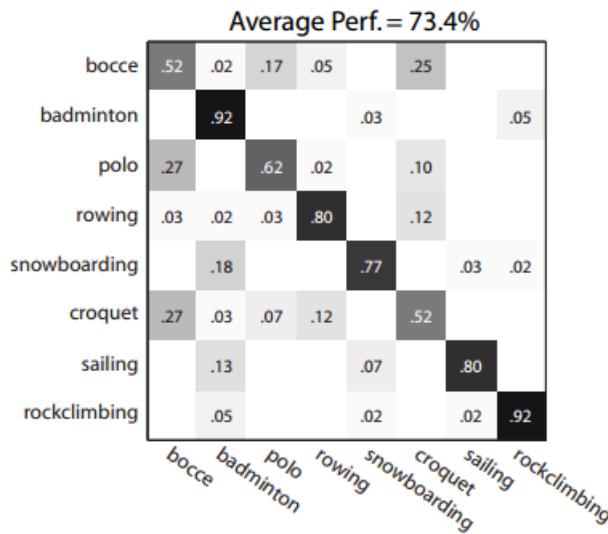


شکل ۲-۱۰: نمونه تصاویر موجود در مجموعه‌داده مورد استفاده [۲۰]

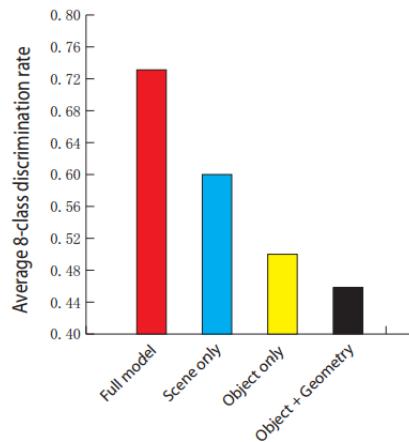
۴-۲ روش‌های مبتنی بر شبکه‌های عصبی کانولوشنی عمیق

علاوه بر فعالیت‌هایی که در زمینه تولید خودکار شرح بر تصاویر با استفاده از مدل‌های گرافی احتمالی انجام شده‌اند، تعداد زیادی از پژوهش‌گران تلاش می‌کنند تا با استفاده از روش‌های مبتنی بر شبکه‌های عصبی با این چالش روبرو شوند. در این بخش تعدادی از پژوهش‌هایی را که با استفاده از شبکه‌های عصبی سعی در درک صحنه‌های موجود در تصاویر دارند را مورد بررسی قرار می‌دهیم.

یکی از مهم‌ترین بخش‌هایی که به نحوی در پژوهش‌های قبلی انجام می‌شد، اختصاص یک معنا به قطعه‌های مختلف یک تصویر است. این چالش، در پژوهش‌های مرتبط با تولید خودکار شرح بر تصاویر که با استفاده از روش‌های مبتنی بر شبکه‌های عصبی به دنبال حل مشکل هستند نیز مطرح است. در ابتدا به بررسی یکی از روش‌های اختصاص معنا به هر قطعه از تصویر می‌پردازیم.



شکل ۱۱-۲: ماتریس درهم‌ریختگی مدل کامل ارائه شده برای مجموعه‌داده شامل ۸ دسته تصویر ورزشی. [۲۰]



شکل ۱۲-۲: نتیجه مقایسه مدل‌های مختلف به وجود آمده بسته به سطح اطلاعات مورد استفاده برای استنتاج. [۲۰]

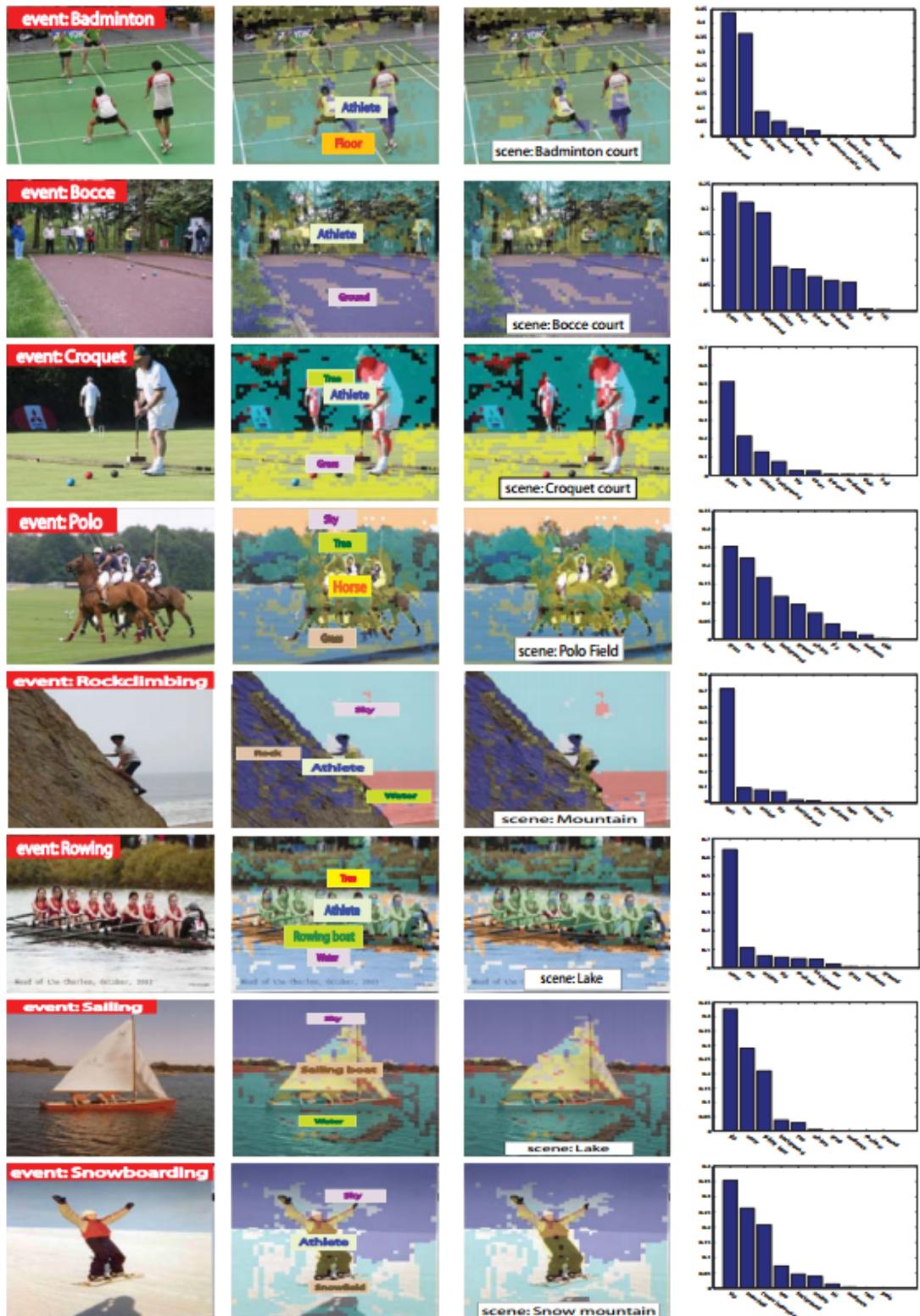
۱-۴-۲ اختصاص معنا به قطعه‌های مختلف تصویر [۱۰]

در پژوهش [۱۰] که توسط گرشیک و همکارانش در سال ۲۰۱۴ انجام شده، روشی ارائه شده است که با استفاده از یک شبکه عصبی کانولوشنی عمیق، علاوه بر این که می‌تواند یک تصویر را به شکل پایین به بالا، در قالب نواحی سلسله‌مراتبی قطعه‌بندی کند، قادر به استفاده به عنوان یک شبکه از پیش آموزش دیده شده در پژوهش‌های مرتبط دیگر باشد.

فرایند تشخیص اجسام در این پژوهش از سه بخش اصلی تشکیل شده است:

۱. طرح پیشنهاداتی برای نواحی به طور مستقل از دسته‌بندی^{۳۴}
۲. یک شبکه عصبی عمیق کانولوشنی که وظیفه استخراج ویژگی برای هر ناحیه را بر عهده دارد (طول بردار ویژگی استخراج شده برای تمام نواحی یکسان است).
۳. مجموعه‌ای از ماشین‌های بردار پشتیبان خطی مخصوص هر دسته

^{۳۴}Category-independent region proposals



شکل ۱۳-۲: نتایج نهایی به دست آمده از مدل بر روی تصاویر. [۲۰]

در ادامه به بررسی نحوه پیشنهاد نواحی و شبکه عصبی کانولوشنی عمیق مورد استفاده در ای پژوهش می‌پردازیم.

۱. طرح پیشنهاد نواحی

روش‌های مختلفی برای پیشنهاد نواحی ارائه شده‌اند که در اینجا از روشی موسوم به جستجوی انتخابی^{۳۵} استفاده می‌شود. نسخه‌های مختلفی از این روش ارائه شده است. نسخه ارائه شده در پژوهش [۳۶]، یکی از سریع‌ترین نسخه‌های ارائه شده است که در این بخش از همین روش استفاده می‌شود.

در پژوهش [۳۶] دو ویژگی مطرح شده است که یک جستجوی انتخابی برای ارائه نواحی معنایی تصویر باید آن‌ها را داشته باشد. ویژگی اول این است که اجسام موجود در فضای می‌توانند در هر اندازه‌ای باشند و در نتیجه نواحی ارائه شده باید بتوانند ابعاد مختلف داشته باشند. این ویژگی عموماً با روش‌های سلسله‌مراتبی قابل دست‌یابی است. ویژگی دوم این است که نواحی مختلف باید براساس ویژگی‌های مختلفی تولید شوند. در صورتی که یک ویژگی مثل رنگ، بافت، روشنایی یا مواردی از این دست، به عنوان تنها ویژگی برای تشخیص نواحی به کار گرفته شود، الگوریتم قادر به ارائه نواحی مناسب در شرایط مختلف نخواهد بود.

بنابراین ترکیب چند معیار و ویژگی باید برای تشخیص نواحی مورد استفاده قرار بگیرد.

برای دست‌یابی به ویژگی اول، ابتدا نواحی اولیه کوچکی روی تصویر ایجاد می‌شود. سپس با اتخاذ یک روش حریصانه و تعریف یک معیار شباهت بین نواحی همسایه، ناحیه‌هایی که شباهت زیادی با یکدیگر دارند و همسایه هستند، با هم ترکیب شده و یک ناحیه بزرگ‌تر ساخته می‌شود. به این ترتیب یک روش سلسله‌مراتبی برای ساخت نواحی با ابعاد مختلف به دست می‌آید. برای دست‌یابی به ویژگی دوم، از فضاهای رنگی مختلف، معیارهای شباهت مختلف و نواحی اولیه متفاوت و ترکیب پاسخ این ویژگی‌ها با هم برای ارائه نواحی و ترکیب نواحی کوچک‌تر استفاده می‌شود.

۲. شبکه عصبی کانولوشنی عمیق (استخراج ویژگی‌ها)

در این بخش از یک شبکه عصبی کانولوشنی عمیق از پیش‌آموزش دیده برای استخراج ویژگی از هر ناحیه ارائه شده در قسمت قبل، استفاده می‌شود. بردار ویژگی استخراج شده برای هر ناحیه یک بردار شامل ۴۰۹۶ مولفه است که خروجی شبکه کریشفسکی^{۳۷} آزمایش شده در چالش دسته‌بندی اجسام مسابقه ImageNet است. اطلاعات دقیق درباره این شبکه عصبی در پژوهش [۱۶] در دسترس است.

شبکه عصبی کانولوشنی عمیق ارائه شده در این پژوهش با استفاده از یک مجموعه‌داده^{۳۸} آموزش دیده شده است. از این شبکه عصبی که تحت عنوان RCNN^{۳۹} شناخته می‌شود می‌توان به عنوان یک شبکه از پیش‌آموزش دیده استفاده کرد.

۲-۴-۲ ناحیه‌بندی عمیق تصاویر به منظور نگاشت دوطرفه جملات و تصاویر [۱۶]

مدل ارائه شده در این پژوهش، مدلی است که قادر به نگاشت دوطرفه تصاویر و جملات به یکدیگر است. شکل ۱۴-۲ طرح‌واره‌ای از این مدل را نمایش می‌دهد. ورودی مدل در سمت چپ، تصاویر، تصاویر و در سمت راست، جملات هستند. در این مدل، ابتدا تصاویر ورودی با استفاده از یک شبکه عصبی RCNN تبدیل به نواحی مختلف شده و برای هر ناحیه یک بردار ویژگی ۴۰۹۶ بعدی استخراج می‌شود. سپس با اعمال روش خاصی روی جملات ورودی از سمت راست (که در بخش تولید جملات زبان طبیعی به بررسی آن خواهیم پرداخت) قطعات مختلف موجود

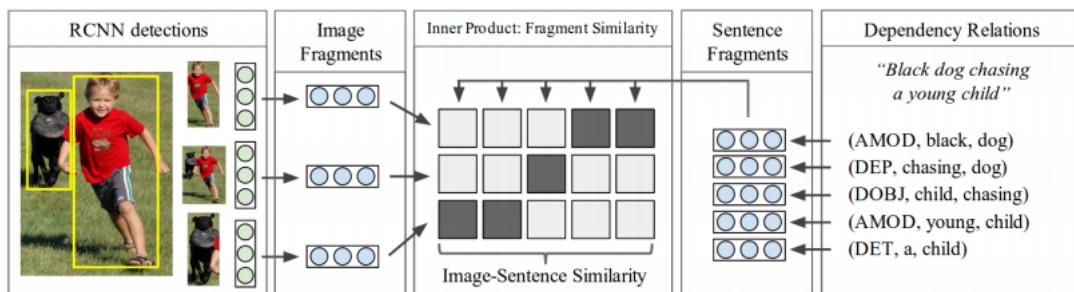
^{۳۵}Selective Search

^{۳۶}Krizhevsky

^{۳۷}ILSVRC 2012

^{۳۸}Regional Convolutional Neural Network

در جملات نیز استخراج شده و بین هر قطعه از جمله با تمام نواحی استخراج شده از تصویر یک معیار شباهت محاسبه می‌شود و شبیه‌ترین قطعه جمله با ناحیه مربوط به خود در تصویر، جفت می‌شوند.



شکل ۱۴-۲: مدل استفاده شده برای نگاشت دوطرفه تصاویر و جملات به یکدیگر با استفاده از شبکه عصبی عمیق کانولوشنی.^{۱۴}

در این پژوهش پس از ناحیه‌بندی تصویر توسط شبکه RCNN، برای هر تصویر ۱۹ ناحیه استخراج می‌شود. این ۱۹ ناحیه در کنار تصویر اصلی، یک مجموعه شامل ۲۰ تصویر ایجاد می‌کنند که در پردازش‌های بعدی مورد استفاده قرار خواهند گرفت. در این مرحله باید تمام تصاویر موجود را با استفاده از یک نگاشت به فضای برداری ویژگی‌ها تبدیل نمود. برای این کار از رابطه $14-2$ استفاده می‌شود. در این رابطه، I_b مجموعه تمام پیکسل‌های موجود در ناحیه b . شبکه عصبی آموزش‌دیده است که در آن θ_c شبکه $RCNN_{\theta_c}$ برای تصویر b ، بردار نگاشت تصویر به فضای معنایی خواهد بود که محاسبه مقادیر آن مبتنی بر پیشنهاد نواحی معنایی مختلف و محاسبه ویژگی‌های مختلف روی هر ناحیه است.

$$\nu = W_m[RCNN_{\theta_c}(I_b)] + b_m \quad (14-2)$$

از طرفی با در نظر گرفتن بردار s_j به عنوان بردار حاصل از نگاشت جمله زام به فضای معنایی و در نظر گرفتن ضرب داخلی به عنوان شباهت، $s_j \cdot \nu_i^T$ معیار شباهت بین یک تصویر و یک جمله را تعریف می‌کند. با توجه به توضیحات ارائه شده، می‌توان تابع هدف را برای شبکه کلی معادل سیستم ارائه داد. دو هدف اصلی در این شبکه قابل تعریف است:

۱. رتبه‌بندی سراسری تصاویر و جملاتی که در فرایند محاسبات شبکه عصبی بیشترین شباهت را با یکدیگر دارند باید در واقعیت هم بیشترین شباهت و ارتباط را داشته باشند.

۲. هم‌ترازسازی ناحیه‌ای^{۱۵} نواحی استخراج شده تصویر و عبارات استخراج شده جملات که در محاسبات شبکه عصبی بیشترین شباهت را با یکدیگر دارند، باید در واقعیت هم بیشترین شباهت و ارتباط را داشته باشند.

با توجه به مطالب گفته شده، می‌توان تابع هدف کلی را مطابق با رابطه $15-2$ تعریف کرد. در این رابطه، Θ مجموعه پارامترهای شبکه عصبی شامل $\{W_m, b_m, \theta_c, W_e, W_R\}$ است (پارامترهای W_e و W_R مربوط به C_G بخش تحلیل جمله هستند که در فصل مربوطه بررسی خواهند شد). تابع هدف هم‌ترازسازی ناحیه‌ای،

^{۱۴}Fragment Alignment

تابع هدف سراسری، α و β دو ابرپارامتر^{۴۰} (با آزمون و خطای تعیین می‌شوند) و $\|\Theta\|_2^2$ یک عبارت تنظیم‌کننده^{۴۱} هستند.

$$C(\Theta) = C_F(\Theta) + \beta C_G(\Theta) + \alpha \|\Theta\|_2^2 \quad (15-2)$$

در ادامه به تعریف هریک از اهداف بیان شده می‌پردازیم.

۱. هم‌ترازسازی ناحیه‌ای

هدف از هم‌ترازسازی ناحیه‌ای این است که اگر عبارتی از یک جمله با یک تصویر شباهت زیادی پیدا کرد، حداقل یک ناحیه از تصویر وجود داشته باشد که نمایش‌دهنده این عبارت باشد و بقیه نواحی تصویر، ارتباط کمی با این عبارت داشته باشند. به عبارت بهتر، در صورتی که شباهت یک عبارت از یک جمله با یک تصویر از حدی بیشتر شد، شباهت حداقل یکی از نواحی موجود در تصویر با این عبارت زیاد شده و شباهت بقیه نواحی تصویر با آن کم شود. این فرض در سه حالت، رد می‌شود. اولین حالت، حالتی است که در آن ناحیه‌ای که در واقعه نمایش‌دهنده عبارت است، توسط RCNN تشخیص داده نشده باشد. دومین حالت، حالتی است که عبارت موجود به هیچ بخشی از ویژگی‌های بصری تصویر اشاره نکند و آخرین حالت، حالتی است که عبارت توصیف‌کننده، در هیچ یک از تصاویر دیگر تکرار نشده باشد در صورتی که ممکن است تصاویر دیگری هم وجود داشته باشند که شامل ویژگی‌های بصری متناظر با عبارت باشند. با توجه به شرایطی که فرض در آن‌ها نقض می‌شود، می‌توان آن را یک فرض خوب تلقی کرد که در اکثر موارد عملکرد خوبی دارد. رابطه^{۱۶-۲} تابع هدف هم‌ترازسازی ناحیه‌ای را تعریف می‌کند. در این رابطه، y_{ij} برای تصویر i ام و جمله j ام در صورتی که با هم در مجموعه‌داده حضور داشته باشند، $+1$ و در غیر این صورت، -1 خواهد شد.

$$C_{\circ}(\Theta) = \sum_i \sum_j \max(0, 1 - y_{ij} \nu_i^T \cdot s_j) \quad (16-2)$$

تابع C_{\circ} تعریف شده، باعث می‌شود در حالاتی که تصویر و عبارت، در مجموعه‌داده، با یکدیگر وارد شده باشند امتیاز تابع هدف بیشتر از $+1$ شود و در غیر این صورت از -1 کمتر شود. شکل^{۱۵-۲}، دو نمونه از تصاویر و جملات موجود در مجموعه‌داده را نمایش می‌دهد. C_{\circ} در سلول‌هایی که با رنگ قرمز مشخص شده‌اند، امتیاز را به سمت کمتر از -1 حرکت می‌دهد و در بقیه سلول‌ها به سمت بیشتر از $+1$.

به عبارت بهتر، C_{\circ} یک امتیاز برای مجموع تفاوت‌های نواحی مختلف مختلط جملات است. به دلیل این‌که این معیار، باعث دیده نشدن موارد کمیاب می‌شود، با متغیر گرفتن پارامتر y_{ij} سعی در یافتن کمترین مقدار آن می‌کنیم. رابطه^{۱۷-۲} معیار متناظر با هدف کلی هم‌ترازسازی ناحیه‌ای را بیان می‌کند.

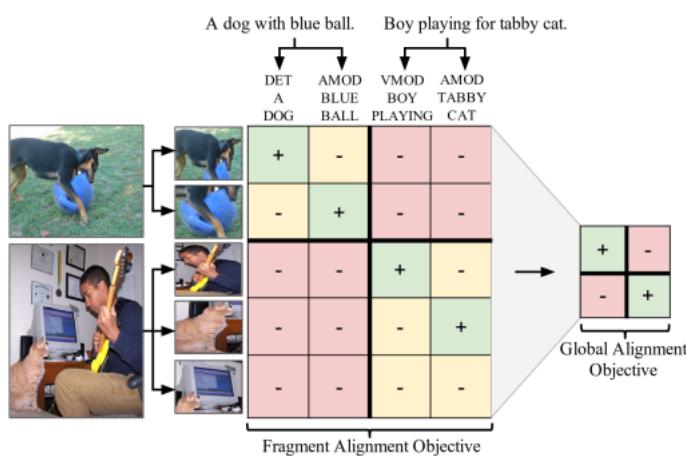
^{۴۰} Hyperparameter

^{۴۱} Regularization Term

$$C_F(\Theta) = \min_{y_{ij}} C_*(\Theta)$$

$$\text{s.t. } \sum_{i \in p_j} \frac{y_{ij} + 1}{2} \geq 1 \quad y_{ij} = -1, \forall i, j; m_\nu(i) \neq m_s(j) \wedge y_{ij} \in \{+1, -1\} \quad (17-2)$$

در این رابطه، p_j مجموعه تصاویر موجود در کیسه مثبت^{۴۲} مربوط به عبارت زام است. شایان ذکر است، تنها تصاویری که در مجموعه‌داده همراه با عبارت زام مشاهده شده‌اند در کیسه مثبت مربوط به این عبارت قرار می‌گیرند و بقیه تصاویر در کیسه منفی^{۴۳} این عبارت قرار می‌گیرند. (i) و (j) به ترتیب، شماره تصویر و عبارت را در مجموعه‌داده مشخص می‌کنند.



شکل ۱۵-۲: دو نمونه از تصاویر و جملات مرتبط با آن‌ها و نتایج عملکرد اهداف تعریف شده روی آن‌ها. سطرها نمایش دهنده نواحی مختلف تصویر و ستون‌ها نمایش دهنده قطعه‌های مختلف جملات هستند. سلوهای قرمز رنگ حالتی هستند که در آن‌ها $y_{ij} = 1$ است. سلوهای زرد نمایش دهنده اعضای کیسه‌های مثبت هستند که در آن‌ها $y_{ij} = -1$ است. [۱۴]

۲. رتبه‌بندی سراسری

هدف از رتبه‌بندی سراسری این است که شباهت بین یک تصویر و یک جمله، بیشینه شود اگر و تنها اگر تصویر و جمله در واقعیت نیز بیشترین شباهت را به یکدیگر داشته باشند. برای این منظور، ابتدا یک امتیاز شباهت بین یک تصویر و یک جمله تعریف می‌شود. این امتیاز مطابق با رابطه^{۱۸-۲} تعریف شده و برابر است با میانگین امتیاز شباهت دوبه‌دوی نواحی مختلف تصویر با عبارات مختلف جمله.

$$S_{kl} = \frac{1}{|g_k|(|g_l| + n)} \sum_{i \in g_k} \sum_{j \in g_l} \max(0, \nu_i^T \cdot s_j) \quad (18-2)$$

از آنجا که برای دسته‌بندی از روش mi_SVM استفاده می‌شود، تمام امتیازها به صفر محدود می‌شوند. مقدار n که در مخرج کسر اضافه شده است، به صورت تجربی و با آزمون و خطا به دست آمده که نتایج را

^{۴۲}Positive Bag

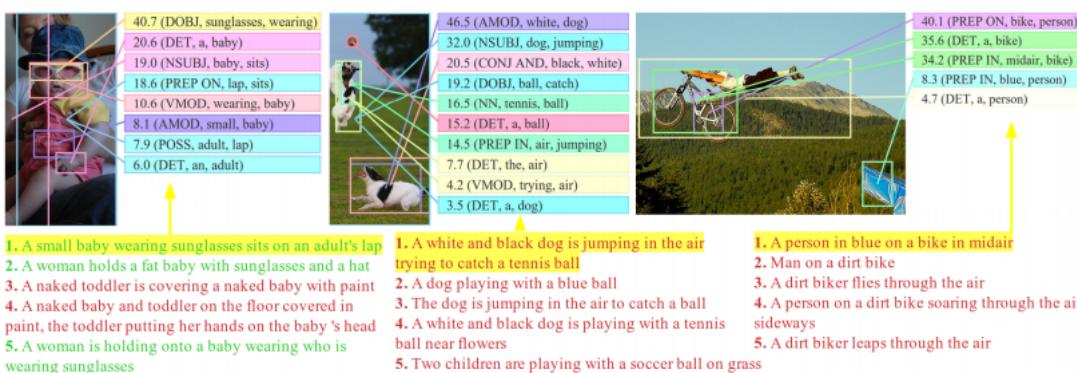
^{۴۳}Negative Bag

بهبود می‌بخشد. مقدار پیشنهاد شده در پژوهش، $n = 5$ است. تابع کلی هدف سراسری مطابق با رابطه ۱۹-۲ تعریف می‌شود.

$$C_G(\Theta) = \sum_k (\sum_l \max(0, S_{kl} - Skk + \Delta)) + \sum_l \max(0, S_{lk} - Skk + \Delta)) \quad (19-2)$$

در رابطه ارائه شده، Δ یک ابرپارامتر است که با آزمون و خطا به دست می‌آید. عبارت اول درون پرانتز بیان‌کننده امتیاز تصویر و عبارت دوم بیان‌کننده امتیاز جمله هستند.

شکل ۱۶-۲ نتایج روش پیشنهاد شده در این پژوهش را ارائه می‌دهد. همان‌طور که در شکل مشخص است، این شبکه قادر به تشخیص اجسام مختلف در تصویر و تولید یک سه‌تایی متناظر هر جسم (ناحیه معنایی) مبتنی بر جملات موجود در مجموعه‌داده مورد استفاده است.



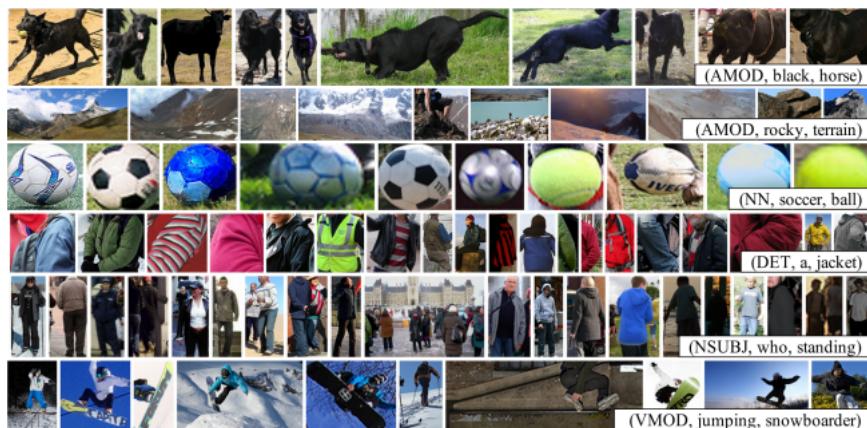
شکل ۱۶-۲: نتایج نهایی شبکه عصبی ارائه شده. برای هر ناحیه معنایی از تصویر، یک سه‌تایی مبتنی بر جملات موجود در مجموعه‌داده تولید شده است. همین‌طور ۵ جمله تولید شده برای هر تصویر به ترتیب امتیاز، درج شده‌اند.^[۱۴]

به علاوه، با توجه به مدل ارائه شده و نگاشت دوطرفه موجود بین تصاویر و جملات، می‌توان با ورودی دادن یک جمله، تصاویر مربوط به آن جمله را استخراج نمود. شکل ۱۷-۲ با ثابت در نظر گرفتن جملات، تصاویر مربوط به هر جمله را استخراج و نمایش داده است. هر سطر از این شکل، نمایش‌دهنده تصاویر استخراج شده مرتبط با جمله موجود در آن سطر است.

روش ارائه شده در این پژوهش، به طور کامل و دقیق در پژوهش [۱۲] هم مورد استفاده قرار گرفته است، با این تفاوت که در فرایند تحلیل جمله، تغییراتی ایجاد شده است. جزئیات این روش در فصل تولید جملات زبان طبیعی مورد بررسی قرار خواهد گرفت.

۳-۴-۲ مدل دوطرفه نگاشت تصاویر و جملات مبتنی بر یادگیری عمیق

یکی از مشکلات عمدۀ در روش‌های مبتنی بر یادگیری عمیق، وجود حافظه مناسب برای به خاطر سپاری رخدادهای گذشته است. در شبکه‌های عصبی پیش‌رو عمیق که دارای l لایه هستند، ظرفیت حداکثر حافظه موجود برای رخدادهای گذشته $1-l$ است و شبکه قادر است تنها $1-l$ رخداد گذشته را به خاطر سپارد. شبکه‌های عصبی بازگشتی، تا حد خوبی این مشکل را برطرف می‌نمایند. به همین دلیل، استفاده از این دسته از شبکه‌ها در بخش تولید جمله، منجر به ایجاد نتایج بهتر می‌شود.



شکل ۱۷-۲: نتایج حاصل از جستجوی جملات. با ورودی دادن یک جمله، شبکه عصبی ارائه شده در این پژوهش، قادر به استخراج تصاویر مربوط به آن جمله است.^[۱۴]

با این حال، شبکه‌های عصبی بازگشتی نیز در مواردی که طول جمله زیاد باشد، قادر به به خاطرسپاری مناسب رخدادهای گذشته نیستند. برای رفع این مشکل، معمولاً از واحدهای گیت در شبکه‌های عصبی حافظه کوتاه‌مدت بلند استفاده می‌شود. در پژوهش [۲۷]^{۴۴} که توسط خانم مایکولوف در سال ۲۰۱۰ ارائه شده است، شبکه عصبی ای ارائه شده است که بدون استفاده از واحدهای گیت، قادر به حفظ رخدادهای گذشته دور است. پژوهش [۲]^{۴۵} که در سال ۲۰۱۵ توسط آقای زیتنیک و همکارانش ارائه شده است، با استفاده از شبکه عصبی ارائه شده توسط خانم مایکولوف، مدلی دوطرفه برای نگاشت تصاویر و جملات به یکدیگر ارائه شده است که با داشتن تصویر قادر به تولید شرح متناظر و با داشتن شرح، قادر به بازسازی تصویر مربوطه است. در ادامه، ابتدا مدل مطرح شده توسط خانم مایکولوف را به طور مختصر شرح داده و سپس به بررسی مدل ارائه شده توسط آقای زیتنیک می‌پردازیم.

۴-۴-۲ مدل زبانی مبتنی بر شبکه عصبی بازگشتی

در این قسمت به بررسی مدل زبانی ارائه شده توسط خانم مایکولوف در پژوهش [۲۷]^{۴۶} می‌پردازیم. مدل ارائه شده در این پژوهش، یک مدل بسیار ساده از یک شبکه عصبی بازگشتی است. در لایه ورودی شبکه، کلمات موجود در جمله به ترتیب وارد می‌شوند. برای افزایش سرعت عملیات، به جای خود کلمات از نشان^{۴۵} در نظر گرفته شده برای کلمه استفاده می‌شود. برای محاسبه خروجی شبکه می‌توان از روابط (۲۰-۲) تا (۲۴-۲) استفاده نمود که در آن‌ها، $w(t)$ کلمه t ام موجود در جمله، $s(t-1)$ بردار حالت شبکه در زمان $t-1$ ، t ، $u_{j,i}$ وزن مربوط به اتصال ورودی واحد به بردار حالت شبکه، v_{kj} بردار وزن مربوط به بردار حالت شبکه و خروجی آن و u_k خروجی مرحله k ام مدل را نمایش می‌دهند.

^{۴۴}Mikolov

^{۴۵}Token

$$x(t) = W(t) + s(t - 1) \quad (20-2)$$

$$s_j(t) = f(\sum_i x_i(t) u_{ji}) \quad (21-2)$$

$$y_k(t) = g(\sum_j s_j(t) v_{kj}) \quad (22-2)$$

$$f(z) = \frac{1}{1 + e^{-z}} \quad (23-2)$$

$$g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}} \quad (24-2)$$

۵-۴-۲ مدل دوطرفه نگاشت تصاویر و جملات با استفاده از شبکه عصبی بازگشتی

در مدل ارائه شده در پژوهش [۲] که توسط آقای زیتنیک در سال ۲۰۱۵ ارائه شد، با تغییر مدل زبانی ارائه شده توسط خانم مایکولوف و تبدیل آن به یک مدل دوطرفه، روشی برای نگاشت دوطرفه تصاویر و جملات به یکدیگر ارائه شده است. در این بخش به بررسی این مدل و نحوه عمل کرد آن به طور اجمالی، خواهیم پرداخت.

در این پژوهش، دو متغیر جدید به مدل زبانی مطرح شده اضافه شده‌اند. متغیر V که بیان گر بردار ویژگی تصویر است و برای منوط کردن معنای جمله به ویژگی‌های تصویر مورد اسفاده قرار می‌گیرد و متغیر U که یک متغیر مخفی است و بیان گر تفسیر بصری آخرین کلمه مشاهده شده یا تولید شده است.

برای تولید یک مدل دوطرفه، کافیست بتوانیم احتمال رخداد جمله به شرط داشتن تصویر و همین‌طور احتمال رخداد تصویر به شرط جمله را محاسبه نماییم. همین‌طور این کار را می‌توان با بخش‌هایی از تصویر و کلمات جمله انجام داد؛ به این معنی که با مدل کردن احتمال رخداد بخش‌هایی از تصویر به شرط داشتن کلمه‌ای از جمله و همین‌طور احتمال رخداد کلمه‌ای در جمله با داشتن بخشی از تصویر به طور همزمان، یک نگاشت دوطرفه بین تصاویر و جملات مرتبط با آن‌ها ایجاد نماییم.

این کار را می‌توان مطابق با رابطه (۲۵-۲) انجام داد. این رابطه، محاسبه‌کننده میزان درست‌نمایی کلمه w_t و بردار ویژگی V به شرط داشتن کلمات قبلی W_{t-1} و تفسیر بصری هرکدام از آن‌ها U_{t-1} است.

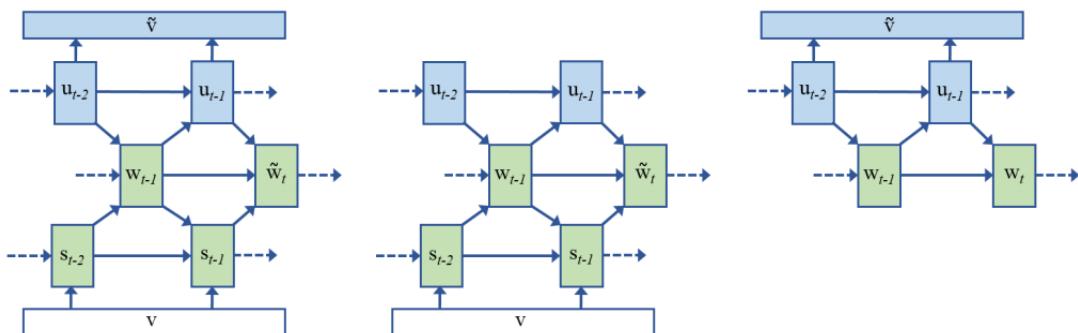
$$P(w_t, V | W_{t-1}, U_{t-1}) = P(w_t | V, W_{t-1}, U_{t-1}) P(V | W_{t-1}, U_{t-1}) \quad (25-2)$$

همان‌طور که در رابطه (۲۵-۲) مشخص است، می‌توان این رابطه را به شکل حاصل‌ضرب دو عبارت نوشت که هریک از آن‌ها قابلیت مدل‌شدن توسط یک شبکه عصبی بازگشتی را دارند. از طرفی متغیرهای مورد استفاده در هر دو عبارت یکسان است و فقط جهت محاسبات متفاوت است. این نکته باعث می‌شود بتوانیم از یک شبکه عصبی بازگشتی به شکل دوطرفه برای مدل‌سازی کامل رابطه درست‌نمایی توأم استفاده نماییم.

شکل ۱۸-۲ ساختار کلی شبکه ارائه شده در این پژوهش را نمایش می‌دهد. در این تصویر، شکل سمت چپ نمایش‌دهنده مدل به طور کامل است و شکل‌های وسط و سمت راست به ترتیب نمایش‌دهنده بخش‌هایی از مدل هستند که برای تولید جمله با داشتن تصویر و تولید تصویر با داشتن جمله مورد استفاده قرار می‌گیرند.

شکل ۱۸-۲ ساختار مدل زبانی ارائه شده توسط خانم مایکولوف را نمایش می‌دهد که متغیرهای V و W به آن اضافه شده‌اند. اضافه کردن یک لایه V به مدل زبانی، که در شکل با رنگ سفید مشخص شده است، این امکان را می‌دهد که اطلاعات مختلفی را بتوان در مدل زبانی در نظر گرفت. این اطلاعات می‌توانند اطلاعات مربوط

به نقش کلمات در جمله، مدل عنوان^{۴۶} و مواردی از این دست باشد. در این پژوهش از بردار ویژگی تصویر که مشخص کننده معنای تصویر است برای این قسمت استفاده شده است. این کار باعث می‌شود، معنای جمله تولید شده به محتوای تصویر منوط شود و این ضمانتی است که جمله تولید شده، توصیف کننده تصویر باشد.



شکل ۱۸-۲: ساختار کلی شیوه ارائه شده برای نگاشت دوطرفه تصاویر و جملات در پژوهش [۲]

مدل دوطرفه ارائه شده، روی سه مجموعه داده Flickr30K، MS COCO و Flickr8k آزمایش شده است. برای بررسی کیفیت عمل کرد مدل، باید در دو آزمایش مجزا، کیفیت نگاشت تصاویر به جملات و همین‌طور کیفیت نگاشت جملات به تصاویر توسط مدل، مورد بررسی قرار گیرند. در این قسمت قصد داریم با گزارش نتایج آزمایشات در قالب جداول و تصاویر، به بررسی عمل کرد مدل بپردازیم.

در جدول ۱-۲، مدل RNN + IF^{۴۷} یک شبکه عصبی بازگشتی است که ویژگی‌های استخراج شده از تصویر نیز به عنوان ورودی به آن داده شده است. مدل RNN + FT^{۴۸} شبکه عصبی با ورودی بردار ویژگی تصویر است که در آن خطای ایجاد شده از خروجی شبکه بازگشتی، به شبکه کانولوشنی نیز منتقل می‌شود و وزن‌های دوشبکه بازگشتی و کانولوشنی با هم به روزرسانی می‌شوند.

جدول ۱-۲: امتیاز BLEU کسب شده توسط مدل نگاشت دوطرفه ارائه شده در مقایسه با مدل‌های دیگر [۲].

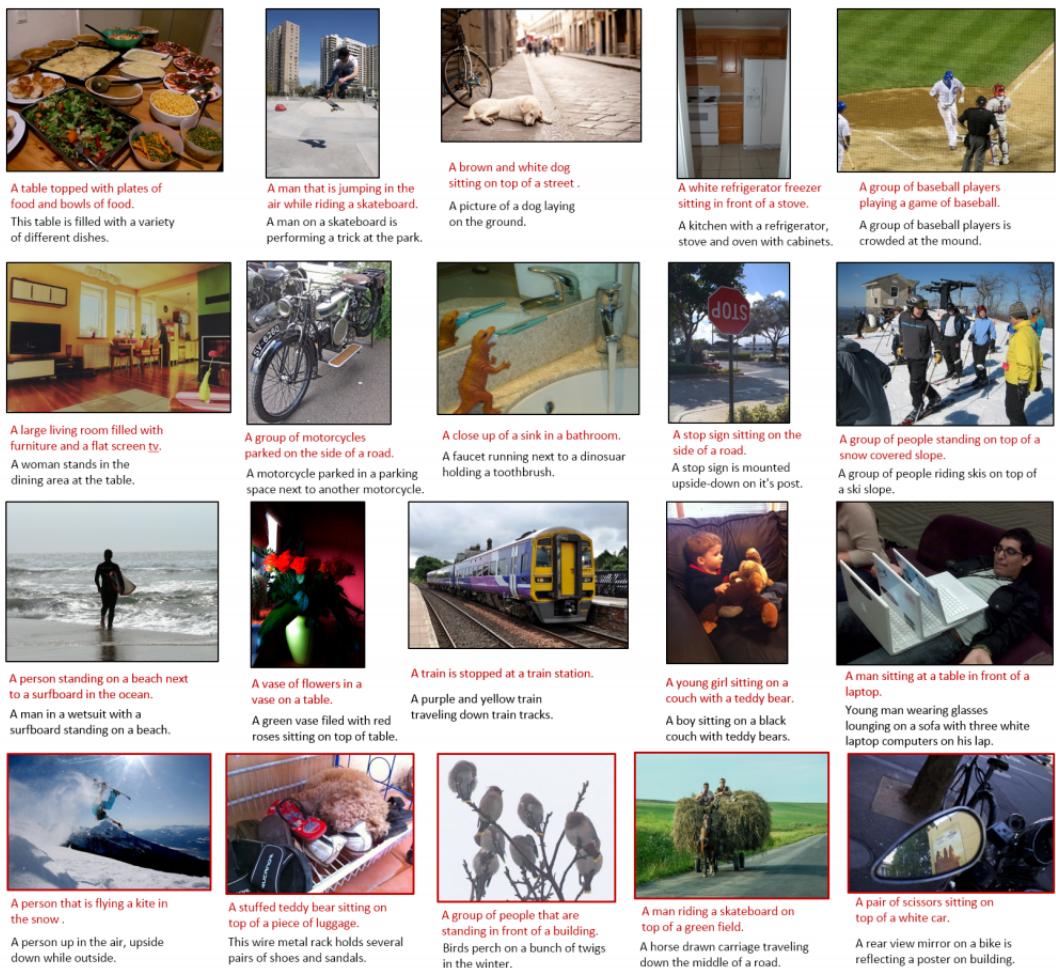
نام مدل	MS COCO	Flickr30k	Flickr8k
RNN	۴.۷	۶.۳	۴.۵
RNN + IF	۱۶.۳	۱۱.۳	۱۱.۹
RNN + IF + FT	۱۷.۰	۱۱.۶	۱۲.۰
RNN + VGG	۱۸.۴	۱۱.۹	۱۲.۴
روش ارائه شده	۱۶.۳	۱۱.۳	۱۲.۲
روش ارائه شده + FT	۱۶.۸	۱۱.۶	۱۲.۴
روش ارائه شده + VGG	۱۸.۸	۱۲.۰	۱۲.۱
انسان	۱۹.۲	۱۸.۹	۲۰.۶

علاوه بر جدول فوق که نتایج عمل کرد مدل پیشنهادی را در قالب میزان امتیاز BLEU نمایش داده و با مدل‌های دیگر مقایسه می‌کند، برای بررسی کیفیت عمل کرد مدل، شکل ۱۹-۲ جملات تولید شده مدل را با جملات نوشته شده توسط عوامل انسانی مورد مقایسه قرار می‌دهد. جملات قرمز رنگ در این تصویر، جملاتی هستند که توسط مدل ارائه شده تولید شده‌اند و جملات مشکی رنگ، جملاتی هستند که توسط عوامل انسانی نوشته شده‌اند. سطر آخر در این تصویر، نشان‌دهنده تعدادی از نمونه‌هایی است که در آن‌ها جملات تولید شده توسط مدل، چهار خطای شده‌اند.

^{۴۶}Topic Model

^{۴۷}Image Feature

^{۴۸}Fine Tuned



شکل ۲-۱۹: نمونه‌ای از جملات تولید شده برای تصاویر توسعه مدل پیشنهاد شده در [۲]

علاوه بر موارد فوق، جدول ۲-۲ نتایج بازیابی تصاویر با وارد کردن جمله را در این مدل با مدل‌های دیگر مورد مقایسه قرار می‌دهد. در مدل‌های ارائه شده در این جدول، استفاده از عبارت T در انتهای نام مدل، بیان‌گر این نکته است که در مدل مشخص شده، جملات بر اساس درستنمایی آن‌ها با داشتن تصویر ورودی مرتب شده‌اند. به علاوه، استفاده از عبارت I در نام مدل‌ها نمایان‌گر این نکته است که در این مدل‌ها، از خطای بازسازی تصویر نیز برای مرتب‌سازی جملات خروجی استفاده شده است.

جدول ۲-۲: جدول نتایج بازیابی تصاویر با استفاده از جملات ورودی در مدل ارائه شده در [۲]

نام مدل	R@1	R@5	R@10	Med r 500
M-RNN	۱۲.۶	۳۱.۲	۴۱.۵	۱۶
RNN + VGG	۱۵.۱	۴۱.۱	۵۴.۱	۹
T	۱۷.۷	۴۴.۹	۵۷.۲	۷.۵
T + I	۱۸.۵	۴۵.۷	۵۸.۱	۷

۵-۲ تولید شرح بر تصاویر با استفاده از روش‌های مبتنی بر توجه بصری

ایده اصلی روش‌های مبتنی بر توجه بصری از پژوهش‌های موجود در زمینه ترجمه ماشینی گرفته شده است. این دسته از پژوهش‌ها مدلی ارائه می‌دهند که با استفاده از آن بتوان هر کلمه از جملات تولیدی را با تمرکز بر یک یا بخشی از کلمات موجود در جمله مبدا، تولید کرد. به طور مشابه، در حوزه تولید خودکار شرح بر تصاویر، از این دسته از پژوهش‌ها به منظور حصول مدلی استفاده می‌شود که قادر باشد هر یک از کلمات موجود در جمله را با استفاده از بخشی از تصویر ورودی، تولید نماید.

در این فصل، ابتدا ایده اصلی ترجمه مبتنی بر توجه بصری را در حوزه ترجمه ماشینی ارائه خواهیم کرد و سپس کاربردهای این ایده را در حوزه تولید خودکار شرح بر تصاویر مورد بررسی قرار می‌دهیم.

۱-۵-۲ روش‌های مبتنی بر توجه بصری در حوزه ترجمه ماشینی

همان‌طور که گفته شد، تمام روش‌های قبلی را می‌توان به دو مرحله زیر تقسیم کرد.

۱. نگاشت نمونه‌ها از فضای تصاویر به فضای ویژگی‌ها

۲. نگاشت نمونه‌ها از فضای ویژگی‌ها به فضای جملات

در حوزه ترجمه ماشینی، به تابع نگاشت مرحله اول، رمزگذار^{۴۹} و به تابع نگاشت مرحله دوم، رمزگشا^{۵۰} گفته می‌شود. در این بخش، ما از این عبارات برای ارجاع به مراحل اول و دوم الگوریتم استفاده می‌نماییم. چارچوب کاری رمزگذار-رمزگشا، در تعداد زیادی از پژوهش‌های حوزه ترجمه ماشینی به عنوان چارچوب کاری اصلی مورد استفاده قرار گرفته است. تمام روش‌های قبلی که در فصول قبل ذکر شد نیز از همین چارچوب کاری به عنوان چارچوب اصلی بهره برده‌اند. به عنوان مثال در روش‌های مبتنی بر یادگیری عمیق برای تولید خودکار شرح بر تصاویر از شبکه‌های عصبی کانولوشنی به طور معمول به عنوان رمزگذار و از شبکه‌های عصبی بازگشتی به عنوان رمزگشا استفاده می‌شود.

شکل ۲۰-۲ نشان‌دهنده ساختار کلی چارچوب کاری رمزگذار-رمزگشا است.

در ادامه به بررسی بخش‌های مختلف این چارچوب کاری می‌پردازیم و سپس ایده اصلی روش‌های مبتنی بر توجه بصری را که توسط آقای بنجیو در پژوهش [۱] در سال ۲۰۱۴ ارائه شده است، مورد بررسی قرار خواهیم داد.

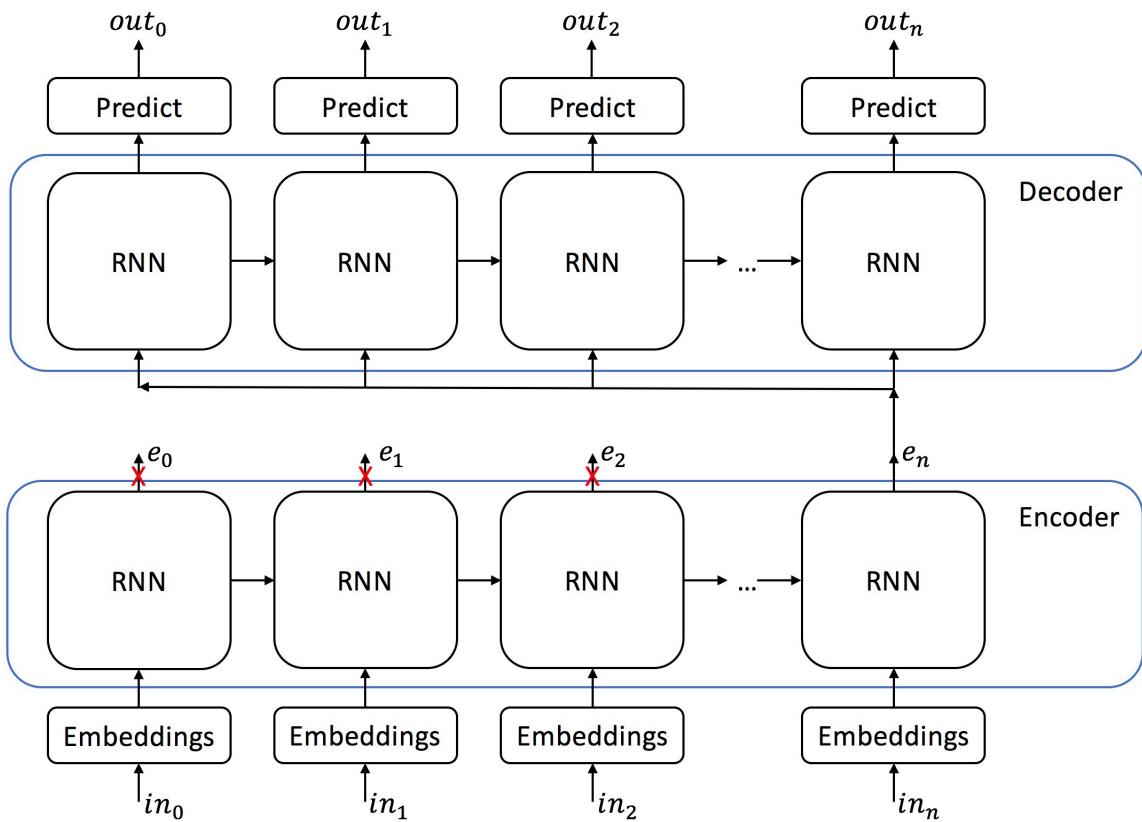
رمزگذار

رمزگذار در این چارچوب کاری، با گرفتن یک جمله به عنوان ورودی، بردار ویژگی متناظر جمله مبدا را تولید می‌کند. جمله ورودی با دنباله‌ای از کلمات مدل می‌شود. همین‌طور هر کلمه را با یک بردار n بعدی، که n تعداد کلمات موجود در دیکشنری است، مدل می‌شود. به این ترتیب، هر جمله ورودی، یک بردار با طول متغیر است که هر مولفه آن خودش برداری به ابعاد n است. از طرفی بردار خروجی، که همان بردار ویژگی‌ها است، یک بردار با طول ثابت و قراردادی خواهد بود.

عموماً در کاربردهای ترجمه ماشینی در هر دو بخش رمزگذار و رمزگشا از شبکه‌های عصبی بازگشتی استفاده می‌شود. در شبکه‌های عصبی بازگشتی، خروجی هر مرحله تابعی از ورودی آن مرحله و حالت شبکه در مرحله

^{۴۹}Encoder

^{۵۰}Decoder



شکل ۲۰-۲: ساختار کلی چارچوب کاری رمزگذار-رمزگشا

جاری است. با فرض این که h_t حالت شبکه در زمان t را نمایش دهد می‌توان رابطه تولید خروجی توسط شبکه عصبی بازگشتی را مطابق با (۲۶-۲) تعریف نمود.

$$\begin{aligned} h_t &= f(X_t, h_{t-1}) \\ C &= q(h_1, \dots, h_L) \end{aligned} \quad (26-2)$$

به طور معمول از شبکه LSTM به عنوان تابع f استفاده می‌شود و همین‌طور به جای استفاده از تابع q حالت نهایی شبکه به عنوان بردار ویژگی مورد استفاده قرار می‌گیرد [۱].

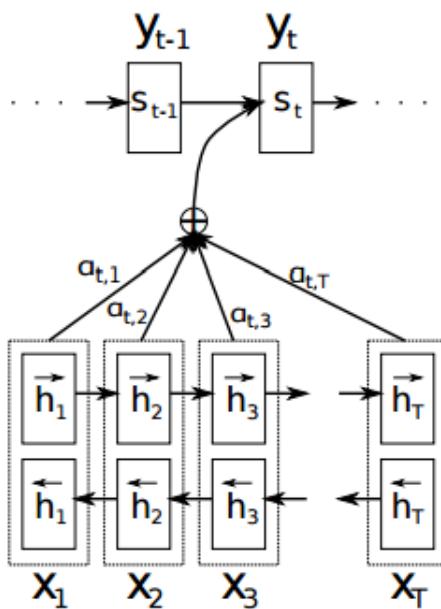
رمزگشا

رمزگشا به منظور نگاشت فضای ویژگی‌ها به فضای جملات مورد استفاده قرار می‌گیرد. خروجی رمزگذار، ورودی رمزگشا است. با این فرض، ورودی رمزگشا یک بردار ویژگی با طول ثابت است و خروجی آن که یک جمله به زبان مقصد است، همانند جمله مبدا، یک بردار با طول متغیر شامل بردارهای بازنمایی کلمات است. رمزگشا در اصل در هر مرحله، به دنبال یافتن کلمه‌ای است که با داشتن کلمات تولید شده قبلی و بردار ویژگی موجود، محتمل‌ترین کلمه نسبت به بقیه کلمات موجود در دیکشنری باشد. تابع احتمال مربوطه را می‌توان به فرم (۲۷-۲) تعریف

نمود.

$$p(y_t | C, y_1, y_2, \dots, y_{t-1}) = g(y_t, s_t, C) \quad (27-2)$$

در رابطه (27-2)، C نشان‌دهنده بردار ویژگی، y_i نشان‌دهنده لغت i ام تولید شده از زبان مقصد و بردار s_t نشان‌دهنده حالت شبکه بازگشتی مورد استفاده به عنوان رمزگشا است. شکل ۲۱-۲ ساختار کلی رمزگشا را نمایش می‌دهد.



شکل ۲۱-۲: ساختار رمزگشا مورد استفاده در چارچوب کاری [۱]

ایده اصلی استفاده از توجه بصری

همان‌طور که بیان شد، در چارچوب کاری رمزگذار-رمزگشا، ابتدا جمله ورودی که شامل تعداد نامعلوم کلمه است به یک بردار با طول متغیر مدل می‌شود. بردار تولید شده توسط یک رمزگذار به یک بردار با طول ثابت، که همان بردار ویژگی‌ها است، نگاشت شده و در نهایت بردار ویژگی تولید شده توسط یک رمزگذار به یک بردار با طول متغیر که نماینده جمله زبان مقصد است، نگاشت می‌شود.

فرآیند مذکور یک محدودیت جدی دارد و آن این است که رمزگذار باید بتواند تمام اطلاعات مورد نیاز برای تولید جمله را در یک بردار با طول ثابت بگنجاند و رمزگشا باید بتواند تمام اطلاعات مورد نیاز خود را از همین بردار با موجود با طول ثابت، استخراج کند. این محدودیت باعث می‌شود قدرت کد کردن اطلاعات در بردار ویژگی کاهش یابد. برای حل این مشکل از ایده نقاط توجه استفاده می‌نماییم.

در این دسته از روش‌ها به جای این که رمزگذار فقط یک بردار ویژگی تولید کند، بردارهای ویژگی مختلفی ایجاد می‌کند که هر بردار با تمرکز بر روی یک یا بخشی از جمله مبدا تولید شده است. به این ترتیب، هر بردار تولید شده شامل اطلاعات معنایی یک یا بخشی از جمله مبدا می‌باشد. به این طریق، رمزگشا می‌تواند با انتخاب بین

بردارهای معنایی تولید شده در هر مرحله، کلمه تولیدی را با تمرکز بر روی معنای یک کلمه و کلمات مجاور آن در جمله مبدأ، تولید کند.

در ادامه به بررسی تغییراتی که باید در رمزگذار و رمزگشنا اتفاق بیفتد تا بتوان به جای یک بردار ویژگی مجموعه‌ای از بردارهای ویژگی با تمرکز محلی ایجاد نمود و از آن‌ها برای تولید جمله استفاده نمود را مورد بررسی قرار می‌دهیم. برای سهولت فهم تغییرات، ابتدا تغییرات رمزگشنا را مطرح نموده و سپس به بررسی تغییرات رمزگذار خواهیم پرداخت.

رمزگشنا در روش مبتنی بر توجه بصری

فرض می‌کنیم به جای تنها یک بردار ویژگی، L بردار ویژگی از ورودی استخراج شده باشد. آن‌ها را در یک ماتریس به شکل $C = [c_1, \dots, c_L]^T$ بازنمایی می‌نماییم. فرض می‌کنیم بردار ویژگی c_i به دنباله حاشیه‌نویسی‌های^{۵۱} $[h_1, \dots, h_L]^T$ وابسته است. حاشیه‌نویسی h_i خود یک متغیر تصادفی به شکل برداری است که دارای دو ویژگی بسیار مهم می‌باشد.

۱. حاوی اطلاعات استخراج شده از تمام جمله است

۲. تمرکز استخراج اطلاعات بر روی کلمه نام و کلمات اطراف آن بوده است.

با تعریف این دو ویژگی، حاشیه‌نویسی‌ها را می‌توان همان بردار ویژگی جمله تصور کرد با این شرط که علاوه بر این که معنای کل جمله را کد کرده‌اند، تمرکز بیشتری بر معنای کلمه نام و کلمات مجاور آن دارند. به عبارت بهتر هر حاشیه‌نویسی علاوه بر این که معنای کلی جمله را کد می‌کند، حاوی معنای محلی مربوط به کلمات هم هست.

با تعریف حاشیه‌نویسی به شکل فوق و با تکیه بر فرض‌های انجام شده، می‌توانیم مدل احتمالاتی ارائه شده را به شکل (۲۸-۲) تغییر دهیم.

$$p(y_i|y_1, \dots, y_{i-1}, X) = g(y_{i-1}, S_i, c_i) \quad (28-2)$$

که در آن:

$$c_i = \sum_{j=1}^L \alpha_{ij} h_j \quad (29-2)$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^L \exp(e_{ik})} \quad (30-2)$$

$$e_{ij} = f(s_{i-1}, h_j) \quad (31-2)$$

متغیر تصادفی e_{ij} که در رابطه (۳۱-۲) تعریف شده است نمایان گر میزان شباهت کلمه نام در جمله خروجی به کلمه زام در جمله ورودی است. وظیفه این متغیر، هم‌ترازسازی^{۵۲} ورودی و خروجی است. α_{ij} یک نرمال‌سازی روی امتیازهای محاسبه شده انجام می‌دهد. از این متغیر نرمال شده به عنوان وزن حاشیه‌نویسی‌ها استفاده می‌شود. مطابق با رابطه (۲۹-۲) بردار ویژگی مورد استفاده برای تولید کلمه در جمله مقصد، از طریق یک میانگین‌گیری بر اساس وزن معنایی کلمات تولید می‌شود.

^{۵۱} Annotation

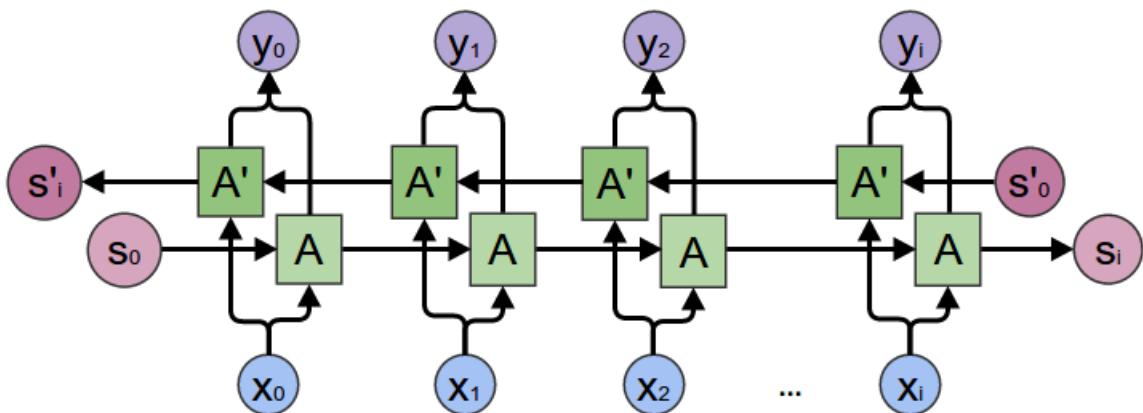
^{۵۲} Alignment

در رابطه (۳۱-۲) s_{i-1} بودار حالت شبکه رمزگشا در زمان $1 - i$ و f یک تابع امتیاز مورد استفاده در این رابطه را می‌توان با یک شبکه عصبی پیش‌رو^{۵۳} مدل‌سازی کرد. در صورت استفاده از شبکه عصبی پیش‌رو برای مدل‌سازی تابع شباهت، در صورتی که از هم‌ترازسازی نرم^{۵۴} استفاده شود، تابع هدف مشتق‌پذیر شده و می‌توانیم از الگوریتم پسانشتر خطا برای آموزش استفاده نماییم.

رمزگذار در روش مبتنی بر توجه بصری

برای طراحی رمزگذار در این بخش، باید مکانیزمی ارائه شود که قادر باشد حاشیه‌نویسی‌های h_L را طوری تولید کند که دو شرط مطرح شده در بخش قبلی را ارضاء نمایند. به عبارت دیگر باید بودارهای ویژگی‌ای استخراج نماییم که علاوه بر این که حاوی معنای کل جمله باشند، هر یک از آن‌ها بر روی معنای یک کلمه و کلمات اطراف آن تمکن بیشتری نسبت به سایر بودارها داشته باشند تا بتوانیم علاوه بر مدل‌سازی معنای کلی جمله، از معنای محلی کلمات هم استفاده نماییم.

به این منظور از یک شبکه عصبی بازگشتی دوطرفه در مدل‌سازی رمزگذار استفاده می‌نماییم. شکل ۲۲-۲ ساختار کلی یک شبکه عصبی بازگشتی دوطرفه را نمایش می‌دهد.



شکل ۲۲-۲: ساختار کلی یک شبکه عصبی بازگشتی دوطرفه

همان‌طور که در شکل ۲۲-۲ مشخص است، یک شبکه عصبی بازگشتی دوطرفه شامل دو شبکه پیش‌رو در خلاف جهت یک‌دیگر است. حالت‌های مخفی شبکه پیش‌رو را به راست را با h_i^+ و حالت‌های مخفی شبکه پیش‌رو را به چپ را با h_i^- نمایش می‌دهیم. همان‌طور که در شکل پیداست، خروجی‌های شبکه در این ساختار هم به حالت‌های سمت راست و کلمات سمت راست در جمله و هم به حالات و کلمات سمت چپ وابسته هستند. پس همین خروجی‌ها را می‌توان به عنوان حاشیه‌نویسی‌هایی که هر دو ویژگی را دارند مورد استفاده قرار داد. یکی از راههای ساده برای ایجاد حاشیه‌نویسی با استفاده از حالات شبکه‌های پیش‌رو را به راست و رو به چپ این است که مطابق با رابطه (۳۲-۲) با پشت سر هم قرار دادن حالات شبکه، حاشیه‌نویسی مورد نیاز را تولید نماییم.

$$h_j = [h_j^{\rightarrow T}, h_j^{\leftarrow T}]^T \quad (32-2)$$

^{۵۳}Feed Forward Neural Network

^{۵۴}Soft Alignment

۲-۵-۲ روش‌های مبتنی بر توجه بصری در حوزه تولید شرح متناظر تصویر

در بخش قبل به بیان ایده اصلی روش‌های مبتنی بر توجه بصری در حوزه ترجمه ماشینی پرداختیم. ساختار کلی رمزگذارها و رمزگشاها در این قالب و همین طور نحوه تولید بردارهای ویژگی مختلف از جمله مبدا و استفاده از این بردارها در تولید جمله مقصود را مورد بررسی قراردادیم. در این بخش به بررسی پژوهش‌های خواهیم پرداخت که از این ایده در حوزه تولید شرح متناظر تصویر بهره جسته‌اند.

یکی از برجسته‌ترین و مورد توجه ترین پژوهش‌ها از این دست، پژوهشی است که آقای بنجیو و همکارانش در سال ۲۰۱۵ ارائه داده‌اند [۳۶]. در این بخش به بررسی این پژوهش خواهیم پرداخت.

تولید شرح متناظر تصویر با استفاده از توجه بصری و شبکه‌های عصبی [۳۶]

در این پژوهش که در سال ۲۰۱۵ توسط آقای بنجیو و همکارانش ارائه شده است از ایده استفاده از توجه در حوزه ترجمه ماشینی استفاده شده است تا شرح متناظر تصاویر با دقت بیشتری تولید شود. چارچوب کاری رمزگذار-رمزگشا مانند آن‌چه در بخش قبلی مطرح شد در این پژوهش مورد استفاده قرار گرفته است. رمزگذار ارائه شده در این پژوهش، یک شبکه عصبی کانولوشنی است که قادر به تولید L بردار ویژگی مختلف است. به هر یک از این بردارهای ویژگی یک حاشیه‌نویسی^{۵۵} تصویر گفته می‌شود. بردارهای حاشیه‌نویسی، همان‌طور که در بخش قبل ذکر شد، باید دارای دو شرط زیر باشند:

۱. حاوی معنای تصویر به طور کلی باشند.
۲. تمکن بیشتری روی یکی از بخش‌های تصویر داشته باشند.

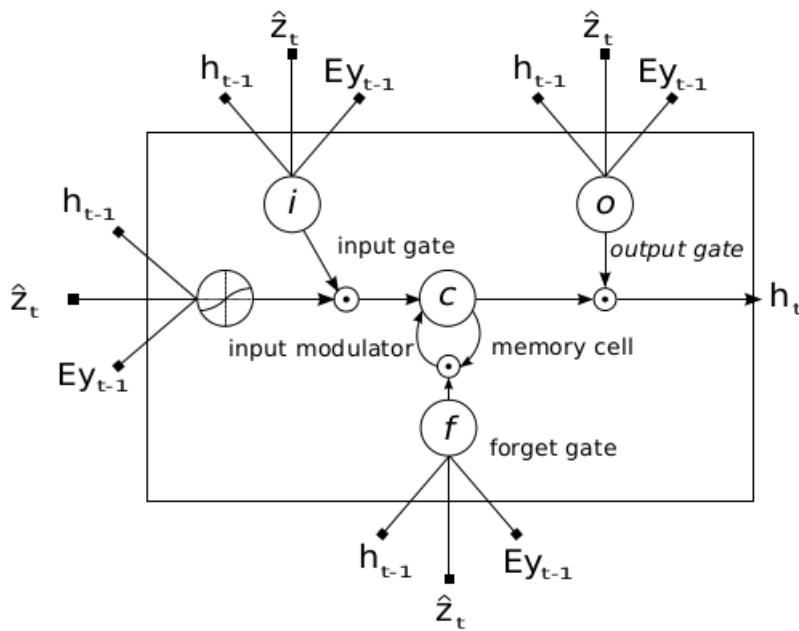
برای این‌که بتوانیم دو شرط فوق را در بردارهای حاشیه‌نویسی تولید شده از رمزگذار بگنجانیم از خروجی لایه ما قبل آخر شبکه عصبی کانولوشنی به عنوان بردارهای حاشیه‌نویسی استفاده می‌کنیم. هر بردار حاشیه‌نویسی یک بردار D بعدی است که مربوط به یک بخش از تصویر می‌شود و آن را با a_i نمایش می‌دهیم. بنابر این داریم:

$$a = \{a_1, a_2, \dots, a_L\}, a_i \in R^D \quad (۳۳-۲)$$

در این پژوهش از یک شبکه حافظه کوتاه‌مدت بلند به عنوان رمزگشا استفاده شده است. این شبکه با دریافت مجموعه بردارهای حاشیه‌نویسی a ، جمله‌ای به زبان انگلیسی تولید می‌کند که شامل دنباله‌ای از C کلمه است. هر کلمه با یک بردار K بعدی نمایش داده می‌شود که K تعداد کلمات موجود در دیکشنری است. در هر یک از بردارهای بازنمایی کلمات فقط یک مولفه یک است و مابقی مولفه‌ها صفر هستند. مولفه‌ای که برابر با یک است نمایش‌دهنده اندیس کلمه در دیکشنری است.

شکل ۲۳-۲ یک سلول از شبکه حافظه کوتاه‌مدت بلند مورد استفاده در این پژوهش به عنوان رمزگشا را نمایش می‌دهد. روابط مربوط به یادگیری این شبکه را می‌توان مطابق با روابط (۳۴-۲) تا ۳۹-۲ نمایش داد. در همه روابط،تابع T یک تابع نگاشت خطی به شکل $T : R^{D+m+n} * R^n \rightarrow R^D$ است که پارامترهای آن آموخت داده شده‌اند.

^{۵۵}Annotation



شکل ۲۳-۲: یک واحد از شبکه حافظه کوتاه‌مدت بلند مورد استفاده در رمزگشا پژوهش [۳۶]

متغیر i_t ورودی، f_t خروجی سلول فراموشی، c_t حافظه، o_t خروجی و h_t حالت مخفی شبکه را نمایش می‌دهند.

$$i_t = \sigma(T(Ey_{t-1}, h_{t-1}, \hat{z}_{t-1})) \quad (34-2)$$

$$f_t = \sigma(T(Ey_{t-1}, h_{t-1}, \hat{z}_{t-1})) \quad (35-2)$$

$$o_t = \sigma(T(Ey_{t-1}, h_{t-1}, \hat{z}_{t-1})) \quad (36-2)$$

$$g_t = \sigma(T(Ey_{t-1}, h_{t-1}, \hat{z}_{t-1})) \quad (37-2)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (38-2)$$

$$h_t = o_t \odot \tanh(c_t) \quad (39-2)$$

بردار \hat{z}_{t-1} بردار معنای تصویر را نمایش می‌دهد که با استفاده از بردارهای حاسیه‌نویسی تولید شده در رمزگذار تولید می‌شود. ماتریس E ، ماتریس جانمایی^{۵۶} به ابعاد $m * K$ است. تابع σ تابع فعالیت سیگموئیدی و \odot حاصل ضرب مولفه‌های نظیر به نظیر بردارها را نمایش می‌دهند.

فرایند آموزش رمزگشا کاملاً مطابق با فرایند آموزش معمول شبکه حافظه کوتاه‌مدت بلند است. تنها تفاوت در این پژوهش وجود و نحوه محاسبه بردار معنای \hat{z}_{t-1} است که توجه بصری را تعریف می‌کند. برای محاسبه این متغیر تابع ϕ را تعریف می‌نماییم. این تابع در هر لحظه از زمان با استفاده از مجموعه بردارهای حاسیه‌نویسی a برداری تولید می‌کند که به عنوان بردار ویژگی استخراج شده از تصویر در هر لحظه مورد استفاده قرار می‌گیرد. تابع phi می‌تواند به دو شکل بردار ویژگی را تولید نماید. روش اول این است که ابتدا با تولید وزن‌های مثبت برای هر ناحیه از تصویر با بردار حاسیه‌نویسی a_i یک احتمال برای میزان مناسب بودن ناحیه i از تصویر برای

^{۵۶}Embedding Matrix

استفاده در تولید کلمه در زمان t تعریف شود. سپس بردار حاشیه‌نویسی با بیشترین احتمال برای تولید کلمه انتخاب شده و به مراحل بعدی ارسال شود. این روش را تحت عنوان روش توجه سخت^{۵۷} نام‌گذاری می‌نماییم. روش دوم برای تولید بردار ویژگی تصویر در هر لحظه توسطتابع ϕ این است که اعداد مثبت تولید شده α_i را به طور مستقیم به عنوان معیاری جهت سنجش میزان مناسببودن نسبی نواحی نسبت به یکدیگر مورد استفاده قرار دهیم و با استفاده از یک میانگین‌گیری وزن‌دار بر حسب همین وزن‌های مثبت از بردارهای حاشیه‌نویسی اقدام به تولید بردار ویژگی تصویر نماییم. به این روش، روش توجه نرم^{۵۸} می‌گوییم.

به جهت سهولت در امر رابطه‌بندی توجه بصری در فرایند آموزش رمزگذار و رمزگشا، متغیر تصادفی $s_{t,i}$ را معرفی می‌نماییم که نشان‌دهنده این است که آیا در زمان t ، از بردار ویژگی مربوط به ناحیه i ام تصویر، برای تولید کلمه استفاده می‌شود یا خیر. اگر در زمان t از بردار ویژگی ناحیه i ام، که همان بردار حاشیه‌نویسی با آندیس i است، به منظور تولید کلمه استفاده شود، مقدار متغیر $s_{t,i}$ برابر با یک و در غیر این صورت برابر با صفر قرار می‌گیرد. با استفاده از متغیر تصادفی تعریف شده می‌توان به راحتی روابط مربوط به مدل‌سازی توجه بصری نرم و سخت را به شرح زیر تشکیل داد. نکته آخر این‌که در پژوهش موردنبررسی، به منظور تولید احتمال کلمه بعدی با توجه به کلمات تولید شده قبلی و بردار ویژگی استخراج شده از تصویر از رابطه (۴۰-۲) استفاده شده است.

$$p(Y_t|a, Y_1^{t-1}) \propto \exp(L_o(EY_{t-1} + L_h h_t + L_z \hat{z}_t)) \quad (40-2)$$

در رابطه (۳۹-۲) ماتریس‌های E ، $L_z \in R^{m*d}$ ، $L_h \in R^{m*n}$ ، $L_o \in R^{K*m}$ پارامترهای شبکه هستند که باید آموزش داده شوند.

۱. توجه بصری سخت

با فرض یک توزیع Multinoulli مطابق رابطه (۴۱-۲)^{۴۱} می‌توان متغیر \hat{z}_t را به عنوان بردار ویژگی استخراج شده نهایی با توجه به بردارهای حاشیه‌نویسی a و توجه بصری $s_{t,i}$ به شکل رابطه (۴۲-۲)^{۴۲} محاسبه نمود.

$$p(s_{t,i} = 1 | s_{j < t}, a) = \alpha_{t,i} \quad (41-2)$$

$$\hat{z}_t = \sum_t s_{t,i} a_i \quad (42-2)$$

برای آموزش وزن‌های شبکه یکتابع هدف به نام L_s مطابق با رابطه (۴۳-۲)^{۴۳} مطرح می‌شود که یک کران پایین از بیشینه درستنمایی $p(Y|a)$ است که در آن Y دنباله کلمات تولید شده نهایی و a بردارهای حاشیه‌نویسی تولید شده از روی تصویر را نمایش می‌دهند.

$$\log p(Y|a) = \log \sum_s p(s|a)p(Y|s, a) \geq \sum_s p(s|a) \log p(Y|s, a) = L_s \quad (43-2)$$

با ارائه تابع هدف L_s مطابق با رابطه (۴۳-۲) و بهینه‌سازی آن می‌توان رابطه بهروزرسانی وزن‌ها در فرایند

^{۵۷}Hard Attention

^{۵۸}Soft Attention

آموزش را محاسبه نمود. رابطه (۴۴-۲) محاسبات مربوطه را نمایش می‌دهد.

$$\frac{\partial L_s}{\partial W} = \Sigma_s p(s|a) \left[\frac{\partial \log p(Y|s,a)}{\partial W} + \log p(Y|s,a) \frac{\partial \log p(s|a)}{\partial W} \right] \quad (44-2)$$

به جای متغیر s_t در رابطه (۴۴-۲) می‌توان با استفاده از روش نمونه‌برداری مونت کارلو^{۵۹} نمونه‌های تصادفی \tilde{s}_t تولید کرد و سپس با استفاده از رابطه (۴۵-۲) تابع هدف را بهینه نمود.

$$\frac{\partial L_s}{\partial W} \approx \frac{1}{N} \Sigma_{n=1}^N \left[\frac{\partial \log p(Y|\tilde{s}^n,a)}{\partial W} + \log p(Y|\tilde{s}^n,a) \frac{\partial \log p(\tilde{s}^n|a)}{\partial W} \right] \quad (45-2)$$

۲. توجه بصری نرم

همان‌طور که گفته شد، تولید بردار ویژگی را می‌توان با میانگین‌گیری وزن‌دار روی بردارهای حاشیه‌نویسی انجام داد. در شرایطی که وزن‌های تخصیص داده شده به بردارهای حاشیه‌نویسی برابر صفر نباشند، توجه بصری نرم، فرایندی خواهد بود شامل تولید بردار ویژگی با استفاده از تمام حاشیه‌نویسی‌های موجود و با تمرکز روی تعدادی از حاشیه‌نویسی‌ها که ضریب بیشتری دارند. از آنجا که این شیوه محاسبه بردار ویژگی شامل یک بردار میانگین‌گیری وزن‌دار است، تمام تابع هدف مشتق‌پذیر شده و امکان استفاده از روش پسانشان خطا برای یادگیری وزن‌ها فراهم می‌شود.

در این روش به طور کلی می‌توان بردار ویژگی \hat{z}_t را مطابق با رابطه (۴۶-۲) محاسبه نمود.

$$E_{p(s_t|a)}[\hat{z}_t] = \Sigma_{i=1}^L \alpha_{t,i} a_i \quad (46-2)$$

مطابق با رابطه (۳۷-۲) حالت مخفی شبکه یک ترکیب خطی از بردار ویژگی استخراج شده از تصویر به همراه یک غیرخطی‌سازی با استفاده از تابع $tanh$ است. برای تقریب مرتبه اول حالت مخفی شبکه می‌توان از امید ریاضی بردار ویژگی \hat{z}_t در رابطه (۳۷-۲) استفاده کرد. با در نظر گرفتن رابطه (۳۹-۲) می‌توان متغیر n_t را به شکل $n_t = L_o(EY_{t-1} + L_h h_t + L_z \hat{z}_t)$ تعریف نمود. با این تعریف، متغیر $n_{t,i}$ مشخص‌کننده متغیر n_t است در شرایطی که $a_i = \hat{z}_t$ باشد. با استفاده از متغیر تعریف شده، میانگین هندسی وزن‌دار نرمال‌شده^{۶۰} را برای تولید کلمه k مطابق با رابطه (۴۷-۲) تعریف می‌نماییم.

$$NWGM[P(y_t = k|a)] = \frac{\Pi_i \exp(n_{t,k,i})^{p(s_{t,i}=1|a)}}{\sum_j \Pi_i \exp(n_{t,j,i})^{p(s_{t,i}=1|a)}} = \frac{\exp(E_{p(s_{t,i}|a)}[n_{t,k}])}{\sum_j \exp(E_{p(s_{t,i}|a)}[n_{t,j}])} \quad (47-2)$$

از آنجا که $E[n_t] = L_o(EY_{t-1} + L_h E[h_t] + L_z E[\hat{z}_t])$ می‌تواند به خوبی توسط بردار ویژگی \hat{z}_t تخمین زده شود. این بدین معناست که میانگین هندسی وزن‌دار نرمال‌شده لایه نهایی

^{۵۹}Monte Carlo Sampling

^{۶۰}Normalized Weighted Geometric Mean (NWGM)

شبکه می‌تواند با اعمال تابع $soft\ max$ به امید ریاضی ترکیبات خطی لایه‌های پایین‌تر محاسبه شود.

آزمایشات انجام شده در این پژوهش روی سه مجموعه‌داده Flickr30k، Flickr8k و Microsoft COCO اجرا شده است که به ترتیب شامل ۳۰۰۰۰، ۸۰۰۰ و ۸۲۷۸۳ تصویر با شرح تولیدشده توسط عوامل انسانی هستند. دو مجموعه‌داده اول برای هر تصویر، ۵ شرح مختلف و مجموعه‌داده سوم در برخی تصاویر بیش از ۵ شرح را شامل می‌شوند. در تمام پژوهش‌ها به منظور یکسان‌سازی آزمایشات و نتایج، از ۵ شرح برای هر تصویر استفاده شده است.

هر دو نوع محاسبه توجه بصری در این پژوهش مورد آزمایش قرار گرفته‌اند و نتایج هریک به طور جداگانه بیان شده است. در این پژوهش از معیارهای METEOR و BLEU به منظور ارزیابی مدل استفاده شده است. همان‌طور که در جدول ۲-۲ مشخص است، در هر سه مجموعه‌داده، پژوهش [۳۶] بهترین عملکرد را نسبت به روش‌های دیگر از خود نشان داده است. استفاده از توجه بصری نرم در مجموعه‌داده‌های Flickr30k و MS COCO عمل کرد بهتری نسبت به روش‌های دیگر از لحاظ معیار METEOR از خود نشان داده است. همین‌طور استفاده از توجه بصری سخت، بهترین عملکرد را در معیار BLEU از خود نشان داده است.

یکی از فعالیت‌های مفید برای بررسی نحوه عملکرد مدل که در این پژوهش مورد استفاده قرار گرفته است، بصری کردن فرایند تولید کلمه توسط مدل است. در این پژوهش، توجه بصری روی تصویر در هر مرحله به همراه کلمه تولید شده در هر مرحله مشخص شده‌اند که در درک نحوه عمل کرد مدل و همین‌طور پیدا کردن دلایل ایجاد کلمات غیر مرتبط بسیار کمک‌کننده هستند.

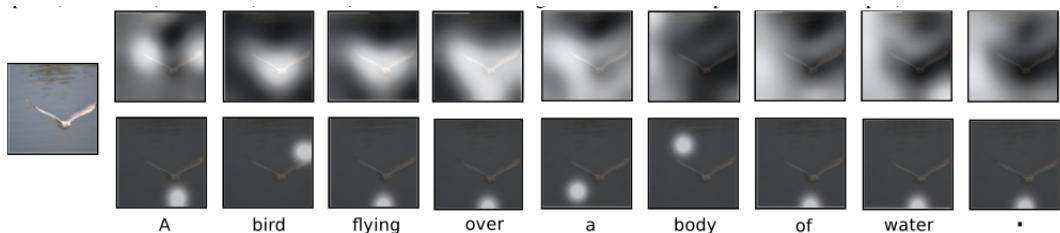
جدول ۲-۳: نتایج اعمال روش [۳۶] بر روی مجموعه‌داده‌های مختلف در مقایسه با روش‌های مختلف. [۳۶]

METEOR	BLEU-4	BLEU-3	BLEU-2	BLEU-1	نام مدل	مجموعه‌داده
-	-	۲۷.۰	۴۱.۰	۶۳.۰	Google NIC	Flickr8k
۱۷.۳۱	۱۷.۷	۲۷.۷	۴۲.۴	۶۵.۶	Log Bilinear	Flickr8k
۱۸.۹۳	۱۹.۵	۲۹.۹	۴۴.۸	۶۷.۰	Soft Attention	Flickr8k
۲۰.۴۰	۲۱.۳	۳۱.۴	۴۵.۷	۶۷.۰	Hard Attention	Flickr8k
-	۱۸.۳	۲۷.۷	۴۲.۳	۶۶.۳	Google NIC	Flickr30k
۱۶.۸۸	۱۷.۱	۲۵.۴	۳۸.۰	۶۰.۰	Log Bilinear	Flickr30k
۱۸.۴۹	۱۹.۱	۲۸.۸	۴۳.۴	۶۶.۷	Soft Attention	Flickr30k
۱۸.۴۶	۱۹.۹	۲۹.۶	۴۲.۹	۶۶.۹	Hard Attention	Flickr30k
۲۰.۴۱	-	-	-	-	CMU/MS Research	MS COCO
۲۰.۷۱	-	-	-	-	MS Research	MS COCO
-	۲۰.۳	۳۰.۴	۴۵.۱	۶۴.۲	BRNN	MS COCO
-	۲۴.۶	۳۲.۹	۴۶.۱	۶۶.۶	Google NIC	MS COCO
۲۰.۰۳	۲۴.۳	۳۴.۴	۴۸.۹	۷۰.۸	Log Bilinear	MS COCO
۲۲.۹۰	۲۴.۳	۳۴.۴	۴۹.۲	۷۰.۷	Soft Attention	MS COCO
۲۳.۰۴	۲۵.۰	۳۵.۷	۵۰.۴	۷۱.۸	Hard Attention	MS COCO

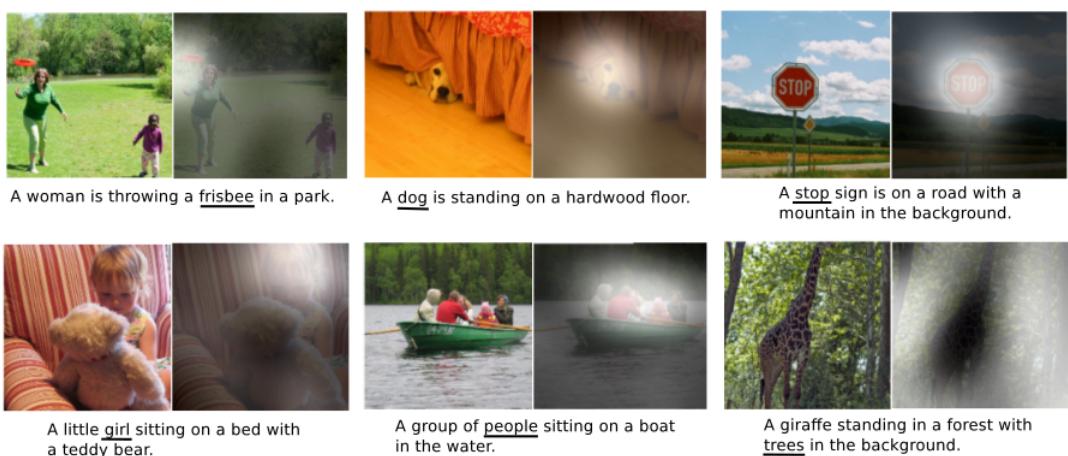
شکل ۲۴-۲ توجه بصری در هر زمان را برای تولید هر کلمه برای یک تصویر نمونه نمایش می‌دهد. ردیف بالا نمایش‌دهنده عمل کرد روش با استفاده از توجه بصری نرم و ردیف پایین نمایش‌دهنده عمل کرد روش با استفاده از توجه بصری سخت است. در این نمونه خاص، نتیجه تولید جمله برای هر دو روش یکسان بوده است.

در تمام تصاویر، محدوده‌های روش‌تر، محدوده‌هایی هستند که در آن‌ها ضریب میانگین‌گیری بیشتر بوده و توجه بیشتری در محاسبات روی آن‌ها متمرکز شده است. شکل ۲۵-۲ چند نمونه از تصاویر را نمایش می‌دهد که در آن‌ها توجه بصری روی یک جسم منجر به تولید کلمه دقیق متناظر آن جسم شده است. کلمه تولید شده در شرح نهایی تولید شده برای تصویر در زیر هر تصویر نمایش داده شده است.

به علاوه، شکل ۲۶-۲ نمایش‌دهنده شرایطی است که در آن کلمه تولید شده متناظر توجه بصری بصری نیست.



شکل ۲-۲: نحوه عمل کرد الگوریتم در تغییر توجه بصری بصری و کلمه تولید شده در هر نقطه. [۳۶]



شکل ۲-۳: چند نمونه از تصاویر که در آنها توجه بصری روی یک جسم منجر به تولید کلمه دقیق متناظر شده است. [۳۶]

با استفاده از بصری‌سازی محل توجه بصری و کلمه تولید شده در هر مرحله، می‌توان به راحتی مشاهده کرد که کلمه تولید شده متناظر کدام نقطه از تصویر، نامناسب است.

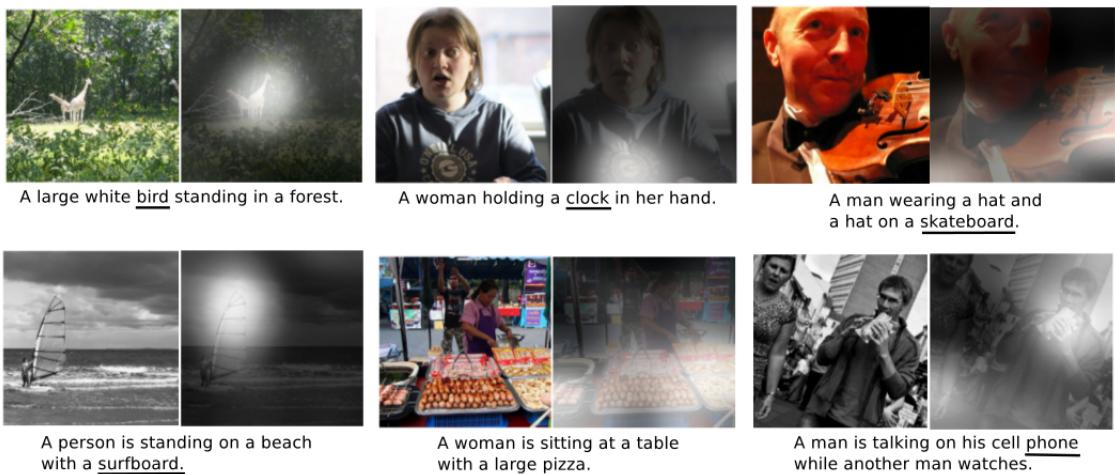
علاوه بر موارد فوق، نمونه‌ای از بررسی تمام مراحل تولید شرح متناظر صحنه برای یک تصویر را در حالت‌های استفاده از توجه بصری سخت در شکل ۲۷-۲ و توجه بصری نرم در شکل ۲۸-۲ قابل مشاهده است. هر کلمه تولید شده در هر مرحله در کنار میزان فعال‌سازی شبکه مربوط به آن کلمه نمایش داده شده است.

۳-۵-۲ فعالیت‌های مشابه دیگر

استفاده از توجه تولید شرح متناظر تصویر، از سال ۲۰۱۵، توجه بسیاری از پژوهش‌گران را به خود جلب نموده است و پژوهش‌های زیادی با استفاده از این ایده سعی در تولید جمله برای تصاویر، ویدئوها، صوت و انواع ورودی‌های مشابه نموده‌اند. همین‌طور در حوزه ترجمه ماشینی، نسخه‌های متفاوت و متنوعی از این ایده برای دست‌یابی به ترجمه‌های بهتر ارائه شده‌اند.

یکی از پژوهش‌هایی که در این زمینه برای بهبود عمل کرد ترجمه ماشینی با استفاده از نقطه توجه ارائه شده است، پژوهشی است که آقای منینگ و همکارانش در سال ۲۰۱۵ ارائه دادند [۲۴]. در این پژوهش، که بر روی مجموعه‌داده WMT که شامل جملات انگلیسی و معادل آلمانی آن‌ها است اجرا شده، از یک ساختار پشتیاهی مطابق شکل ۲۹-۲ استفاده شده است. در این ساختار، برای آموزش، جمله انگلیسی و معادل آلمانی آن به یکدیگر الصاق شده و ساختار رمزگذار-رمزگشای با هم آموزش می‌بینند. سپس با دریافت ورودی جمله انگلیسی یا آلمانی، معادل آن‌ها تولید می‌شود.

در پژوهش ارائه شده نیز مانند پژوهشی که آقای بنجیو در سال ۲۰۱۵ در حوزه ترجمه ماشینی انجام دادند،



شکل ۲: نمونه‌هایی از تولید کلمات نامناسب مطابق با نقاط توجه استفاده شده در مدل [۳۶]

دو نوع توجه محاسبه و مورد آزمایش قرار گرفته است. توجه اول که معادل توجه نرم است، تحت عنوان توجه سراسری^{۶۱} و توجه دوم که معادل توجه سخت است، تحت عنوان توجه ناحیه‌ای^{۶۲} مطرح شده‌اند. در این آزمایش هم مشابه نتایج پژوهش [۳۶] توجه ناحیه‌ای، در بسیاری موارد عمل کرد بهتری نسبت به توجه سراسری از خود نشان داده است.

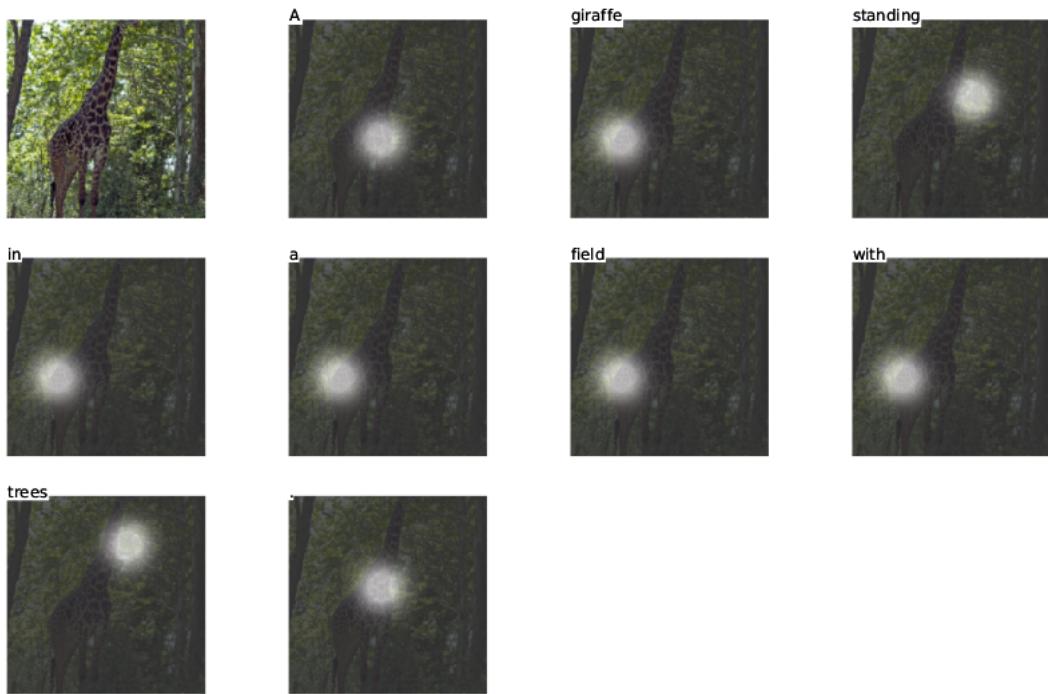
پژوهش مشابهی در حوزه پرسش و پاسخ بصری توسط ایده مشابه پژوهش [۲۴] در سال ۲۰۱۶ توسط آقای ینگ و همکارانش در [۳۷] ارائه شده است. در این پژوهش، بردار ویژگی تصویر به شرح متناظر آن که در مجموعه‌داده موجود است، الصاق شده و ساختار رمزگذار-رمزگشایی به شکل مشابهی آموزش می‌بینند. سپس با ورود یک تصویر جدید، بردار ویژگی آن به ساختار داده شده و جمله مرتبط با تصویر جدید توسط ساختار تولید می‌گردد. آقای بنجیو در پژوهش [۳] در سال ۲۰۱۵، چارچوب کاری‌ای را مبتنی بر استفاده از نقطه توجه ارائه کردند که قابل استفاده در حوزه‌های ترجمه ماشینی، تولید شرح متناظر تصویر، توصیف ویدئو و گفتار است. در این پژوهش، علاوه بر ارائه یک روش برای محاسبه نقطه توجه بصری و استخراج بردار ویژگی با استفاده از بردارهای حاشیه‌نویسی، صحبت‌هایی در مورد امکان انتقال یادگیری در چارچوب ارائه شده انجام شده است. در این پژوهش در مورد هر یک از چهار حوزه‌ای که ذکر شد، صحبت شده و نحوه استفاده از چارچوب کاری در هر یک از این حوزه‌ها تبیین شده است. همین‌طور در این پژوهش اثبات شده است که استفاده از مکانیزم نقطه توجه، این امکان را به مدل می‌دهد که به طور بدون نظارت، رابطه همترازی بین بخش‌های ورودی و خروجی را یاد بگیرد تا بتوان از این ویژگی در انتقال یادگیری استفاده نمود. نتایج استفاده از این چارچوب کاری در حوزه‌های مختلف بهتر از روش‌های دیگر گزارش شده است. شکل ۳۰-۲ ساختار چارچوب را در حوزه تولید شرح متناظر تصویر نمایش می‌دهد.

۶-۲ جمع‌بندی

در این بخش، پژوهش‌های ارائه شده پیشین در حوزه تولید خودکار شرح متناظر تصاویر را مرور نمودیم. مانند تمام مسائل دیگر موجود در حوزه هوش مصنوعی، بررسی روش‌های تولید خودکار شرح بر تصاویر را با بررسی عمل کرد

^{۶۱}Global Attention

^{۶۲}Local Attention



(a) A giraffe standing in a field with trees.

شکل ۲-۲: فرایند تولید شرح متناظر تصویر با استفاده از توجه بصری سخت [۳۶]

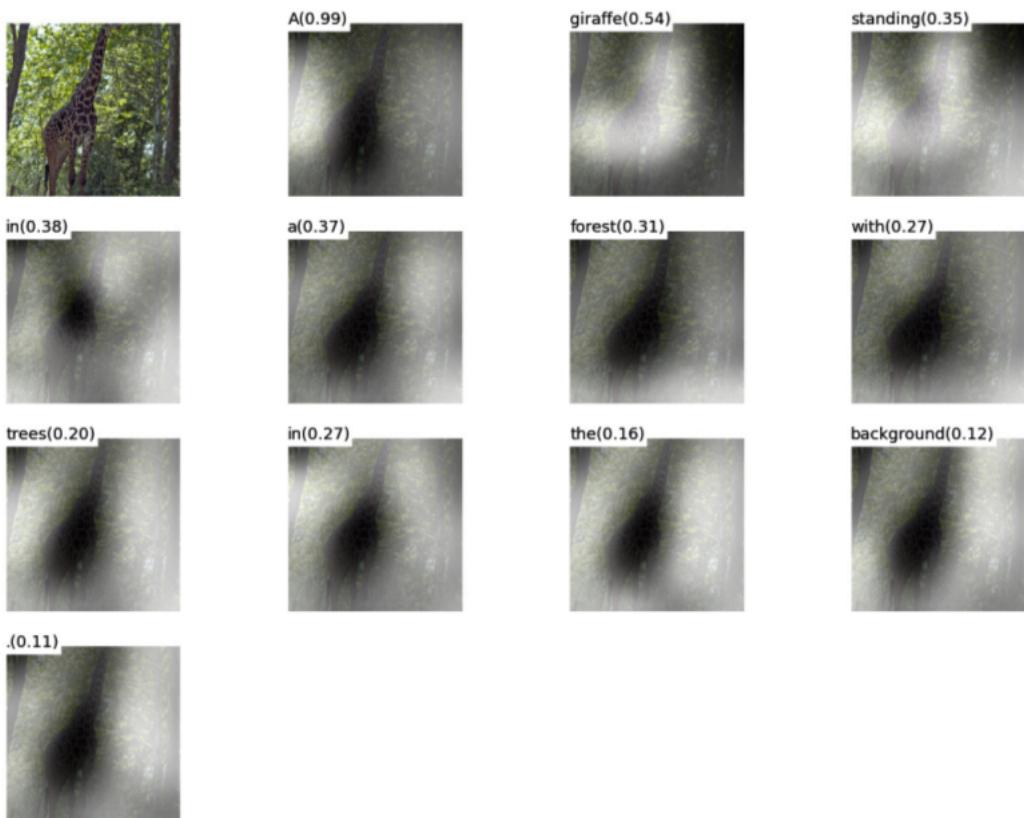
مغز انسان در درک و توصیف تصاویر، شروع کردیم.

مطابق با نتایج آزمایشات انجام شده روی مغز انسان، مغز ما قادر است در کمتر از ۲۰۰ تا ۵۰۰ میلی ثانیه، تمام اطلاعات مورد نیاز برای توصیف صحنه را از تصویر، استخراج نماید. در حالتی که اطلاعات مورد نیاز، به شکل ساختاربندی شده و در قالب فرم‌های پرسشنامه، جمع‌آوری شوند، قدرت بازیابی مغز به مراتب بیشتر از حالتی است که فرد ملزم به توصیف صحنه در چارچوب جملات باشد. با این مقدمه، می‌توان دریافت، فرایند درک صحنه، باید به مراتب سریع‌تر از فرایند تولید جمله، انجام شود.

استفاده از مدل‌های گرافی احتمالی، در سال‌های قبل از ۲۰۱۴، روش اصلی و اساسی در درک صحنه و استخراج اطلاعات مورد نیاز از تصاویر، به حساب می‌آمد. مدل‌های استاندارد مختلفی در بین پژوهش‌های انجام شده در این حوزه به چشم می‌خورد. در این بین، نمونه‌هایی از مدل‌های میدان تصادفی مارکف، میدان تصادفی شرطی و مدل‌های مولد خاص منظوره طراحی شده توسط پژوهش‌گران را مورد بررسی قرار دادیم.

از اواخر سال ۲۰۱۳، روش‌های مبتنی یادگیری عمیق، نظرسیاری از پژوهش‌گرانی را که در حوزه تولید شرح متناظر تصویر فعالیت می‌کردند، به خود جلب نمودند. این دسته از روش‌ها، به دلیل عمل کرد بهتری که از خود نشان دادند، توانستند جایگزین روش‌های گرافی احتمالاتی شوند.

از جمله پژوهش‌هایی که با استفاده از شبکه‌های عصبی عمیق اقدام به تولید شرح متناظر تصویر کردند، می‌توان به پژوهش خانم لی و همکارانش [۱۳] در سال ۲۰۱۵ اشاره کرد. در مرحله آموزش این پژوهش، ابتدا با استفاده از روش شبکه عصبی کانولوشنی ناحیه‌ای که در بخش قبل، ارائه شد، نواحی تصویر که شامل تصویر یک جسم هستند، انتخاب شده و بردار ویژگی مربوط به هر کدام از این بخش‌ها، استخراج می‌شود.



(b) A giraffe standing in a forest with trees in the background.

شکل ۲۸-۲: فرایند تولید شرح متناظر تصویر با استفاده از توجه بصری نرم [۳۶]

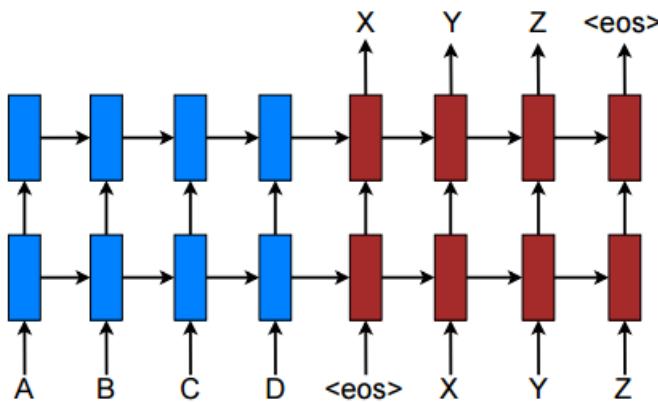
پس از این مرحله، بردار ویژگی مربوط به جملات موجود در مجموعه‌داده، توسط یک شبکه عصبی بازگشتی دوطرفه، استخراج می‌شود. برای این کار، ابتدا بردار ویژگی مربوط به هر کلمه با استفاده از یک شبکه کلمه به بردار Word To Vec، استخراج شده و به عنوان ورودی به شبکه بازگشتی دوطرفه داده می‌شوند. استفاده از شبکه بازگشتی دوطرفه این امکان را می‌دهد که تاثیر کلمات قبل و بعد از هر کلمه، در تولید بردار ویژگی جملات لحاظ شود.

با بهینه‌سازی یکتابع انرژی روی این قسمت، شبکه عصبی بازگشتی دوطرفه و شبکه عصبی کانولوشنی با هم آموزش داده می‌شوند. از این طریق، بخش‌هایی از مدل که مربوط به تولید بردار ویژگی از جملات و استخراج نواحی تصاویر و بردار ویژگی مربوط به آن‌ها است، به طور کامل آموزش می‌بینند.

در ادامه فرایند آموزش شبکه، با ارائه بردار ویژگی تولید شده توسط شبکه عصبی کانولوشنی آموزش دیده در بخش قبلی به یک شبکه عصبی بازگشتی دیگر، و ارائه جملات موجود در مجموعه‌داده به آن، شبکه عصبی بازگشتی را برای تولید جمله نهایی آموزش می‌دهیم.

آزمایشات انجام شده روی این پژوهش، معیار BLEU حاصل توسط روش را روی مجموعه‌داده MS COCO در مقایسه با روش‌های دیگر ارزیابی کردند. در این آزمایشات، بهترین عمل کرد روش ارائه شده روی این مجموعه‌داده به امتیاز BLEU برابر با ۵۷.۳ رسیده است و این در حالیست که روش [۲۵] روی همان مجموعه‌داده به مقدار ۵۵.۰ رسیده است.

یکی دیگر از روش‌های ارائه شده در این بخش، روشی است که در پژوهش [۲] در سال ۲۰۱۵ ارائه شده است. در



شکل ۲-۲: ساختار پشتهای ارائه شده در [۲۴]

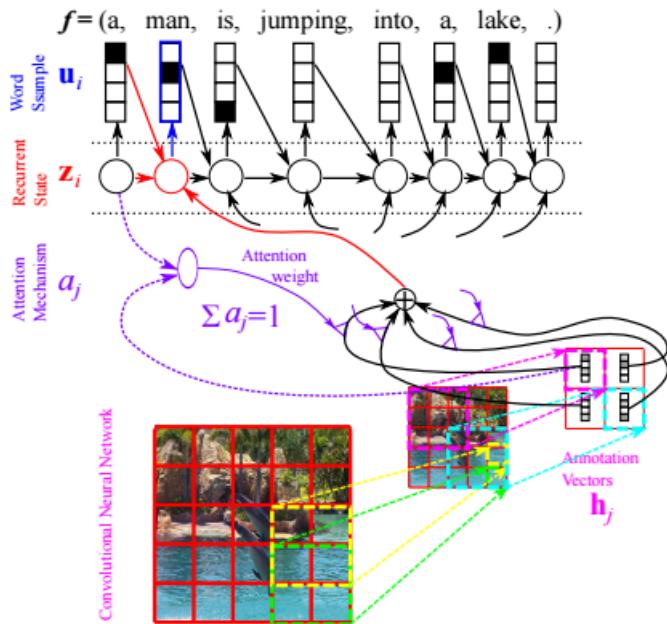
این روش، یک شبکه عصبی بازگشتی دوطرفه برای نگاشت جملات و تصاویر به یکدیگر استفاده شده است. مدل ارائه شده، قادر است با گرفتن تصویر به عنوان ورودی، شرح متناظر آن را در قالب یک جمله تولید و با گرفتن یک جمله به عنوان ورودی، تصویر مربوط به آن را بازیابی نماید. در این روش با در نظر گرفتن واحد عصبی ارائه شده در پژوهش [۲۷] و اضافه کردن دو متغیر دیگر به آن، مدلنهایی تولید شده است. متغیرهای اضافه شده به این مدل، شامل متغیری برای بردار ویژگی تصویر و متغیر دیگر برای تفسیر بصری آخرین کلمه دیده شده، است.

شبکه عصبی ارائه شده در این پژوهش، توزیع احتمال توازن تصاویر و جملات را مدل‌سازی می‌نماید. در صورتی که جمله به عنوان ورودی داده شده باشد، توزیع احتمال تصویر به شرط جمله قابل محاسبه و تصویر مربوطه قابل بازیابی است. در صورتی که تصویر به عنوان ورودی داده شده باشد، توزیع احتمال جمله به شرط تصویر قابل محاسبه است.

نتایج ارائه شده در این پژوهش، با روش‌های دیگر مقایسه شد. برای تولید جمله به شرط داشتن تصویر، میزان امتیاز BLEU حاصل توسط مدل در بهترین حالت برای مجموعه داده Flickr8k مقدار ۱۳.۱، برای مجموعه داده Flickr30k مقدار ۱۲.۰ و برای مجموعه داده MS COCO مقدار ۱۸.۸ بوده است. این در حالیست که نتایج حاصل برای مدل RNN + VGG به ترتیب برابر با ۱۲.۴، ۲۰.۶، ۱۸.۹ و ۱۹.۲ بوده و مقادیر به دست آمده برای جملاتی که توسط عوامل انسانی تولید شده‌اند به ترتیب برابر با ۱۱.۹ و ۱۸.۴ بوده و مقادیر به دست آمده برای جملاتی ارائه شده در حوزه تولید شرح متناظر تصاویر از روش‌های استاندارد دیگر بهتر بوده اما هنوز به جملات تولید شده توسط انسان نمی‌رسد.

همین‌طور برای بازیابی تصاویر با داشتن جمله ورودی، نتایج حاصل توسط مدل برای مجموعه داده Flickr30k به ترتیب برای معیارهای R@10، R@5 و Med r 500 به ترتیب ۵۸.۱، ۴۵.۷ و ۷ است. این در حالیست که نتایج حاصل توسط مدل RNN + VGG به ترتیب برای با ۵۴.۱، ۴۱.۱، ۱۵.۱ و ۹ است.

چارچوب کاری رمزگذار-رمزگشا یکی از اصلی‌ترین چارچوب‌های کاری در حوزه ترجمه ماشینی و پیرو آن تولید شرح متناظر تصویر به شمار می‌رود. رمزگذار در این چارچوب کاری وظیفه نگاشت ورودی به فضای معنا و رمزگشا وظیفه نگاشت فضای معنا به فضای خروجی را بر عهده دارد. در حوزه ترجمه ماشینی معمولاً از یک شبکه عصبی حافظه کوتاه‌مدت بلند به عنوان رمزگشا استفاده می‌شود. این شبکه عصبی با دریافت کلمات جمله



شکل ۲-۳۰: ساختار چارچوب کاری ارائه شده در [۲۶] در حوزه تولید شرح متناظر تصویر

ورودی به ترتیب، بردار حالت مخفی خود را به روزرسانی می‌نماید. در نهایت می‌توان از این بردار به عنوان بردار حاصل نگاشت جمله ورودی به فضای معنا استفاده نمود. رمزگشا در این چارچوب کاری با دریافت بردار ویژگی تولید شده توسط رمزگشا، عمل تولید خروجی را بر عهده خواهد داشت. در حوزه ترجمه ماشینی معمولاً یک شبکه عصبی بازگشتی برای رمزگشا می‌تواند مورد استفاده قرار بگیرد. به طور معمول، بردار ویژگی تولید شده توسط رمزگزار، به عنوان یک ورودی به رمزگشا داده می‌شود و رمزگشا در هر مرحله با تولید یک کلمه به عنوان خروجی، بردار حالت مخفی خود را به روزرسانی نموده و با استفاده از بردار حالت مخفی جدید، اقدام به تولید کلمه جدید می‌نماید.

یکی از محدودیت‌های جدی فرایند مذکور این است که بردار ویژگی فقط یک بردار با طول ثابت است و اولاً رمزگزار باید بتواند تمام اطلاعات قابل استخراج را تنها در این بردار جاسازی نماید و ثانیاً رمزگشا باید بتواند تمام اطلاعات مورد نیاز خود برای تولید کلمه و جمله را فقط از همین یک بردار استخراج نماید. این مشکل، پژوهش‌گران را بر آن داشت تا بردار ویژگی را از یک بردار با طول ثابت به یک دنباله بردار با طول ثابت و تعداد متغیر تغییر دهنده. به بردارهای ویژگی تولید شده در حالت جدید، حاشیه‌نویسی‌ها باید دارای دو شرط زیر باشند:

۱. در برگیرنده تمام معنای ورودی باشند.
۲. تمرکز بیشتری روی معنای یک بخش مشخص از ورودی داشته باشند.

با در نظر گرفتن این ویژگی‌ها، رمزگشا قادر خواهد بود تا هنگام تولید هر کلمه، روی معنای یک بخش از جمله تمرکز بیشتری داشته باشد و فقط از آن بخش برای تولید کلمه استفاده نماید. به این شکل، کلمات تولید شده شباهت بیشتری به ورودی خواهند داشت و ترجمه‌های بهتری حاصل خواهد شد.

در سال ۲۰۱۵، آقای بنجیو و همکارانش در پژوهش [۲۶] روشی ارائه دادند که در آن برای اولین بار از ایده استفاده از نقطه توجه در حوزه ترجمه ماشینی برای تولید شرح متناظر تصویر استفاده نمودند. در این پژوهش، از یک

شبکه عصبی کانولوشنی به عنوان رمزگذار استفاده شده است. خروجی شبکه از لایه ماقبل آخر گرفته شده که منجر به ایجاد تعداد زیادی بردار ویژگی از تصویر می‌شود که هر کدام از این بردارهای ویژگی، از یک ناحیه از تصویر ایجاد شده‌اند و تمرکز بیشتری روی آن ناحیه داشته‌اند.

بدین ترتیب با استفاده از یک شبکه عصبی بازگشتی به عنوان رمزگشا و استفاده از بردارهای حاشیه‌نویسی ایجاد شده توسط رمزگذار می‌توان به راحتی عملیات تولید شرح متناظر تصویر را انجام داد. تنها نکته‌ای که باید مشخص شود، چگونگی استفاده از بردارهای حاشیه‌نویسی است. در این پژوهش دو روش مختلف برای استفاده از بردارهای حاشیه‌نویسی مطرح شده است.

روش اول موسوم به روش توجه سخت، روشی است که در آن فقط یک بردار حاشیه‌نویسی انتخاب شده و از آن برای تولید جمله استفاده می‌شود. در این روش به هر یک از بردارهای حاشیه‌نویسی توسط یک مدل که قبلاً آموزش دیده است، یک وزن اختصاص می‌دهیم و سپس با توجه به وزن‌های تخصیص داده شده به هر بردار حاشیه‌نویسی، یکی از آن‌ها را به عنوان بردار ویژگی تصویر انتخاب کرده و از آن در مراحل بعدی استفاده می‌کنیم.

روش دوم موسوم به روش توجه نرم، روشی است که در آن یک بردار ویژگی کلی از روی بردارهای حاشیه‌نویسی تولید شده و از آن بردار در مراحل بعدی استفاده می‌شود. برای تولید این بردار نیز مانند روش توجه سخت، ابتدا توسط یک مدل که از پیش‌آموزش دیده است، به هر یک از بردارهای حاشیه‌نویسی یک وزن اختصاص می‌دهیم. سپس می‌توان با محاسبه امید ریاضی بردارهای حاشیه‌نویسی با توجه به وزن هر یک از آن‌ها بردار ویژگی نهایی را برای تصویر تولید و از آن برای تولید جمله استفاده کرد.

آزمایشات انجام شده روی این مدل نشان می‌دهد، معیار BLEU-1 حاصل از این روش با استفاده از توجه سخت معمولاً از مدل توجه نرم بیشتر بوده است. مطابق با نتایج گزارش شده در این پژوهش، میزان امتیاز BLEU-1 حاصل توسط توجه سخت روی مجموعه‌داده‌های Flickr8k، Flickr30k، MS COCO و ۶۶.۹ و ۶۷.۰ است. این در حالیست که امتیاز حاصل توسط توجه نرم روی همین مجموعه‌های داده، به ترتیب برابر با ۶۶.۷ و ۶۷.۰ و امتیاز کسب شده توسط مدل Log Bilinear در بهترین حالت، به ترتیب برابر با ۶۰.۰ و ۶۵.۶ بوده است.

مطابق با آزمایشات انجام شده، استفاده از توجه نرم، معیار METEOR را نسبت به استفاده از توجه سخت افزایش می‌دهد. طبق نتایج گزارش شده در پژوهش، امتیاز METEOR حاصل از توجه نرم به ترتیب روی مجموعه‌داده‌های Flickr30k، Flickr8k و MS COCO برابر با ۱۸.۹۳، ۱۸.۴۹ و ۲۳.۹۰ بوده است. این در حالیست که امتیاز Log Bilinear در بهترین حالت به ترتیب برابر با ۲۰.۳۰، ۱۸.۴۶ و ۲۳.۰۴ و امتیاز کسب شده توسط روش Log Bilinear این‌که جملات تولید شده توسط روش توجه سخت با در نظر گرفتن جملات موجود در مجموعه‌داده از امتیاز بالاتری نسبت به جملات تولید شده توسط توجه نرم برخوردارند؛ استفاده از توجه نرم، منجر به تولید جملات قابل قبول‌تری توسط انسان می‌شود.

پژوهش‌های مختلفی از این ایده در حوزه‌های مختلف استفاده نموده‌اند که گزارش مختصراً از تعدادی از این پژوهش‌ها ارائه شده است.

فصل سوم

روش ارائه شده در پژوهش

۱-۳ مقدمه

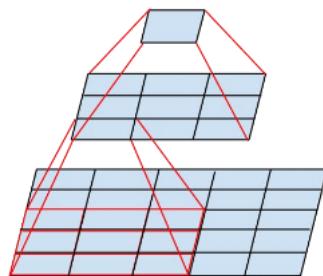
در فصل‌های گذشته، سیر پژوهش‌های مرتبط در حوزه تولید خودکار شرح بر تصاویر را مورد بررسی قرار دادیم. در حال حاضر، تمامی پژوهش‌های انجام شده در حوزه تولید خودکار شرح بر تصاویر، با استفاده از شبکه‌های عصبی و یادگیری عمیق و با بهره‌گیری از چارچوب کاری رمزگذار-رمزگشا انجام می‌شوند. در پژوهش جاری، ما نیز از همین رویکرد استفاده می‌نماییم.

در این فصل، ابتدا به بیان چارچوب کاری رمزگذار-رمزگشا مورد استفاده در پژوهش، پرداخته و سپس مشکلات موجود در این چارچوب کاری را بیان می‌نماییم. سپس در ادامه، به ارائه روش پیشنهادی برای حل این مشکلات و بررسی جوانب مختلف آن می‌پردازیم.

۲-۳ رمزگذار

رمزگذار مورد استفاده در چارچوب کاری رمزگذار-رمزگشا در حوزه تولید خودکار شرح بر تصاویر، یک شبکه عصبی کانولوشنی عمیق است که به ازای هر تصویر ورودی، یک بردار ویژگی تولید می‌نماید. در پژوهش‌های مختلف، از شبکه‌های عصبی کانولوشنی مختلفی به این منظور استفاده می‌شود. عموماً استفاده از این شبکه‌ها به طور از پیش آموزش دیده، اتفاق می‌افتد. در این حالت، شبکه‌های عصبی کانولوشنی را ابتدا روی یک مجموعه داده معتبر، که معمولاً مجموعه داده ImageNet است، آموزش می‌دهند. سپس از شبکه آموزش دیده، برای استخراج ویژگی از تصاویر، استفاده می‌نمایند. در پژوهش جاری، ما نیز از مدل Google Inception V3 که در پژوهش [۳۳] توسط سیگدی و همکارانش در سال ۲۰۱۶ ارائه شده است، به عنوان رمزگذار استفاده کرده‌ایم.

یکی از اساسی‌ترین تغییراتی که در این مدل نسبت به مدل‌های گذشته خود ارائه شده است، جایگزینی فیلترهای کانولوشن $5 * 5$ با یک شبکه عصبی کوچک با دو فیلتر $3 * 3$ است. شکل ۱-۳ نمونه‌ای از این شبکه عصبی را نمایش می‌دهد. همان‌طور که در شکل مشخص است، عملی که این شبکه عصبی کوچک انجام می‌دهد، دقیقاً معادل عملیاتی است که کانوالو کردن یک فیلتر $5 * 5$ روی یک تصویر انجام می‌دهد. این جایگزینی با کاهش تعداد پارامترهای مورد نیاز، سرعت یادگیری و عمل کرد شبکه را بهبود می‌بخشد.

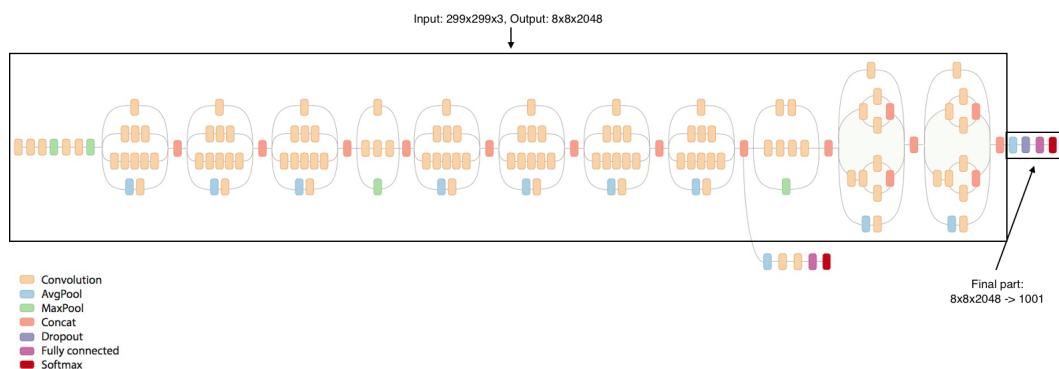


شکل ۱-۳: شبکه عصبی کوچک جایگزین فیلتر $5 * 5$ ارائه شده در پژوهش [۳۳]

به همین ترتیب می‌توان فیلترهای $n * n$ را به فیلترهای $1 * 1$ و $1 * n$ تبدیل کرد. این کار، می‌تواند بهبود قابل ملاحظه‌ای در عمل کرد شبکه ایجاد نماید.

ساختمار کلی رمزگذار مورد استفاده در این پژوهش، در شکل ۲-۳ ارائه شده است. همان‌طور که در این شکل قابل مشاهده است، ورودی‌های این شبکه، تصاویر رنگی به ابعاد ۲۹۹ در ۲۹۹ پیکسل هستند. با توجه به این که تصاویر موجود در مجموعه داده مورد استفاده در این پژوهش، محدودیت ابعاد ندارند، ابعاد تمام تصاویر را به اندازه‌ای

کاهش می‌دهیم که در یک قاب 299×299 پیکسل، جای بگیرند. سپس قسمت‌های اضافی قاب را با پیکسل‌های صفر می‌پوشانیم و تصویر آماده شده را به شبکه می‌دهیم.



شکل ۳-۲: ساختار کلی رمزگذار مورد استفاده در پژوهش [۳۳]

همان‌طور که در شکل ۲-۳ قابل مشاهده است، بخش پایانی رمزگذار شامل لایه‌هایی است که 8×8 بردار ویژگی 2048 بعدی را گرفته، بردارهای ویژگی مورد استفاده برای انتقال یادگیری^۱، و یک بردار ویژگی نهایی 1008 بعدی را خروجی می‌دهد. بردار ویژگی حاصل، با اعمال یک لایه تماماً متصل^۲ روی خروجی، تولید شده است. این لایه به منظور آموزش شبکه عصبی برای دسته‌بندی اجسام موجود در تصاویر، ایجاد شده است. از آن‌جا که دسته‌بندی اجسام، برای تولید خودکار شرح بر تصاویر، به تنهایی کافی نیست، در این پژوهش به جای استفاده از بردار ویژگی لایه آخر، از 8×8 بردار ویژگی 2048 بعدی مربوط به انتقال یادگیری، استفاده می‌نماییم. هر کدام از این بردارهای ویژگی استخراج شده، بیان‌کننده اطلاعات کل تصویر است. با این حال، تمرکز هر یک از این بردارها به بخشی از تصویر اولیه معطوف شده است. با این تفاسیر، این بردارها می‌توانند به عنوان بردارهای حاشیه‌نویسی مناسب مورد استفاده قرار بگیرند.

با توجه به این نکته که، مدل از پیش آموزش دیده این شبکه عصبی، که روی مجموعه‌داده ImageNet آموزش داده شده است، در دسترس و قابل استفاده برای پژوهش گران دیگر است و نتایج اعمال این مدل بر مجموعه‌داده مذکور، بهتر از تمامی مدل‌های قبلی ارائه شده برای دسته‌بندی تصاویر بوده است، در پژوهش حاضر، از نسخه در دسترس این شبکه، بدون آموزش مجدد برای تولید خودکار شرح بر تصاویر به ترتیبی که توضیح داده شد، استفاده می‌شود. جدول ۱-۳ نتایج حاصل از این شبکه عصبی را با بهترین نتایج حاصل از روش‌های قبلی مورد مقایسه قرار داده است.

جدول ۱-۳: مقایسه عملکرد رمزگذار مورد استفاده در پژوهش با مدل‌های دیگر [۳۳]

نام روش	خطای محتمل ترین دسته ^۴	خطای محتمل ترین دسته ^۵
[۳۲] GoogLeNet	-	۶.۶۷٪
[۱۱] PReLU	۲۰.۱٪	۴.۹٪
Inception-V3	۱۷.۲٪	۳.۵۸٪

^۱Transfer Learning

^۲Fully Connected Layer

۳-۳ رمزگشا

در این بخش به بررسی مدل ارائه شده به عنوان رمزگشا و فرایند پیشنهادی تولید جمله می‌پردازیم. در ادامه این قسمت از گزارش، ابتدا مدل استاندارد رمزگشا را مورد بررسی قرار می‌دهیم. سپس با بیان ایده اصلی پژوهش در ارتقا کیفیت جملات تولید شده و بررسی جزئیات آن، نحوه تغییر ساختار رمزگشا، تابع هزینه و متدهای آموزش را بررسی خواهیم نمود.

۱-۳-۱ معماری رمزگشا

وظیفه واحد رمزگشا در چارچوب کاری رمزگذار-رمزگشا این است که با دریافت بردارهای حاشیه‌نویسی تصویر، جملات توصیف‌کننده مربوط به تصویر را تولید نماید. هر جمله را می‌توان به عنوان دنباله‌ای با طول متغیر از کلمات در نظر گرفت که هر کلمه در آن، با یک بردار ویژگی توصیف شده است. رابطه (۱-۳) نمایش یک جمله در مدل ریاضی مورد استفاده در این پژوهش را بیان می‌کند که در آن، L تعداد کلمات موجود در جمله و x_i بردار ویژگی کلمه i ام در جمله است. لازم به ذکر است، در ادامه این بخش، جملات و کلمات ورودی رمزگشا را با X و جملات و کلمات خروجی را با Y نمایش می‌دهیم.

$$X = \{x_1, x_2, \dots, x_L\} \quad (1-3)$$

فرایند یادگیری رمزگشا، باید به نحوی انجام شود که رمزگشا بتواند شرح متناظر هر تصویر را برای تصاویر و شرح موجود در مجموعه‌داده آموزشی، تولید نماید. برای این منظور، با فرض این‌که بردارهای حاشیه‌نویسی استخراج شده توسط رمزگذار برای تصویر I با $\{\theta_0, \theta_1, \dots, \theta_n\} = \Theta$ و جمله مطلوب در مجموعه‌داده برای تصویر I را با $\{s_1, s_2, \dots, s_{L_I}\} = S$ نمایش داده شده باشند، رمزگشا باید بتواند در هر مرحله، احتمال کلمه بعدی را با توجه به کلمات تولید شده قبلی و مجموعه بردارهای حاشیه‌نویسی تولید نماید. به این منظور، باید یک مساله بهینه‌سازی را تعریف نموده و در صدد حل آن باشیم. رابطه (۲-۳) مساله بهینه‌سازی اولیه‌ای را نمایش می‌دهد که با حل آن می‌توان رمزگشای مورد نظر را تولید کرد. در این رابطه Ξ مجموعه تمام متغیرهای قابل آموزش است.

$$\begin{aligned} & \underset{\Xi}{\text{maximize}} \quad \Pr(y_{t+1} | y_t, y_{t-1}, \dots, y_0, \Theta) \\ & \text{s.t.} \quad y_{t+1} = s_{t+1} \end{aligned} \quad (2-3)$$

مطابق با مساله بهینه‌سازی تعریف شده، به دنبال مدلی هستیم که در هر مرحله بتواند یک تابع احتمال روی تمام کلمات موجود در دیکشنری کلمات ایجاد نماید. این تابع احتمال، یک تابع احتمال شرطی، به شرط داشتن کلمات تولید شده قبلی و مجموعه بردارهای حاشیه‌نویسی استخراج شده توسط رمزگذار است. در این تابع احتمال باید تا جای ممکن، احتمال کلمه بعدی موجود در شرح متناظر تصویر، زیاد و احتمال بقیه کلمات، کم باشد. پس باید تمام پارامترهای قابل آموزش را طوری مقداردهی نماییم، که کلمه $t+1$ در شرح متناظر تصویر، بیشترین احتمال را بین کلمات دیکشنری در مرحله t داشته باشد.

واحد اساسی سازنده رمزگشای مورد استفاده در این پژوهش، شبکه عصبی LSTM است. روابط (۳-۳) تا (۷-۳)، ساختار داخلی یک واحد LSTM را نمایش میدهد. در این روابط، x بردار ورودی، h بردار حالت شبکه، f

گیت فراموشی، i_t بردار فعالیت ورودی، o_t بردار خروجی، c_t بردار وضعیت سلول و t اندیس زمان هستند. همین طور σ_g تابع سیگموئید و σ_c تابع تانژانت هایپربولیک را نمایش می‌دهند. به علاوه، بردارهای W و U به ترتیب بردارهای وزن ورودی و حالت شبکه و بردارهای b بردارهای بایاس هستند. نماد \odot بیان‌گر ضرب مولفه‌های نظیر به نظیر، ضرب هادامارد^۵ است.

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \quad (3-3)$$

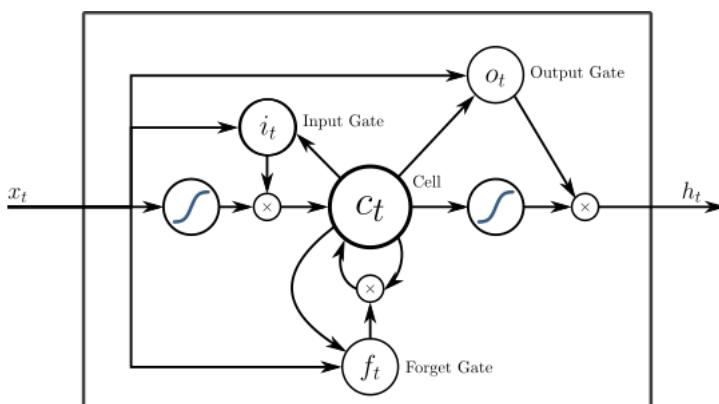
$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \quad (4-3)$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \quad (5-3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \quad (6-3)$$

$$h_t = o_t \odot c_t \quad (7-3)$$

شکل ۳-۳، نمایی از یک سلول شبکه LSTM را نمایش می‌دهد. در پروژه حاضر، از این ساختار به عنوان واحدهای اصلی شبکه رمزگشای استفاده می‌نماییم.



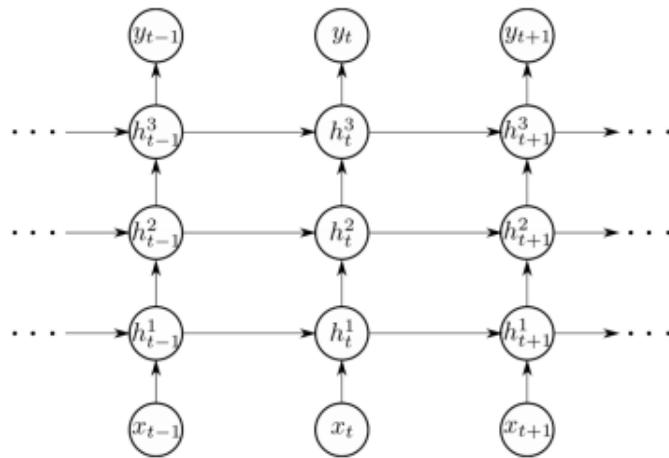
شکل ۳-۳: نمایی از یک سلول شبکه LSTM

به منظور افزایش قدرت شبکه، واحدهای LSTM را به صورت پشت‌های روی هم چیده و از مجموعه واحدهای موجود در یک پشت، به عنوان یک واحد عملیاتی در یک مرحله زمانی استفاده می‌نماییم. شکل ۴-۳ طرح‌واره‌ای از ساختار رمزگشای پشت‌های را نمایش می‌دهد.

نکته قابل توجه این است که بردارهای تولیدی y_t در این ساختار، همگی یک توزیع احتمال روی لغات دیکشنری را نمایش می‌دهند. اگر دیکشنری شامل تمام کلمات موجود در همه توصیفات موجود در مجموعه داده، شامل Γ لغت باشد، همه بردارهای y_t دارای Γ مولفه هستند که هر مولفه i از آن، احتمال رخداد کلمه #am در دیکشنری را در محل i از جمله ورودی مشخص می‌کند.

برای تولید جملات، کافیست در هر مرحله، با توجه به توزیع احتمال مشخص شده توسط بردار خروجی شبکه در همان مرحله، یک کلمه را از دیکشنری کلمات استخراج نموده و در جمله قرار دهیم.

^۵Hadamard Multiplication



شکل ۴-۳: طرح‌واره‌ای از ساختار LSTM پشت‌های

ارائه ایده اصلی برای بهبود روش

مطابق با معماری استاندارد مورد استفاده برای رمزگشا در چارچوب کاری رمزگذار-رمزگشا و با توجه به مساله بهینه‌سازی (۲-۳) خروجی شبکه بردار توزیع احتمال روی تمام کلمات دیکشنری در هر مرحله است. خطای مدل در زمان آموزش با مقایسه بردار خروجی شبکه و بردار مطلوب، که با توجه به شرح مربوط به تصویر در مجموعه‌داده ایجاد شده است، محاسبه می‌شود. این بردار مطلوب به صورت تک‌فعال^۶ تولید می‌شود؛ به این معنی که برای تولید بردار مطلوب در مرحله t ام، کلمه t ام موجود در شرح تصویر را انتخاب کرده، اندیس آن کلمه را در دیکشنری کلمات پیدا می‌نماییم. سپس یک بردار صفر به طول Γ ایجاد نموده و مولفه‌ای را که اندیس آن در بردار، با اندیس کلمه در دیکشنری برابر است، با عدد یک مقداردهی می‌نماییم.

در این مدل آموزش، شبکه سعی در یادگیری تولید دقیق جملات موجود می‌نماید و از آن‌جا که استفاده از معانی کلمات در هیچ‌جایی از این فرایند دیده نشده است، قراردادن کلمات هم‌معنی به جای یک‌دیگر در جملات تولید شده، که باعث ایجاد جملات گوناگون و عمل کرد بهتر شبکه می‌شود، در این مدل، امکان پذیر نیست. علاوه بر مورد فوق، با مدل‌سازی توزیع احتمال روی تمام دیکشنری، ابعاد مورد انتظار در خروجی شبکه بسیار زیاد شده و تعداد پارامترهای قابل آموزش، به بیش از ۱۰۰ میلیون پارامتر می‌رسد. تعداد زیاد پارامترهای قابل آموزش، یادگیری شبکه را دچار مشکل کرده و علاوه بر افزایش زمان یادگیری و استفاده از شبکه، باعث حساسیت بیشتر مدل به فوق‌پارامترهای^۷ الگوریتم‌های یادگیری، مانند نرخ یادگیری، می‌شود.

در این پژوهش، مدل‌سازی جاسازی کلمات به جای مدل توزیع احتمال آن‌ها، برای بهبود معماری فعلی، پیشنهاد می‌شود. معماری فوق را می‌توان به گونه‌ای تغییر داد که شبکه به جای مدل‌سازی توزیع احتمال شرطی کلمات، جاسازی آن‌ها را مدل نماید. جاسازی کلمات در ابعاد به مراتب کمتری از ابعاد دیکشنری قابل انجام است. با جای‌گزینی بردار جاسازی به جای بردار تک‌فعال کلمات، ابعاد خروجی کاهش یافته و پیرو آن، تعداد پارامترهای قابل آموزش شبکه به شکل چشم‌گیری کاهش پیدا می‌کنند. کاهش تعداد پارامترهای قابل آموزش به طور مستقیم، سرعت یادگیری را افزایش داده و از استفاده از بردارهای تنک در فرایند آموزش، جلوگیری می‌نماید. بردار جاسازی کلمات، نماینده معنای کلمات است. استفاده از بردار جاسازی کلمات در لایه خروجی، علاوه بر مزایای فوق، شبکه را قادر می‌سازد تا از اطلاعات معنایی لغات استفاده کرده و بتواند کلمات هم‌معنی با کلمات

⁶One Hot Vector⁷Hyperparameters

موجود در مجموعه‌داده را نیز به جای آن‌ها در جملات قرار دهد. این کار علاوه بر سهولت در یادگیری شبکه، امکان ایجاد جملات جدید را بیشتر کرده و عمل کرد شبکه در تولید جمله را بهبود می‌بخشد. به منظور جایگزینی جاسازی کلمات به جای بردار تک‌فعال، باید اولاً مساله بهینه‌سازی مطرح را تغییر داده و سپس روش مناسب برای بهینه‌سازی را انتخاب نماییم. علاوه بر این، لازم است، مدلی برای تولید جاسازی مناسب برای کلمات ایجاد کرده و آن را به مدل فعلی اضافه نماییم. در ادامه به بررسی جزئیات مربوطه می‌پردازیم.

۲-۳-۳ جاسازی کلمات

مدل جاسازی استفاده شده در این پژوهش، مطابق با مدل ارائه شده در پژوهش [۲۸] است که توسط آقای میکولوف و همکارانش در سال ۲۰۱۳ ارائه شده است. ویژگی برجسته این پژوهش این است که از مدل Skip-Gram، که در پژوهش [۲۶] ارائه شده است، استفاده می‌نماید.

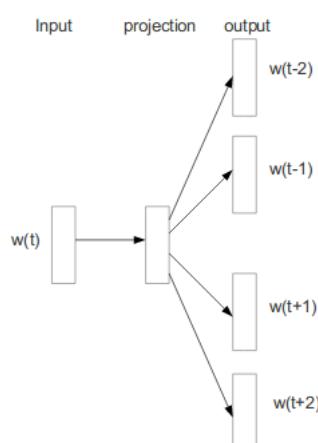
نکته مطلوب در مدل Skip-Gram این است که بردار جاسازی مربوط به کلمات، طوری تعیین می‌شود که احتمال تشخیص صحیح کلمات پیرامون کلمه جاری، بیشینه شود. به عبارت بهتر، با داشتن کلمه جاری، به راحتی بتوان کلمات قبل و بعد از آن را مشخص نمود. رابطه (۸-۳)تابع هدف این مدل را نمایش می‌دهد. در این رابطه، x_t کلمه فعلی را نمایش می‌دهد. مدل مذکور به نحوی آموزش می‌بیند که بتواند رابطه (۸-۳) را بیشینه کند.

$$\frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log Pr(x_{t+j} | x_t) \quad (8-3)$$

تابع احتمال شرطی به کار رفته در رابطه (۸-۳) مطابق با رابطه (۹-۳) تعریف می‌شود که در آن، ν_{x_o} و ν_{x_i} به ترتیب، بردارهای جاسازی x_o و x_i و Γ تعداد کلمات موجود در دیکشنری هستند.

$$Pr(x_o | x_i) = \frac{\exp(\nu_{x_o}^T \cdot \nu_{x_i})}{\sum_{w \in \Gamma} \exp(\nu_w^T \cdot \nu_{x_i})} \quad (9-3)$$

شکل ۳-۵ طرح‌واره‌ای از مدل Skip-Gram را نمایش می‌دهد. همان‌طور که در این شکل مشخص است، در این مدل به نحوی بردارهای جاسازی برای کلمات تولید می‌شود که خطای تشخیص کلمات قبل و بعد از هر کلمه با داشتن بردار جاسازی آن کلمه، به کمترین مقدار ممکن برسد.



شکل ۳-۵: طرح‌واره‌ای از مدل Skip-Gram ارائه شده در پژوهش [۲۸]

با در نظر گرفتن این نکته که در این مدل، هدف اساسی در تولید بردارهای جاسازی، افزایش قابلیت تشخیص بردارهای جاسازی بعدی در جملات است، استفاده از این مدل جاسازی کلمات، علاوه بر مدل‌سازی معنای کلمه، فرایند یادگیری شبکه و تشخیص کلمات بعدی را بهبود می‌بخشد.

در این پژوهش، ابتدا جاسازی کلمات با استفاده از مدل ارائه شده در پژوهش [۲۸] برای تمام کلمات موجود در مجموعه شرح‌های مجموعه‌آموزشی، محاسبه و ذخیره می‌شود. در تمام پژوهش، از همین بردارهای جاسازی به عنوان بردارهای ویژگی کلمات استفاده می‌شود.

با تغییر مدل خروجی شبکه، از حالت بردار تک‌فعال، به بردار جاسازی به شرح فوق، مساله بهینه‌سازی قبلی نمی‌تواند منجر به یادگیری صحیح شبکه شود. در مدل جدید، به جای پیش‌بینی احتمال وقوع کلمات، باید یک مساله رگرسیون مناسب تعریف شود؛ به نحوی که خروجی شبکه در هر مرحله، هرچه بیشتر به بردار جاسازی کلمه مرحله بعدی، نزدیک شود. با این تفاسیر، مساله بهینه‌سازی (۲-۳) به مساله بهینه‌سازی (۱۰-۳) تبدیل می‌شود که در آن، y_{t+1} خروجی شبکه در لایه آخر و s_{t+1} بردار جاسازی کلمه $t+1$ در جمله مربوط به تصویر ورودی است.

$$\underset{\Xi}{\text{minimize}} \text{ } \text{tr}\{E\{(y_{t+1} - s_{t+1})(y_{t+1} - s_{t+1})^T\}\} \quad (10-3)$$

در روش‌های قبلی، از آن‌جا که هدف اصلی، تولید توزیع احتمال کلمات در لایه آخر رمزگشا بود، از یک لایه Softmax به منظور تبدیل خروجی شبکه به توزیع احتمال استفاده می‌شد. با توجه به تغییری که در این پژوهش انجام شد، نیازی به تولید توزیع احتمال در لایه آخر رمزگشا وجود ندارد و این لایه از معماری شبکه، حذف می‌شود. از طرف دیگر، استفاده از تابع هزینه Cross-Entropy به عنوان تابع خطای شبکه، در روش‌های قبلی مرسوم بود. رابطه (۱۱-۳)، نحوه محاسبه این تابع را برای توزیع‌های احتمال گستته p و q ، مشخص می‌نماید. همان‌طور که در این رابطه مشخص است، این تابع هزینه، شباهت زیادی با معیار فاصله توزیع kullback-leibler دارد.

$$H(p, q) = -\sum_x p(x) \log(q(x)) \quad (11-3)$$

این تابع هزینه، برای مدل‌سازی توزیع احتمال و آموزش شبکه برای پیش‌بینی یک توزیع احتمال در طول زمان، کارایی بسیار خوبی از خود نشان می‌دهد. اما از آن‌جا که مساله بهینه‌سازی شبکه رمزگشا در این پژوهش، با جای‌گزینی بردار جاسازی کلمات به جای بردار تک‌فعال، به مساله رگرسیون تغییر پیدا کرد، مطابق با مساله تعریف شده (۱۰-۳)، از میانگین مربع خطای^۴ به عنوان تابع هزینه شبکه استفاده می‌نماییم. رابطه (۱۲-۳) نحوه محاسبه خطای را بیان می‌کند.

$$\epsilon = \text{tr}\{E\{(y_{t+1} - s_{t+1})(y_{t+1} - s_{t+1})^T\}\} \quad (12-3)$$

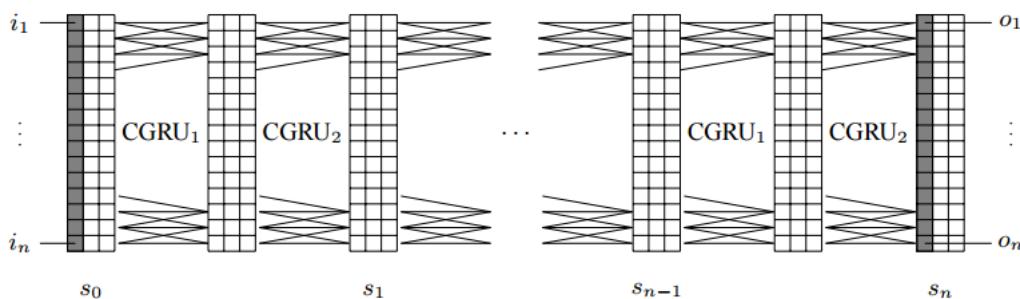
^۴Mean Squared Error

۴-۳ نحوه آموزش و تست شبکه

در مرحله آموزش شبکه در این پژوهش از بهینه‌ساز آدام^۹، که در پژوهش [۱۵] توسط آقای کینگما و همکارانش در سال ۲۰۱۴ ارائه شد، استفاده شده است. بهینه‌ساز آدام، یکی از بهینه‌سازهای دسته نزول تصادفی در امتداد گرادیان^{۱۰} به شمار می‌رود. این بهینه‌ساز در مسائلی که فضای پارامترهای قابل آموزش آن‌ها بسیار بزرگ است، عمل کرد بسیار خوبی از خود نشان می‌دهد. استفاده از این بهینه‌ساز در بین پژوهش‌های مربوط به تولید خودکار شرح بر تصاویر و مدل‌سازی زبان طبیعی، رایج است.

به منظور جلوگیری از بیش‌برازش^{۱۱} شبکه روی مجموعه‌داده آموزشی، از منظم‌سازی^{۱۲} حذف نود^{۱۳} استفاده شده است. این منظم‌سازی روی تمام لایه‌های داخلی شبکه رمزگشا، اعمال شده است. در موقعي که شبکه عصبی بازگشتی، خیلی بزرگ باشد و تعداد پارامترهای قابل آموزش آن زیاد باشند، استفاده از این منظم‌ساز باعث حذف برخی از واحدهای شبکه شده و از بیش‌برازش شبکه روی داده‌های مجموعه‌داده آموزشی، جلوگیری می‌شود.

شکل ۶-۳ معماری ارائه شده در این پژوهش را نمایش می‌دهد.



شکل ۶-۳: معماری ارائه شده در پژوهش حاضر

۵-۳ جمع‌بندی

در این فصل از گزارش، چارچوب کاری رمزگذار-رمزگشا را در حالت استاندارد مورد بررسی قرار دادیم. در این چارچوب، معمولاً از یک شبکه عصبی کانولوشنی عمیق به عنوان رمزگذار استفاده می‌شود. بسته به این که در فرایند پژوهش، توجه بصری مورد استفاده قرار گرفته یا خیر، رمزگذار می‌تواند یک یا بیش از یک بردار ویژگی از تصویر استخراج نماید.

در این پژوهش از شبکه عصبی Google Inception V3 به عنوان رمزگذار استفاده شده است. این شبکه کانولوشنی، برای دسته‌بندی تصاویر موجود در مجموعه‌داده ImageNet مورد آموزش قرار گرفته است و در حال حاضر به صورت از پیش آموزش دیده، در دسترس پژوهش‌گران می‌باشد. برای استفاده از این شبکه عصبی در کاربرد تولید خودکار شرح بر تصایر، لازم است، به جای خروجی لایه آخر شبکه، از خروجی دومین لایه قبل از لایه آخر به عنوان خروجی رمزگذار استفاده شود.

^۹ Adam Optimizer

^{۱۰} Stochastic Gradient Descent

^{۱۱} Overfitting

^{۱۲} Regularization

^{۱۳} Drop Out

خروجی دومین لایه قبل از لایه آخر در این شبکه، به لایه انتقال یادگیری معروف است. زیرا این لایه، تمام اطلاعات استخراج شده از تصویر را در خود نگهداری می‌کند و دو لایه آخر این شبکه که لایه‌های تماماً متصل هستند، به منظور استفاده از این شبکه در حوزه دسته‌بندی تصاویر مورد آموزش قرار گرفته‌اند و در کاربرد تولید خودکار شرح بر تصاویر، قابل استفاده نیستند.

لایه انتقال یادگیری در این شبکه کانولوشنی شامل تعداد 8×8 بردار 20×48 بعدی است که هر یک از این بردارها، علاوه بر این که اطلاعات استخراج شده از کل تصویر را در خود نگهداری می‌کنند، بر روی بخشی از تصویر ورودی تمکز دارند. این بردارها می‌توانند به عنوان بردارهای حاسیه‌نویسی تصویر مورد استفاده قرار بگیرند.

در حال حاضر، طراحی و حل معضلات موجود در بخش رمزگشا در این چارچوب کاری، چالش برانگیزتر از بخش رمزگذار به حساب می‌آید. وظیفه رمزگشا این است که با دریافت بردار یا بردارهای ویژگی تصویر که توسط رمزگذار استخراج شده است، شرح توصیف‌کننده تصویر را به صورت کلمه به کلمه تولید نماید.

در این پژوهش، از شبکه LSTM به عنوان واحد اصلی شبکه بازگشتی استفاده شده است. روابط (۳-۳) تا (۷-۳)، ساختار یک واحد LSTM را مدل سازی می‌نمایند. همین‌طور شکل ۳-۳، طرح‌واره‌ای از یک واحد LSTM را نمایش می‌دهد. چینش این واحدها به صورت پشت‌های بر روی هم، یک لایه از شبکه رمزگشا را تشکیل می‌دهد که در هر واحد زمانی، یک کلمه را تولید می‌نماید. شکل ۴-۳، ساختار پشت‌های شبکه رمزگشا را نمایش می‌دهد. در تمامی پژوهش‌های انجام شده در این حوزه، این تولید کلمه به کلمه را به صورت پیش‌بینی احتمال شرطی کلمه بعدی به شرط داشتن کلمات تولید شده قبلی و بردار یا بردارهای ویژگی تصویر، مدل می‌نمایند.

در این حالت، بردار خروجی شبکه در هر مرحله باید به تعداد کلمات موجود در دیکشنری کلمات، مولفه داشته باشد و هر مولفه از این بردار، احتمال رخداد کلمه متناظر خود در دیکشنری کلمات را نمایش دهد. برای تضمین این نکته که خروجی شبکه در هر مرحله حتماً یک توزیع احتمال خواهد بود، یک لایه Soft Max به انتهای شبکه اضافه می‌شود تا خروجی تولید شده را به یک توزیع احتمال تبدیل نماید.

با توجه به این نکته که شبکه در هر مرحله باید یک توزیع احتمال روی کلمات تولید نماید، در مرحله آموزش شبکه، کلمات موجود در شرح متناظر تصاویر در مجموعه‌داده آموزشی، باید به صورت بردار تک‌فعال درآمده و به عنوان مقدار مطلوب هر مرحله به شبکه داده شوند. به عبارت بهتر، مقدار مطلوب برای توزیع احتمال مطلوب شبکه در مرحله t ام از تولید جمله، یک بردار به طول تعداد کلمات موجود در دیکشنری است که تمام مولفه‌های آن مقدار صفر دارند و فقط مولفه‌ای از این بردار که اندیس آن با اندیس کلمه $1 + t$ در شرح موجود برای تصویر در دیکشنری لغات برابر است، مقدار یک دارد.

استفاده از بردار تک‌فعال به عنوان خروجی مطلوب شبکه، باعث ایجاد مشکلاتی می‌شود که از جمله آن‌ها می‌توان به نکات زیر اشاره کرد:

۱. عدم استفاده از اطلاعات معنایی کلمات، استفاده از کلمات مترادف و تولید جملات متنوع‌تر

۲. تنک‌شدن مقدار مطلوب شبکه و کاهش قدرت شبکه در یادگیری پارامترها

۳. افزایش چشم‌گیر تعداد پارامترهای قابل یادگیری شبکه به دلیل بالابودن ابعاد خروجی در تمام مراحل

روش پیشنهادی در این پژوهش، به منظور حل مشکلات فوق‌الذکر، استفاده از بردار جاسازی کلمات به جای بردار تک‌فعال به عنوان بردار مطلوب شبکه است. این جایگزینی علاوه بر ایجاد امکان استفاده از اطلاعات معنایی کلمات، از تنک شدن مقادیر مطلوب شبکه جلوگیری می‌نماید. همین‌طور استفاده از جاسازی کلمات، امکان کاهش چشم‌گیر اندازه خروجی مطلوب شبکه در هر مرحله را فراهم نموده و پیرو آن باعث کاهش دادن اندازه شبکه شده، تعداد پارامترهای قابل آموزش را کاهش داده و روند یادگیری شبکه را سهولت می‌بخشد.

به منظور ایجاد جاسازی مناسب برای کلمات، از مدل Skip-Gram ارائه شده در پژوهش [۲۸] استفاده شده است. در این مدل، مطابق با رابطه (۸-۳)، جاسازی کلمات به نحوی تولید می‌شود که با داشتن بردار جاسازی یک کلمه، بتوان بردارهای جاسازی کلمات قبل و بعد را دسته‌بندی نمود. این خاصیت در بردارهای جاسازی مورد استفاده، علاوه بر حل مشکلات بردار تک‌فعال، باعث سهولت بیشتر شبکه بازگشتی در تولید بردارهای جاسازی بعدی می‌شود.

لازم به توضیح است، جاسازی تمام کلمات موجود در دیکشنری، در این پژوهش با استفاده از جملات موجود در توصیفات تصاویر مجموعه‌داده آموزشی، انجام شده است. به عبارت بهتر، مدل جاسازی کلمات، با دریافت تمام توصیفات موجود در مجموعه‌داده آموزشی، برای تمام کلمات موجود در دیکشنری، بردار جاسازی متناظر را تولید می‌نماید و در تمام فرایندهای بعدی، شامل فرایندهای آموزش و تست شبکه، از این بردارهای جاسازی از پیش ذخیره شده، مستقیماً استفاده می‌شود.

با تغییر خروجی مطلوب شبکه از حالت بردار تک‌فعال به بردار جاسازی، مساله (۲-۳)، که یک مساله پیش‌بینی توزیع احتمال است، به مساله (۱۰-۳)، که یک مساله رگرسیون به حساب می‌آید، تغییر پیدا می‌کند. برای حل این مساله رگرسیون، ناگزیر به تغییرتابع خطای شبکه از تابع هزینه Cross Entropy به تابع میانگین مربع خطای هستیم.

در فرایند آموزش شبکه، از بهینه‌ساز آدام استفاده شده است. به علاوه برای جلوگیری از بیش‌برازش شبکه بازگشتی روی داده‌های موجود در مجموعه‌داده آموزشی، از منظم‌ساز حذف نود که یکی از پرکاربردترین منظم‌سازهای مورد استفاده در پژوهش‌های حوزه تولید خودکار شرح بر تصاویر به شمار می‌رود، استفاده شده است.

فصل چهارم

پیاده‌سازی، آزمون و ارزیابی

۱-۴ مقدمه

در این گزارش، مساله تولید خودکار شرح بر تصاویر را معرفی نموده و سیر تکاملی پژوهش‌های انجام شده در این حوزه را به طور اجمالی مرور کرده و روش‌های مختلفی که در این حوزه مورد استفاده قرار گرفتند، رویکردها، نقاط ضعف و قوت هر یک از این روش‌ها و روند فعلی پژوهش‌ها در این حوزه را بررسی نمودیم. در ادامه، چارچوب کاری رمزگذار-رمزگشا را، که در حال حاضر عمدۀ پژوهش‌گران را به خود جلب کرده است، در حالت استاندارد مطالعه نموده و نقاط ضعف آن را بیان کردیم.

در فصل گذشته، ایده اصلی پژوهش را، جایگزینی بردار جاسازی کلمات به جای بردار تکفعال، مورد بررسی قرار دادیم و ساختار پیشنهادی خود برای بهبود چارچوب کاری رمزگذار-رمزگشا و جنبه‌های مختلف آموزش و تست این ساختار را بیان نمودیم.

در این فصل از گزارش، ابتدا کلیاتی در رابطه با پیاده‌سازی پروژه را بیان خواهیم نمود. شایان ذکر است که اصلی پروژه به همراه توضیحات جزئی در رابطه با آن، در پیوست اول این گزارش آورده شده است. در ادامه این بخش از گزارش، معیارهای ارزیابی مطرح در حوزه تولید خودکار شرح بر تصاویر را بررسی نموده، نگاهی بر مجموعه‌داده مورد استفاده اندادته و عمل کرد ساختار پیشنهادی خود را با روش‌های مشابه دیگر مقایسه می‌نماییم.

۲-۴ پیاده‌سازی

سامانه تولید خودکار شرح بر تصاویر که در این گزارش، جنبه‌های مختلف آن را مورد بررسی قرار داده‌ایم، به زبان پایتون و با استفاده از نسخه ۳.۵ آن پیاده‌سازی شده است. در پیاده‌سازی این پروژه، از چارچوب کاری Tensorflow^۱ و از کتابخانه جنسیم^۲ به منظور بهره‌گیری از ماثول جاسازی کلمات، استفاده شده است. در این بخش از گزارش به معرفی چارچوب کاری Tensorflow می‌پردازیم. که اصلی پروژه به همراه توضیحات آن در پیوست اول این گزارش ضمیمه شده است.

۱-۲-۴ چارچوب کاری Tensorflow

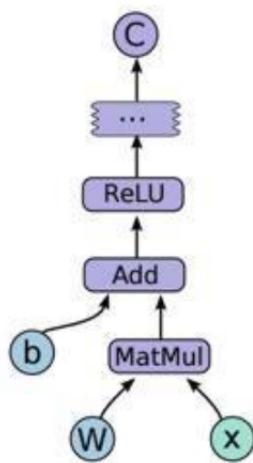
چارچوب کاری Tensorflow، یکی از جدیدترین و پرکاربردترین چارچوب‌های کاری در حوزه یادگیری ماشین، در زمان نگارش این گزارش، به شمار می‌رود. هسته این چارچوب کاری به زبان C++ و با استفاده از پلتفرم محاسبات موازی CUDA پیاده‌سازی شده است. این چارچوب کاری، یک واسط به زبان پایتون، توسعه داده است که استفاده از آن را در زبان پایتون به سهولت امکان‌پذیر می‌سازد. شایان ذکر است این چارچوب کاری توسط تیم Google Brain در حال توسعه است و پشتیبانی می‌شود.

ایدئولوژی اصلی در این چارچوب کاری، بیان محاسبات در قالب گراف است. برای استفاده از این چارچوب کاری، کافیست گراف محاسبات مورد نظر، در این چارچوب طراحی و پیاده‌سازی شود. هر نод این گراف، یک واحد محاسباتی را مشخص می‌کند. با این رویکرد می‌توان هر شبکه عصبی پیچیده‌ای را به راحتی پیاده‌سازی نمود.

¹Tensorflow

²Gensim

شکل ۱-۴ یک مثال ساده از چنین گرافی را برای یک شبکه پیش‌رو ساده نمایش می‌دهد که نودهای دایره‌ای در آن، تنسورها را نمایش داده و نودهای مستطیلی، نمایان گر گره‌های محاسباتی هستند.



شکل ۱-۴: یک نمونه از گراف‌های محاسباتی در چارچوب کاری تنسورفلو

داده‌های ورودی، ثوابت و متغیرهای قابل آموزش، همگی در قالب تنسور در این چارچوب کاری قابل تعریف هستند. روند حرکت یک تنسور در گراف طراحی شده و اعمال محاسبات هر نod از گراف روی آن، فرایند اجرای شبکه را تشکیل می‌دهد. به همین دلیل، این چارچوب کاری را تنسورفلو نامیده‌اند. این چارچوب کاری را با نمونه‌های مشابه خود به طور اجمالی مقایسه می‌نماید.

۳-۴ [۲۹] معیار BLEU

معیار ^۳BLEU، یکی از معیارهای ارزیابی مدل‌های ترجمه ماشینی است که در حوزه تولید خودکار شرح بر تصاویر نیز مورد استفاده قرار می‌گیرد. در حوزه ترجمه ماشینی، ترجمه‌های مختلفی از یک جمله در زبان مبدا، می‌توان در زبان مقصود ارائه داد. برای تشخیص بهترین ترجمه بین ترجمه‌های کاندید برای یک جمله در زبان مبدا، می‌توان از این معیار استفاده نمود.

نکاتی که در تخمین این معیار موثر هستند به شرح زیر می‌باشند:

۱. تعداد n-gram‌های اشتراکی با ترجمه‌های مرجع

۲. طول نامتعارف ترجمه، امتیاز منفی برای ترجمه‌های خیلی بلند یا خیلی کوتاه

۳. امتیاز منفی برای تکرار بیش از حد کلمات

۴. امتیاز منفی برای تکرار کلمات هم‌معنی بیش از حد متعارف

۴-۴ [۱۹] معیار METEOR

این معیار نیز یکی از معیارهایی است که در حوزه ترجمه ماشینی مورد استفاده قرار می‌گیرد. هدف این معیار، اندازه‌گیری میزان اनطباق ترجمه تولید شده توسط ماشین با ترجمه‌های مرجع به صورت کلمه به کلمه است.

^۳Bidirectional Evaluation Understudy

در صورتی که بیش از یک ترجمه مرجع وجود داشته باشد، این معیار با تمام ترجمه‌های مرجع به طور مستقل اندازه‌گیری شده و بیشترین مقدار آن، به عنوان امتیاز ترجمه در نظر گرفته می‌شود. انطباق کلمه به کلمه کلمات دو ترجمه، با در نظر گرفتن یک هم‌ترازسازی کلمات انجام می‌شود. این هم‌ترازسازی، یک نگاشت بین کلمات ترجمه تولیدشده توسط ماشین به حداکثر یکی از کلمات موجود در ترجمه مرجع است که برای هر انطباق میزان انطباق به صورت یک متغیر سه مقداره ذخیره می‌شود. مقادیر این متغیر می‌تواند یکی از موارد زیر باشد:

۱. انطباق دقیق

۲. انطباق ریشه کلمات

۳. انطباق معنی کلمات

پس از یافتن تمام انطباق‌های مذکور بین دو ترجمه، بلندترین زیردبالة مشترک از این انطباق‌ها بین دو ترجمه انتخاب می‌شود. با توجه به تعداد کلمات موجود در ترجمه تولید شده و ترجمه مرجع، تعداد انطباق‌های یافت شده در بلندترین زیردبالة مشترک و تعداد کلمات موجود بدون انطباق در ترجمه مرجع، دقت^۴ و به خاطرسپاری^۵ انطباق‌ها محاسبه شده و بین آن‌ها میانگین هارمونیک محاسبه می‌شود.

در این معیار، پنالتی‌هایی برای خطای موجود در هم‌ترازسازی در نظر گرفته می‌شود. اگر دو ترجمه دقیقاً مطابق یکدیگر باشند، بیشترین امتیاز در این معیار را کسب می‌کنند. اگر با حذف بزرگترین زیردبالة مشترک از انطباق‌ها، ترجمه ماشینی تولیدشده به قطعات ریز دیگری تبدیل شود، هرچه تعداد این قطعات بیشتر باشد، امتیاز بیشتری از نتیجه نهایی کسر می‌شود.

۴-۵ معيار ROUGE-L [۲۱]

معیار ROUGE^۶، یکی از معیارهای مورد استفاده در خلاصه‌سازی متن است. این معیار، کیفیت یک خلاصه کاندید ارائه شده برای یک متن را در مقابل خلاصه‌های مرجع دیگر می‌سنجد. شاخه‌های متنوعی از این معیار، در بین پژوهش‌های مربوط به خلاصه‌سازی خودکار متون ارائه شده است که در حوزه تولید خودکار شرح بر تصاویر، یکی از این شاخه‌ها مورد توجه قرار می‌گیرد. در این قسمت ما معیار L ROUGE را که در بین پژوهش‌های مرتبط مورد استفاده قرار می‌گیرد، بررسی می‌نماییم.

این معیار، بلندترین زیردبالة مشترک بین خلاصه تولیدشده توسط ماشین و خلاصه‌های مرجع را محاسبه کرده و بر اساس این بلندترین زیردبالة مشترک، به خلاصه تولید شده امتیاز می‌دهد.

۴-۶ معيار CIDEr [۳۵]

معیار CIDEr^۷، برخلاف معیارهای دیگر، در بین پژوهش‌گران حوزه تولید خودکار شرح بر تصاویر ارائه شده است. این معیار در سال ۲۰۱۵ توسط ودانتم و همکارانش ارائه شد. هدف اصلی این معیار این است که توافق شرح‌های مرجع تولید شده توسط انسان را یافته و سپس میزان انطباق شرح تولید شده خودکار با این توافق را اندازه‌گیری

^۶Precision

^۷Recall

^۸Recall Oriented Understudy Gisting Evaluation

^۹Consensus-Based Image Description Evaluation

نماید.

در این معیار، ابتدا با اندازه‌گیری TF-IDF برای هر کلمه، میزان اهمیت کلمات و معانی آن‌ها در شرح تولید شده، محاسبه می‌شود. این میزان اهمیت به عنوان وزن کلمه در محاسبات بعدی مورد استفاده قرار می‌گیرد. توصیفات تولید شده به طور خودکار و توصیفات مرجع، به صورت مجموعه‌ای از عبارات n-gram توصیف می‌شوند. این توصیف از عبارات شامل ۱ تا ۴ کلمه پشت‌سرهم را شامل می‌شوند. سپس میزان انطباق این توصیفات با یکدیگر بر اساس وزن کلمات و عبارات، محاسبه می‌شود. در نهایت با استفاده از یک معیار شباهت کسینوسی، میزان شباهت توصیفات تولید شده محاسبه شده و به عنوان امتیاز هر یک از دسته‌های ۱ تا ۴ تایی، گزارش می‌شوند.

۷-۴ آزمایشات

در این قسمت ابتدا به معرفی مجموعه‌داده مورد استفاده پرداخته و سپس آزمایشات انجام شده و نتایج بدست آمده را گزارش می‌نماییم. همین‌طور نمونه‌هایی از خروجی‌های صحیح و غلط مدل ارائه شده را نمایش می‌دهیم. در تمامی آزمایشات انجام شده در این بخش، یک مدل پشتۀ‌ای ۲۰ لایه، با بردار حالت ۱۰۲۴ بعدی توسط مجموعه‌داده آموزشی، آموزش داده شده است. نرخ یادگیری در تمام این موارد ۱۰۰٪ و احتمال حذف نود برابر ۵٪ در نظر گرفته شده است. فضای جاسازی کلمات، یک فضای ۱۰۲۴ بعدی در نظر گرفته شده است و قبل از شروع فرایند آموزش شبکه، مدل جاسازی کلمات روی تمام جملات موجود در مجموعه‌داده آموزشی، آموزش داده می‌شود.

۷-۱ معرفی مجموعه‌داده

مجموعه‌داده مورد استفاده، مجموعه‌داده MS-COCO است که توسط مایکروسافت فراهم شده است. پژوهش [۲۳]، اطلاعات کاملی در رابطه با این مجموعه‌داده فراهم کرده است. در این مجموعه‌داده، حدود ۸۳۰۰۰ تصویر از موضوعات مختلف گردآوری شده به همراه مجموعاً حدود ۴۱۴۰۰۰ شرح متناظر با تصاویر، در مجموعه‌داده آموزشی و در هر کدام از مجموعه‌داده‌های اعتبارسنجی^۱ و آزمون، حدود ۴۱۰۰۰ تصویر با حدود ۲۰۰۰۰۰ شرح متناظر با تصاویر وجود دارد. هر تصویر به طور میانگین دارای ۵ شرح مختلف است که توسط انسان تولید شده است. شایان ذکر است تصاویر موجود، در ابعاد مختلف گردآوری شده‌اند.

۷-۲ نمونه‌هایی از خروجی‌ها

در ادامه، نمونه‌هایی از تصاویر ورودی را به همراه شرح تولید شده توسط مدل ارائه شده در این پژوهش مشاهده می‌نمایید.

خروچی‌های صحیح

در جدول ۱-۴، نمونه‌هایی از تصاویری موجود در مجموعه‌داده که شرح تولید شده توسط مدل پیشنهادی صحیح است را مشاهده می‌نمایید. در این جدول، شرح تولید شده به طور خودکار برای هر تصویر، به همراه مجموعه شرح‌های مرجع برای آن تصویر قابل مشاهده است.

^۱Validation

جدول ۴-۱: نمونه‌هایی از خروجی‌های صحیح مدل ارائه شده در پژوهش

تصویر نمونه	شرح تولید شده	شرح مرجع
	Giraffe standing in a tree filled area	A giraffe standing next to a forest filled with trees. A giraffe eating food from the top of the tree. A giraffe standing up nearby a tree. A giraffe mother with its baby in the forest. Two giraffes standing in a tree filled area.
	Man on a surf board in the ocean	A man laying on a surfboard in the water. A man lying on a surfboard in some small waves in water. A young man paddles a surfboard in the ocean. A giraffe mother with its baby in the forest. The person is riding his surfboard out in the ocean.
	Plane is on the ground on the tarmac	A modern jet airliner on a snow edged runway. The airplane is on the ground on the runway. The front view of a white jet on an airports runway. A large air plane on an airport runway. A plane that has jet landed at the airport.

خروجی‌های غلط

جدول ۲-۴، نمونه‌هایی از تصاویر را که شرح تولیدشده برای آن‌ها غلط بوده، نمایش می‌دهد.

۳-۷-۴ مقایسه با روش‌های مشابه

تعداد پارامترها

دیکشنری کلمات ایجاد شده روی مجموعه‌داده آموزشی مورد استفاده، بالغ بر ۲۳۰۰۰ کلمه را شامل می‌شود. استفاده از بردار تک‌فعال به عنوان بردار مطلوب و پیش‌بینی توزیع احتمال کلمات در هر مرحله، با ساختار مشخص شده شبکه، منجر به ایجاد حدود ۴۰۰ میلیون پارامتر قابل آموزش می‌شود. با جای‌گزینی مدل جاسازی کلمات در ابعاد ۱۰×۲۴ بعدی، ساختار یکسان، برای رگرسیون جاسازی کلمات، تنها ۸۶ میلیون پارامتر قابل آموزش خواهد داشت. به عبارت دیگر، تعداد پارامترهای قابل آموزش شبکه، در حالت استفاده از جاسازی کلمات، حدود ۲۱.۵٪ تعداد پارامترهای مورد نیاز برای بردار تک‌فعال است.

علاوه بر مورد فوق، شبکه در حالت استفاده از جاسازی کلمات، برای هم‌گرایی نیاز به حدود ۲۰ تکرار دارد. این در حالی است که همین ساختار برای استفاده از بردار تک‌فعال، در حدود ۱۱۰ تکرار به هم‌گرایی می‌رسد و این نشان‌گر این نکته است که کاهش ابعاد خروجی و حذف بردارهای تنک باعث سهولت یادگیری شبکه و افزایش سرعت هم‌گرایی در شبکه می‌شود.

جدول ۲-۴: نمونه‌هایی از خروجی‌های غلط مدل ارائه شده در پژوهش

تصویر نمونه	شرح تولید شده	شرح مرجع
	Old man in a coat has a giraffis look on his face	A man stands with a frown on his face. A old man in coat and tie walking down a busy street. An old man in a business suit has a thoughtful face. An older gentleman wearing a suit has a grumpy look on his face. An old man wearing glasses who doesn't look very happy.
	Train of an empty intersection with traffic lights	Two traffic lights are posted near the street intersection. An empty street with some stop lights in a little island. Topside view of an empty intersection with traffic lights. A street intersection with some traffic lights on the side walk. Some stop signs that are at an intersection.
	small black and dog with a frisbee by its feet	A small dog standing on a wet ground looking up. A small black and white dog standing on a sparse grass looking at ahuman. A small black and white dog standing next to a pink and black frisbee. A dog looking up at a person with a frisbee at their feet.

BLEU معيار

۸-۴ بحث درباره نتایج و عملکرد شبکه

۹-۴ جمع‌بندی

فصل پنجم

جمع‌بندی، نتیجه‌گیری و پیشنهادات

تولید و ذخیره‌سازی روزافزون تصاویر، سهولت در استفاده از دوربین‌های تصویربرداری و گوشی‌های موبایل، دسترسی آسان به اینترنت در تمام نقاط شهر و افزایش تعداد شبکه‌های اجتماعی و نرم‌افزارهای موبایل، باعث افزایش نیاز کاربران به سامانه‌های هوشمند مدیریت تصاویر شده است. سامانه‌هایی که علاوه بر مدیریت ذخیره و بازیابی تصاویر، قدرت دسته‌بندی خودکار، جستجوی محتوایی، درک و توصیف تصاویر از هر موضوعی باشند.

ارائه مدل‌های هوشمند که بتوانند به طور خودکار برای هر تصویری، توصیف متناظر در قالب جملات زبان طبیعی تولید کنند، از جمله مهم‌ترین اقدامات در راستای رسیدن به سامانه مدیریت تصاویر به شمار می‌رود.

روند پژوهش در مساله تولید خودکار شرح بر تصاویر، به سال‌های ۱۹۷۰ تا ۱۹۸۰ بر می‌گردد. در این سال‌ها اولین پژوهش‌ها با موضوع شناخت فرایند توصیف تصویر توسط مغز انسان انجام شد. پژوهش‌گران در این پژوهش‌ها، با بررسی افرادی که در یک فضای کنترل شده، تصاویر متفاوتی را توصیف می‌کنند، سعی در شناخت ویژگی‌های فرایند توصیف تصویر توسط مغز انسان داشتند. فرایند توصیف تصویر کاز دو مرحله استخراج اطلاعات تصویر و بیان اطلاعات استخراج شده در قالب جملات تشکیل شده است. طی بررسی‌های انجام شده، مشخص شد مغز انسان در کمتر از حدود ۵۰۰ تا ۲۰۰ میلی‌ثانیه، قادر است تمام اطلاعات مورد نیاز برای توصیف تصویر را استخراج نماید.

با طرح مساله و چالش جدید، توجه پژوهش‌گران به حل این مساله، جلب شد. در سال‌های قبل از ۲۰۰۰، عموم چالش‌های برجسته در این حوزه، مربوط به بخش درک صحنه و استخراج اطلاعات موجود در تصویر بود. در این سال‌ها عموماً، پژوهش‌گران با محدود کردن موضوع تصاویر مورد بررسی و با استفاده از ویژگی‌های تصویر مانند لبه‌ها، توصیف کننده‌های مختلف، بافت و موارد مشابه دیگر، سعی در استخراج اطلاعات تعریف شده داشتند.

در سال‌های ۲۰۰۰ تا ۲۰۱۴، با وارد شدن روش‌های گرافی احتمالی، به حوزه تولید خودکار شرح بر تصاویر، به مرور زمان محدودیت‌های گذشته برطرف شد و فرایند استخراج اطلاعات از تصاویر به خوبی و بدون نیاز به محدود کردن موضوع تصاویر، انجام می‌شد. مدل‌های گرافی احتمالی زیادی در این مسیر به کار گرفته شد. از جمله این مدل‌ها می‌توان به مدل میدان تصادفی مارکف و میدان تصادفی شرطی اشاره کرد. همین‌طور پژوهش‌گرانی هم وجود داشتند که برای حل این چالش، مدل جدیدی را طراحی و تولید نمایند که نمونه‌هایی از آن‌ها را در بخش مرور مطالعات پیشین بررسی نمودیم.

در سال‌های بعد از ۲۰۰۷، می‌توان گفت توجه پژوهش‌گران بیشتر به سمت مدل‌های تولید جمله جلب شد و چالش‌های موجود در این حوزه که اغلب بدون راه حل بودند یا با راه حل‌های ابتدایی حل می‌شدند، بیش از پیش مورد استقبال پژوهش‌گران قرار گرفتند. مدل‌های مختلفی برای تولید جمله به کار گرفته شد. از جمله این مدل‌ها می‌توان به روش‌های موجود در حوزه تولید زبان طبیعی، بازیابی شبیه‌ترین جمله موجود در مجموعه‌داده و استفاده از کلیشه زبانی، اشاره کرد. اما هیچ‌یک از این روش‌ها، نتوانستند تمام معضلات را حل نمایند.

اما با حل مشکل ناپایداری آموزش شبکه‌های عصبی بازگشتی در سال ۲۰۱۱ توسط هینتون، فصل جدیدی در حوزه تولید جمله در این مساله شروع شد. شبکه‌های عصبی بازگشتی، ابزارهای قدرتمندی در کاربرد پیش‌بینی دنباله‌های زمانی و تولید جمله به شمار می‌روند. قابلیت‌های بالای این مدل‌ها، پژوهش‌گران را برآن داشت که تمامی روش‌های گذشته را کنار گذاشته و تماماً از شبکه‌های عصبی بازگشتی برای تولید جمله استفاده نمایند. استفاده از شبکه‌های عصبی بازگشتی، ذهن اغلب پژوهش‌گران را به سمت استفاده از شبکه‌های کانولوشنی عمیق در مرحله استخراج اطلاعات از تصاویر می‌کشاند. شبکه‌های عصبی کانولوشنی عمیق، در استخراج ویژگی‌های بسیار خوب از تصاویر، قدرت بالایی دارند. از حدود سال ۲۰۱۴ به بعد و با جابجایی مدل‌های گرافی احتمالی با شبکه‌های عصبی کانولوشنی عمیق، صفر تا صد فرایند تولید خودکار شرح بر تصاویر، با استفاده از شبکه‌های عصبی و یادگیری عمیق انجام می‌شد.

با گرایش تمامی پژوهش‌ها در این حوزه به یادگیری عمیق، عمل کرد مدل‌های ارائه شده، دچار یک جهش اساسی در این زمینه شد. همین طور در سال ۲۰۱۵، چارچوب کاری رمزگذار-رمزگشا، یکی از مشهورترین چارچوب‌های کاری یادگیری عمیق، که پیش‌تر کاربردهای بسیار زیادی در حوزه ترجمه ماشینی یافته بود، پا به عرصه پژوهش‌های مرتبط با تولید خودکار شرح بر تصاویر نهاد. از آن پس، این چارچوب کاری تبدیل به جزو لاینفکی از پژوهش‌های حوزه تولید خودکار شرح بر تصاویر شده است.

ورود چارچوب کاری رمزگذار-رمزگشا از حوزه ترجمه ماشینی به حوزه تولید خودکار شرح بر تصاویر، باب انتقال روش‌های دیگر را بین این دو حوزه، بیش از پیش گشود. در سال‌های اخیر، روش‌های مبتنی بر توجه که در حوزه ترجمه ماشینی، عمل کرد بسیار خوبی از خود نشان دادند، به حوزه تولید خودکار شرح بر تصاویر وارد شدند. در حال حاضر، بیش‌ترین پژوهش‌های انجام شده در حوزه تولید خودکار شرح بر تصاویر، مبتنی بر روش‌های توجه بصری هستند.

چارچوب کاری رمزگذار-رمزگشا در حوزه تولید خودکار شرح بر تصاویر در تمامی پژوهش‌ها، تقریباً به طور مشابه مورد استفاده قرار می‌گیرد. رمزگذار مورد استفاده در این چارچوب کاری، یک شبکه عصبی کانولوشنی عمیق است که به ازای هر تصویر ورودی، یک یا چند بردار ویژگی تولید می‌نماید. در پژوهش‌های مختلف، از شبکه‌های عصبی کانولوشنی مختلفی به این منظور استفاده می‌شود. عموماً استفاده از این شبکه‌ها به طور از پیش آموزش دیده، اتفاق می‌افتد.

وظیفه واحد رمزگشا در چارچوب کاری رمزگذار-رمزگشا این است که با دریافت بردار ویژگی استخراج شده یا بردارهای حاشیه‌نویسی تصویر، جملات توصیف کننده مربوط به تصویر را تولید نماید. هر جمله را می‌توان به عنوان دنباله‌ای با طول متغیر از کلمات در نظر گرفت که هر کلمه در آن، با یک بردار ویژگی توصیف شده است. در تمام پژوهش‌های انجام شده در این حوزه، رمزگشا در هر مرحله، سعی در پیش‌بینی توزیع احتمال شرطی کلمه بعدی به شرط داشتن کلمات تولید شده قبلی و بردار ویژگی تصویر می‌نماید.

به منظور آموزش شبکه رمزگشا به نحوی که قادر به تولید توزیع احتمال شرطی کلمات بعدی به شرط داشتن کلمات قبلی و بردار ویژگی تصویر باشد، باید مقادیر مطلوب در هر مرحله، به شکل یک توزیع احتمال روی تمام کلمات، به شبکه داده شود. برای این‌کار، در هر مرحله از یک بردار تک‌فعال به عنوان بردار مطلوب استفاده می‌شود. استفاده از بردار تک‌فعال به عنوان بردار مطلوب در هر مرحله مشکلاتی را به همراه دارد که از آن جمله می‌توان به نکات زیر اشاره کرد:

۱. عدم استفاده از اطلاعات معنایی کلمات، استفاده از کلمات مترادف و تولید جملات متنوع‌تر

۲. تنک‌شدن مقدار مطلوب شبکه و کاهش قدرت شبکه در یادگیری پارامترها

۳. افزایش چشم‌گیر تعداد پارامترهای قابل یادگیری شبکه به دلیل بالابودن ابعاد خروجی در تمام مراحل

ایده اصلی ارائه شده در این پژوهش، استفاده از بردار جاسازی کلمات به جای بردار تک‌فعال به عنوان مقدار مطلوب در هر مرحله است. بر همین اساس، مدل رمزگشا به نحوی تغییر می‌کند که رمزگشا به جای پیش‌بینی توزیع احتمال شرطی کلمات، بردار جاسازی کلمه بعدی را در هر مرحله تولید نماید. به عبارت دیگر، مساله پیش‌بینی توزیع احتمال شرطی در شبکه رمزگشا، با تغییر مذکور، به یک مساله رگرسیون تغییر پیدا می‌کند.

بردار جاسازی مورد استفاده در این پژوهش، با استفاده از مدل Skip-Gram تولید می‌شود. ویژگی برجسته این مدل در این است که بردار جاسازی کلمات به نحوی تولید می‌شود که با داشتن آن بتوان بردار جاسازی کلمات بعدی را به راحتی دسته‌بندی نمود. این ویژگی، علاوه بر این که مدل رمزگشا را قادر به استفاده از اطلاعات معنایی

کلمات می‌کند، یادگیری پیش‌بینی بردارهای جاسازی بعدی در هر مرحله را سهولت می‌بخشد.

مراجع و مراجع

- [1] Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
- [2] Chen, Xinlei and Lawrence Zitnick, C. Mind's eye: A recurrent visual representation for image caption generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2422–2431, 2015.
- [3] Cho, Kyunghyun, Courville, Aaron, and Bengio, Yoshua. Describing multimedia content using attention-based encoder-decoder networks. *IEEE Transactions on Multimedia*, 17(11):1875–1886, 2015.
- [4] Divvala, Santosh K, Hoiem, Derek, Hays, James H, Efros, Alexei A, and Hebert, Martial. An empirical study of context in object detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1271–1278. IEEE, 2009.
- [5] Farhadi, Ali, Hejrati, Mohsen, Sadeghi, Mohammad Amin, Young, Peter, Rashtchian, Cyrus, Hockenmaier, Julia, and Forsyth, David. Every picture tells a story: Generating sentences from images. In *Computer Vision–ECCV 2010*, pages 15–29. Springer, 2010.
- [6] Fei-Fei, Li, Iyer, Asha, Koch, Christof, and Perona, Pietro. What do we perceive in a glance of a real-world scene? *Journal of vision*, 7(1):10–10, 2007.
- [7] Felzenszwalb, Pedro, McAllester, David, and Ramanan, Deva. A discriminatively trained, multiscale, deformable part model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.
- [8] Felzenszwalb, Pedro F, Girshick, Ross B, McAllester, David, and Ramanan, Deva. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

-
- [9] Fidler, Sanja, Sharma, Abhishek, and Urtasun, Raquel. A sentence is worth a thousand pixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1995–2002, 2013.
 - [10] Girshick, Ross, Donahue, Jeff, Darrell, Trevor, and Malik, Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
 - [11] He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.
 - [12] Karpathy, Andrej and Fei-Fei, Li. Deep visual-semantic alignments for generating image descriptions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
 - [13] Karpathy, Andrej and Fei-Fei, Li. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.
 - [14] Karpathy, Andrej, Joulin, Armand, and Li, Fei Fei F. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897, 2014.
 - [15] Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
 - [16] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
 - [17] Ladicky, Lubor, Russell, Chris, Kohli, Pushmeet, and Torr, Philip HS. Graph cut based inference with co-occurrence statistics. In *Computer Vision–ECCV 2010*, pages 239–253. Springer, 2010.
 - [18] Ladicky, Lubor, Sturgess, Paul, Alahari, Karteek, Russell, Chris, and Torr, Philip HS. What, where and how many? combining object detectors and crfs. In *Computer Vision–ECCV 2010*, pages 424–437. Springer, 2010.

- [19] Lavie, Alon and Agarwal, Abhaya. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231. Association for Computational Linguistics, 2007.
- [20] Li, Li-Jia and Fei-Fei, Li. What, where and who? classifying events by scene and object recognition. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.
- [21] Lin, Chin-Yew. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain, 2004.
- [22] Lin, Dahua, Fidler, Sanja, and Urtasun, Raquel. Holistic scene understanding for 3d object detection with rgbd cameras. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.
- [23] Lin, Tsung-Yi, Maire, Michael, Belongie, Serge, Hays, James, Perona, Pietro, Ramanan, Deva, Dollár, Piotr, and Zitnick, C Lawrence. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [24] Luong, Minh-Thang, Pham, Hieu, and Manning, Christopher D. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.
- [25] Mao, Junhua, Xu, Wei, Yang, Yi, Wang, Jiang, and Yuille, Alan L. Explain images with multimodal recurrent neural networks. *arXiv preprint arXiv:1410.1090*, 2014.
- [26] Mikolov, Tomas, Chen, Kai, Corrado, Greg, and Dean, Jeffrey. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [27] Mikolov, Tomas, Karafiat, Martin, Burget, Lukas, Cernocky, Jan, and Khudanpur, Sanjeev. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.
- [28] Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg S, and Dean, Jeff. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

-
- [29] Papineni, Kishore, Roukos, Salim, Ward, Todd, and Zhu, Wei-Jing. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.
 - [30] Potter, Mary C. Short-term conceptual memory for pictures. *Journal of experimental psychology: human learning and memory*, 2(5):509, 1976.
 - [31] Potter, Mary C, Staub, Adrian, Rado, Janina, and O’Connor, Daniel H. Recognition memory for briefly presented pictures: the time course of rapid forgetting. *Journal of Experimental Psychology: Human Perception and Performance*, 28(5):1163, 2002.
 - [32] Szegedy, Christian, Liu, Wei, Jia, Yangqing, Sermanet, Pierre, Reed, Scott, Anguelov, Dragomir, Erhan, Dumitru, Vanhoucke, Vincent, and Rabinovich, Andrew. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
 - [33] Szegedy, Christian, Vanhoucke, Vincent, Ioffe, Sergey, Shlens, Jon, and Wojna, Zbigniew. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.
 - [34] Uijlings, Jasper RR, van de Sande, Koen EA, Gevers, Theo, and Smeulders, Arnold WM. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.
 - [35] Vedantam, Ramakrishna, Lawrence Zitnick, C, and Parikh, Devi. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.
 - [36] Xu, Kelvin, Ba, Jimmy, Kiros, Ryan, Cho, Kyunghyun, Courville, Aaron C, Salakhutdinov, Ruslan, Zemel, Richard S, and Bengio, Yoshua. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 14, pages 77–81, 2015.
 - [37] Yang, Zichao, He, Xiaodong, Gao, Jianfeng, Deng, Li, and Smola, Alex. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29, 2016.

پیوست

در این بخش قصد داریم سورس کد اصلی برنامه را به صورت دقیق مورد بررسی قرار دهیم. در ادامه، ابتدا نگاه کوتاهی بر ساختار فایل ایجاد شده در این پروژه انداخته و سپس مازول‌های کلیدی برنامه را مورد بررسی قرار می‌دهیم.

ساختار فایل

در این پروژه، سه بسته^۱ مختلف به شرح زیر ایجاد شده است.

۱. بسته data_utils

این بسته شامل یک مازول به نام data_helper است که امکاناتی را برای سهولت در کار با مجموعه‌داده مورد استفاده فراهم می‌نماید. این مازول تنها یک کلاس به نام COCOHelper را در خود جای داده که تمامی توابع مورد نیاز برای کار با مجموعه‌داده MSCOCO را فراهم می‌نماید.

۲. بسته main.neuralnetworks.cnn

این بسته شامل یک مازول به نام CNN و یک بسته داخلی دیگر به نام inception است. وظیفه این بسته، فراهم‌کردن امکانات مربوط به رمزگذار پروژه می‌باشد. از آن‌جا که ممکن است در آینده به بیش از یک CNN مدل به عنوان رمزگذار نیاز شود، در این بسته از الگوی طراحی^۲ نما^۳ استفاده شده است. کلاس نقش نما را در این بسته ایفا می‌نماید. بسته داخلی inception، مدل از پیش آموزش دیده شده Google Inception V3 را دانلود کرده و مورد استفاده قرار می‌دهد.

۳. بسته main.neuralnetworks.rnn

این بسته شامل مازول‌های زیر می‌باشد:

(آ) مازول StackedRNN

این مازول شامل یک کلاس با همین نام به منظور پیاده‌سازی شبکه رمزگشا در پروژه است.

(ب) مازول RNNUtils

این مازول شامل دو کلاس است. کلاس اول با نام RNNUtils وظیفه فراهم‌سازی برخی از متدهای مرتبط با شبکه رمزگذار را بر عهده دارد. کلاس دیگری که در این مازول ایجاد شده است، کلاسی با

^۱Package

^۲Design Pattern

^۳Facade

نام RNNOptions است که شامل تمامی ورودی‌های و تنظیمات زمان اجرای مورد نیاز برای کار با شبکه رمزگشایی باشد.

کد اصلی پروژه در یک فایل پایتون با نام image_caption_train.py در ریشه سیستم فایل، قرار دارد. در بخش‌های بعدی به بررسی کد مربوط به هر یک از این بخش‌ها می‌پردازیم.

بررسی کدهای پروژه

فایل image_caption_train.py

در ابتدای این فایل، کتابخانه‌ها و ماثوله‌ای مورد استفاده، معرفی شده‌اند. برای این کار از قطعه کد زیر استفاده شده است.

```
import tensorflow as tf
import numpy as np
import nltk
import os
import time
import resource
import math
import copy
import re
import logging

from data_utils import data_helper
from main.neuralNetworks.cnn import CNN as Encoder
from main.neuralNetworks.rnn import StackedRNN as Decoder
from main.neuralNetworks.rnn import RNNUtils as DecoderUtils

from matplotlib import pyplot as plt
from multiprocessing import Process
```

ساختار این پروژه به شکلی است که داده‌های موجود در مجموعه‌داده به صورت دسته‌ای^۴ در حافظه بارگذاری می‌شوند. بارگذاری داده‌ها در حافظه، توسط CPU انجام می‌شود. وقتی یک دسته از داده‌ها در حافظه بارگذاری شد، پردازنده گرافیکی می‌تواند این دسته از داده‌ها را پردازش نموده و شبکه را آموزش دهد. در زمانی که پردازنده گرافیکی در حال پردازش داده‌های بارگذاری شده و آموزش شبکه است، CPU غیرفعال است.

⁴batch

برای بهبود کارایی سیستم، فرایند بارگذاری داده‌ها و آموزش شبکه، در این پروژه به صورت موازی انجام شده است. به همین منظور، توابعی برای بارگذاری یک دسته از داده‌ها و آموزش شبکه توسط یک دسته داده بارگذاری شده، به طور جداگانه نوشته شده‌اند. این توابع، در فرایندهای^۴ جداگانه و به طور موازی با هم اجرا می‌شوند. در ادامه کدهای مربوط به بارگذاری یک دسته از داده‌ها را نمایش داده‌ایم.

```
def getNextBatch(batchId, rawCaptions, cocoHelper, rnnOptions, vocab, word2ind):
    batchInstanceCount = min(rnnOptions.batch_insts, len(rawCaptions))
    batchData = np.zeros(shape=(rnnOptions.time_step,
                                cocoHelper.word2vec.layer1_size, rnnOptions.batch_size))
    batchLabel = np.zeros(shape=(rnnOptions.time_step,
                                rnnOptions.word_embedding_size, rnnOptions.batch_size))
    batchImg = []
    for i in range(batchInstanceCount - 1):
        embeddedCaptionInsts, instImgFileNames, embeddedLabels =
            getNextInstance(i + batchId, rawCaptions, cocoHelper,
                            rnnOptions.rnn_utils, vocab, word2ind=word2ind)
        cnt = 0
        offset = i * 5
        for embeddedCaptionInst in embeddedCaptionInsts:
            batchData[1:embeddedCaptionInst.shape[0], :, offset + cnt] =
                embeddedCaptionInst[0:min(rnnOptions.time_step,
                                           embeddedCaptionInst.shape[0]) - 1]
            cnt += 1
        cnt = 0
        for embeddedLabel in embeddedLabels:
            batchLabel[0:embeddedLabel.shape[0], :, offset + cnt] =
                embeddedLabel[0:rnnOptions.time_step]
            cnt += 1
        for instImgFileName in instImgFileNames:
            batchImg.append(instImgFileName)
    return batchData, batchImg, batchLabel
```

^۴Process

```

def getNextInstance(iteration, data, cocoHelper, rnnUtils, vocab, word2ind):
    global testLabel

    inst_image = [cocoHelper.imgs[i] for i in cocoHelper.imgs][iteration % len(data)]
    dataInsts = [cocoHelper.anns[i] for i in cocoHelper.anns if cocoHelper.anns[i]
                ["image_id"] == inst_image["id"]]
    embeddedCaptions = [rnnUtils.embed_inst_to_vocab(dataInst=
        dataInsts[i], cocoHelper=cocoHelper) for i in range(len(dataInsts))]
    embeddedLabels = [rnnUtils.embed_inst_to_vocab(dataInst=dataInsts[i],
        cocoHelper=cocoHelper) for i in range(len(dataInsts))]
    inst_image_filenames = [inst_image["file_name"] for _ in range(len(dataInsts))]
    testLabel.append(dataInsts)

    return embeddedCaptions, inst_image_filenames, embeddedLabels

def extractNextBatch(batchId, capWord2Ind, captionsDict, cocoHelper, rawCaptions,
                     rnnOptions, validation_mode=False):
    global batch_data_buffer, batch_img_filename_buffer, batch_label_buffer
    global test_batch_data_buffer, test_batch_img_filename_buffer
    global test_batch_label_buffer

    batch_data, batch_img_filename, batch_label = getNextBatch(batchId=batchId,
        rawCaptions=rawCaptions,
        cocoHelper=cocoHelper,
        rnnOptions=rnnOptions,
        vocab=captionsDict,
        word2ind=capWord2Ind)

    if not validation_mode:
        batch_data_buffer = batch_data
        batch_img_filename_buffer = batch_img_filename
        batch_label_buffer = batch_label
    else:
        test_batch_data_buffer = batch_data
        test_batch_img_filename_buffer = batch_img_filename

```

```
test_batch_label_buffer = batch_label
```

همین‌طور، تابع مربوط به آماده‌سازی داده‌ها و آموزش شبکه به صورت زیر نوشته شده است.

```
def prepare_data_and_train_structure(batchData, i, batchImgFileName, cnn,
                                     data_dir, imageFeaturesSize, rnn, rnnOptions, batchLabelRaw):
    global costs
    batchInput = np.zeros(shape=(batchData.shape[0], batchData.shape[2],
                                 batchData.shape[1]))
    batchLabel = np.zeros(shape=(batchLabelRaw.shape[0], batchLabelRaw.shape[2],
                                 batchLabelRaw.shape[1]))
    batchImageFeatures = np.zeros(shape=(batchData.shape[2],
                                         rnnOptions.image_feature_size[0], rnnOptions.image_feature_size[1]))
    for batchCnt in range(len(batchImgFileName)):
        imageFeatures = cnn.extract_features(os.path.join(
            data_dir, "train2014/" + batchImgFileName[batchCnt]))
        imageFeatures = imageFeatures.reshape(-1, rnnOptions.image_feature_size[-1])
        batchInput[:, batchCnt, :] = batchData[:, :, batchCnt]
        batchImageFeatures[batchCnt, :, :] = imageFeatures
        batchLabel[0:batchLabel.shape[0] - 1, batchCnt, :] =
            [batchLabelRaw[i + 1, :, batchCnt] for i in
             range(batchLabelRaw.shape[0] - 1)]
        costs[i] = rnn.train_batch(Xbatch=batchInput,
                                   annotationsBatch=batchImageFeatures, Ybatch=batchLabel, keep_prob=0.5)
```

تابع اصلی پروژه که فراخوانی و انجام فعالیت‌های اصلی بر عهده آن است، به شکل زیر نوشته شده است. این تابع، اولین تابعی است که شروع به اجرا شدن می‌نماید. خطوط ابتدایی این تابع، عملیات تنظیم محدودیت‌های سیستمی را بر عهده دارند.

```
def start():
    global testLabel
    testLabel = []
    rsrc = resource.RLIMIT_DATA
    soft, hard = resource.getrlimit(rsrc)
```

```

print('Soft limit starts as :', soft)

resource.setrlimit(rsrcc, (32 * 1024 * 1024 * 1024, hard)) # limit to one kilobyte

# set logger to print logs in console

logging.getLogger().addHandler(logging.StreamHandler())

soft, hard = resource.getrlimit(rsrcc)

print('Soft limit changed to :', soft)

plt.interactive(True)

print("defining constants")

data_dir, imageFeaturesSize, rnnUtils, maxIterCount =
initializeParameters()

print("loading dataset")

cocoHelper, captionsDict, rawCaptions, capWord2Ind, capInd2Word =
loadDataset(data_dir, rnnUtils)

test_cocoHelper, test_captionsDict, test_rawCaptions, test_capWord2Ind,
test_capInd2Word = loadTestDataset(data_dir, rnnUtils)

print("creating encoder's structure")

cnn = createEncoder(data_dir)

print("creating decoder's structure and initialization")

rnn, rnnOptions = createAndInitializeDecoder(captionsDict, imageFeaturesSize,
rnnUtils, word_embedding_size=cocoHelper.word2vec.layer1_size, cnn=cnn)

printGraph(rnnOptions, cnn=cnn.net.graph)

extractNextBatch(0, capWord2Ind, captionsDict, cocoHelper, rawCaptions, rnnOptions)

batchData = copy.copy(batch_data_buffer)

batchImgFileName = list(batch_img_filename_buffer)

batchLabelRaw = copy.copy(batch_label_buffer)

batchInput = np.zeros(shape=(batchData.shape[0], batchData.shape[2],
batchData.shape[1]))

batchLabel = np.zeros(shape=(batchLabelRaw.shape[0], batchLabelRaw.shape[2],
batchLabelRaw.shape[1]))

batchImageFeatures = np.zeros(shape=(batchData.shape[2],
rnnOptions.image_feature_size[0], rnnOptions.image_feature_size[1]))

for batchCnt in range(len(batchImgFileName)):

```

```

imageFeatures = cnn.extract_features(os.path.join(data_dir, "train2014/" +
+ batchImgFileName[batchCnt]))
imageFeatures = imageFeatures.reshape(-1, rnnOptions.image_feature_size[-1])
batchInput[:, batchCnt, :] = batchData[:, :, batchCnt]
batchImageFeatures[batchCnt, :, :] = imageFeatures
batchLabel[0:batchLabel.shape[0] - 1, batchCnt, :] = [
batchLabelRaw[i + 1, :, batchCnt] for i in
range(batchLabelRaw.shape[0] - 1)
]
print("test image name: ", batchImgFileName[0])
print("starting to train the structure")
global costs
startTime = time.time()
costs = np.zeros(maxIterCount)
checkPoint = 1
statShowPeriod = 1

batchId = 0
for i in range(maxIterCount):
    loop_counter = math.ceil((len(rawCaptions) / 5) / rnnOptions.batch_size)
    loop_counter = min(loop_counter, 10)
    for internal_loop_counter in range(loop_counter):
        print("iteration:" + repr(i) + ", internal loop: processing",
100 * internal_loop_counter / loop_counter,
"%")
        batchData = copy.copy(batch_data_buffer)
        batchImgFileName = list(batch_img_filename_buffer)
        batchLabelRaw = copy.copy(batch_label_buffer)
        load_data_process = Process(target=extractNextBatch, args=(batchId,
capWord2Ind, captionsDict, cocoHelper, rawCaptions, rnnOptions))
        load_data_process.start()
        prepare_data_and_train_structure(batchData=batchData, i=i,

```

```

batchImgFileName=batchImgFileName, cnn=cnn,
data_dir=data_dir, imageFeaturesSize=imageFeaturesSize, rnn=rnn,
rnnOptions=rnnOptions, batchLabelRaw=batchLabelRaw)
load_data_process.join()
batchId += 1
endTime = time.time()
duration = (endTime - startTime) / statShowPeriod
print("batch ", i, ", train time per batch: ", duration, "
current gained cost: ", costs[i])
startTime = endTime
print("drawing error curve...")
if i % statShowPeriod == 0 and i > 0:
    plt.plot([i - 1, i], costs[i - 1:i + 1])
    plt.show()
    plt.pause(1)
if i % checkPoint == 0:
    print("saving current model...")
    rnnOptions.saver.save(rnnOptions.session,
    rnnOptions.saved_model_path)
    print("testing current model:")
    testModel(rnn, rnnOptions, test_cocoHelper,
    test_capInd2Word, test_capWord2Ind, test_captionsDict,
    test_rawCaptions, cnn, data_dir)

```

تابع `test_model` تابعی است که در آن، مدل آموزش داده شده، برای تولید شرح متناظر یک دسته تصویر، مورد استفاده قرار می‌گیرد. این تابع به شکل زیر نوشته شده است.

```

def testModel(rnn, rnnOptions, cocoHelper, ind2word, word2ind, vocab,
             raw_captions, cnn, data_dir):
    global testLabel
    testLabel = []
    extractNextBatch(0, capWord2Ind=word2ind, captionsDict=vocab,
                    cocoHelper=cocoHelper, rawCaptions=raw_captions,
                    rnnOptions=rnnOptions, validation_mode=True)

```

```

batchData = copy.copy(test_batch_data_buffer)
batchImgFileName = list(test_batch_img_filename_buffer)
batchLabelRaw = copy.copy(test_batch_label_buffer)
batchInput = np.zeros(shape=(batchData.shape[0], batchData.shape[2],
batchData.shape[1]))
batchLabel = np.zeros(shape=(batchLabelRaw.shape[0], batchLabelRaw.shape[2],
batchLabelRaw.shape[1]))
batchImageFeatures = np.zeros(shape=(batchData.shape[2],
rnnOptions.image_feature_size[0], rnnOptions.image_feature_size[1]))
for batchCnt in range(len(batchImgFileName)):
    imageFeatures = cnn.extract_features(os.path.join(data_dir, "val2014/" +
+ batchImgFileName[batchCnt]))
    imageFeatures = imageFeatures.reshape(-1, rnnOptions.image_feature_size[-1])
    batchInput[:, batchCnt, :] = batchData[:, :, batchCnt]
    batchImageFeatures[batchCnt, :, :] = imageFeatures
    batchLabel[0:batchLabel.shape[0] - 1, batchCnt, :] = [
        batchLabelRaw[i + 1, :, batchCnt] for i in
        range(batchLabelRaw.shape[0] - 1)
    ]
predicted_captions = ["" for _ in range(rnnOptions.batch_insts)]
BLEUScores = np.zeros(shape=rnnOptions.batch_insts)
test_input = np.zeros(shape=(batchInput.shape[0], 1, batchInput.shape[2]))
test_image_features = np.zeros(shape=(1, batchImageFeatures[1],
batchImageFeatures[2]))
for batch_inst in range(rnnOptions.batch_insts - 1):
    i = batch_inst * 5
    test_input[:, 0, :] = batchInput[:, i, :]
    out = rnn.run_step(X=test_input, annotations=batchImageFeatures[i, :, :],
init_zero_state=True)[0][0]
    for testInd in range(rnnOptions.time_step - 1):
        new_word = cocoHelper.word2vec.most_similar(positive=[out], topn=1)[0][0]
        predicted_captions[batch_inst] += " " + new_word

```

```

batchInput[testInd + 1, i, 0:cocoHelper.word2vec.layer1_size] =
cocoHelper.word2vec[new_word]

test_input[:, 0, :] = batchInput[:, i, :]

out = rnn.run_step(X=test_input, init_zero_state=False)[testInd + 1]
out = out[0]

BLEUScores[batch_inst] = nltk.translate.bleu_score.sentence_bleu(
[ref["caption"] for ref in testLabel[batch_inst]],
predicted_captions[batch_inst], emulate_multibleu=True)

print("prediction: ", i, ":", predicted_captions[batch_inst])
print("human label: ", testLabel[batch_inst])
print("BLEU SCORE: ", BLEUScores[batch_inst])

```

در ادامه، توابع دیگری که در این فایل مورد استفاده قرار گرفته‌اند، آورده شده‌اند.

```

def printGraph(rnnOptions, cnn):
    print("writing graphs to /tmp/graph in order to show it in tensor board")
    writer = tf.summary.FileWriter("/tmp/graph")
    writer.add_graph(rnnOptions.session.graph)
    writer.add_graph(cnn)

def createAndInitializeDecoder(captionsDict, imageFeaturesSize, rnnUtils,
                               word_embedding_size, cnn):
    rnnOptions = DecoderUtils.RNNOptions(vocab=captionsDict, rnnUtils=rnnUtils,
                                         image_feature_size=imageFeaturesSize,
                                         word_embedding_size=word_embedding_size)
    rnn = Decoder.StackedRNN(input_size=rnnOptions.input_size, image_feature_size=
        rnnOptions.image_feature_size,
        lstm_size=rnnOptions.lstm_size, number_of_layers=rnnOptions.num_layers,
        output_size=rnnOptions.out_size, session=rnnOptions.session,
        learning_rate=rnnOptions.learning_rate, batch_size=rnnOptions.batch_size,
        name=rnnOptions.name, cnn_graph=cnn.net.session.graph)
    rnnOptions.session.run(tf.global_variables_initializer())
    rnnOptions.saver = tf.train.Saver(tf.global_variables())

```

```

return rnn, rnnOptions

def createEncoder(data_dir):
    cnn = Encoder.CNN(os.path.join(data_dir, "train2014"))
    return cnn

def loadDataset(data_dir, rnnUtils):
    cocoHelper = data_helper.COCOHelper(data_dir +
                                         "annotations/captions_train2014.json")
    rawCaptions, captionsDict, capIndToWord, capWordToInd, capIndToWord =
        cocoHelper.extract_captions()
    sentenceSize=100,
    return cocoHelper, captionsDict, rawCaptions, capWordToInd, capIndToWord

def loadTestDataset(data_dir, rnnUtils):
    cocoHelper = data_helper.COCOHelper(data_dir + "annotations/captions_val2014.json")
    rawCaptions, captionsDict, capIndToWord, capWordToInd, capIndToWord =
        cocoHelper.extract_captions()
    return cocoHelper, captionsDict, rawCaptions, capWordToInd, capIndToWord

def initializeParameters():
    rnnUtils = DecoderUtils.RNNUtils()
    data_dir = "../..../Dataset/MS-COCO/"
    imageFeaturesSize = [64, 2048]
    maxIterCount = 20000
    return data_dir, imageFeaturesSize, rnnUtils, maxIterCount

```

در انتهای این فایل، با فراخوانی تابع `start()`، برنامه آغاز به کار می‌نماید.

StackedRNN.py فایل

```

import tensorflow as tf
import numpy as np

from tensorflow.contrib.seq2seq import BahdanauAttention
from main.neuralNetworks.rnn.GuidedAttentionRNNCell import GuidedAttentionCell


class StackedRNN:

    def __init__(self, input_size, image_feature_size, lstm_size, number_of_layers,
                 output_size, session, learning_rate, batch_size, attn_length=0,
                 attn_size=0, attn_mechanism="bahdanau", name="rnn", cnn_graph=None):
        self.scope = name
        self.cnn_graph = cnn_graph
        self.attn_length = attn_length
        self.attn_size = attn_size
        self.attn_mechanism = attn_mechanism
        self.input_size = input_size
        self.image_feature_size = image_feature_size
        self.batch_size = batch_size
        print("input size:", input_size)
        self.lstm_size = lstm_size
        self.number_of_layers = number_of_layers
        self.output_size = output_size
        self.session = session
        self.learning_rate = tf.constant(learning_rate)
        self.lstm_last_state = np.zeros(
            shape=(self.number_of_layers * 2 * self.lstm_size,))
    )

with tf.device('/device:GPU:0'):
    with tf.variable_scope(name_or_scope=self.scope):

```

```

with tf.variable_scope(name_or_scope="Input"):
    self.X = tf.placeholder(dtype=tf.float32, shape=(None, None,
                                                    self.input_size), name="X")
    self.annotations = tf.placeholder(dtype=tf.float32, shape=(None,
                                                               self.image_feature_size[0],
                                                               self.image_feature_size[1]),
                                       name="annotations")

with tf.variable_scope(name_or_scope="DropOutParams"):
    self.keep_prob = tf.placeholder(dtype=tf.float32)

with tf.name_scope(name="attentionMechanism"):
    if self.attn_mechanism == "bahdanau":
        self.attention_mechanism = attention.BahdanauAttention(
            num_units=self.attn_length,
            memory=,
            memory_sequence_length=)

    elif self.attn_mechanism == "luong":
        self.attention_mechanism = attention.LuongAttention(
            num_units=self.attn_length,
            memory=,
            memory_sequence_length=)

    else:
        raise ValueError(
            "attention mechanism '%s' is not defined." % self.attn_mechanism)

with tf.name_scope(name="RNN_Core"):
    self.lstm_cells = [GuidedAttentionCell(self.lstm_size, forget_bias=1.0,
                                           state_is_tuple=True)
                      for _ in
                      range(self.number_of_layers)]
    self.lstm_cells = [tf.nn.rnn_cell.DropoutWrapper(lstm_cell,
                                                     output_keep_prob=self.keep_prob)
    
```

```

                for lstm_cell in self.lstm_cells]

    self.lstm = tf.nn.rnn_cell.MultiRNNCell(self.lstm_cells,
                                            state_is_tuple=True)

    self.lstm = tf.nn.rnn_cell.DropoutWrapper(self.lstm,
                                              output_keep_prob=self.keep_prob)

    self.lstm_init_states = self.lstm.zero_state(
        batch_size=tf.shape(self.X)[1], dtype=tf.float32)

    self.outputs, self.lstm_current_state = tf.nn.dynamic_rnn(
        cell=self.lstm, inputs=self.X,
        dtype=tf.float32, time_major=True,
        initial_state=self.lstm_init_states)

with tf.variable_scope(name_or_scope="Output"):

    self.OUT_W = tf.Variable(initial_value=
        tf.random_normal(shape=(self.lstm_size, self.output_size),
                         stddev=0.01, name="output_W"))

    self.OUT_B = tf.Variable(initial_value=
        tf.random_normal(shape=(self.output_size,), ,
                         stddev=0.01, name="output_B"))

self.outputs_reshaped = tf.reshape(tensor=self.outputs,
                                   shape=[-1, self.lstm_size])

self.net_out = tf.nn.batch_normalization(x=tf.matmul(
    self.outputs_reshaped, self.OUT_W) + self.OUT_B,
                                         mean=0, variance=1, offset=0, scale=1,
                                         variance_epsilon=1e-10)

self.batch_time_shape = tf.shape(self.outputs)
self.final_output = tf.reshape(self.net_out,
                               shape=(self.batch_time_shape[0],
                                      self.batch_time_shape[1],
                                      self.output_size))

```

```

self.Y = tf.placeholder(dtype=tf.float32,
                        shape=(None, None, self.output_size))

self.Y_long = tf.reshape(tensor=self.Y, shape=(-1, self.output_size))

self.cost = tf.reduce_sum(tf.losses.absolute_difference(
    predictions=self.softmax_net_out,
    labels=self.Y_long))

self.train_op = tf.train.AdamOptimizer(
    learning_rate=self.learning_rate).minimize(self.cost)

self.print_number_of_parameters()

def run_step(self, X, annotations=None, init_zero_state=True, keep_prob=1):
    if init_zero_state:
        init_value = np.zeros(shape=(self.number_of_layers * 2 * self.lstm_size,))
    else:
        init_value = self.lstm_last_state

    out, next_lstm_state = self.session.run([self.final_output, self.lstm_current_st
                                             feed_dict={self.X: X,
                                                       self.annotations: annotations,
                                                       self.keep_prob: keep_prob})

    self.lstm_last_state = next_lstm_state[len(next_lstm_state) - 1]

    return out

def train_batch(self, Xbatch, annotationsBatch, Ybatch, keep_prob=1):
    init_value = np.zeros(shape=(Xbatch.shape[0],
                                 self.number_of_layers * 2 * self.lstm_size))

```

```

cost, _ = self.session.run([self.cost, self.train_op],
                           feed_dict={self.X: Xbatch, self.Y: Ybatch,
                                      self.annotations: annotationsBatch,
                                      self.keep_prob: keep_prob})

return cost

@staticmethod
def print_number_of_parameters():
    total_parameters = 0
    for variable in tf.trainable_variables():
        shape = variable.get_shape()
        variable_parameters = 1
        for dim in shape:
            variable_parameters *= dim.value
        total_parameters += variable_parameters
    print("total number of parameters: ", total_parameters)

```

فایل GuidedAttentionRNNCell.py

مازول GuidedAttentionRNNCell دارای یک کلاس به همان نام است. این کلاس، واحد اصلی شبکه را پیاده‌سازی می‌نماید. هر واحد شبکه دیکودر، یک سلو LSTM است، که مکانیزم توجه بصری ارائه شده را پیاده‌سازی می‌نماید. برای استفاده از امکانات موجود در کلاس BasicLSTMCell در تنسورفلو، کلاس GuidedAttentionRNNCell را به گونه‌ای ایجاد نموده‌ایم که از آن ارثبری نماید.

```

import tensorflow as tf
from tensorflow.contrib.seq2seq.python.ops.attention_wrapper import AttentionWrapper

class GuidedAttentionCell(tf.nn.rnn_cell.BasicLSTMCell):
    def __init__(self, num_units, forget_bias=1.0,
                 state_is_tuple=True, activation=None, reuse=None):
        super(GuidedAttentionCell, self).__init__(num_units=num_units,
                                                forget_bias=forget_bias,

```

```

state_is_tuple=state_is_tuple,
activation=activation,
reuse=reuse)

def __call__(self, inputs, state, scope=None):
    return super(GuidedAttentionCell, self).call(inputs=inputs, state=state)

```

آموزش مدل جاسازی کلمات

برای آموزش جاسازی کلمات از کتابخانه Gensim استفاده نموده‌ایم. این کتابخانه، جاسازی کلمات را مطابق با پژوهشی که در فصل سوم، به تفصیل مورد بررسی قرار گرفت، پیاده‌سازی نموده است. برای ایجاد جاسازی‌های مناسب برای کلمات، مدل مورد استفاده، تا جایی که خطای جاسازی به کمتر از ۵۴۰۰۰ برسد، روی مجموعه کلمات، آموزش داده می‌شود. برای کنترل برنامه در حالتی که خطای جاسازی کلمات، کمتر از مقدار موردنظر نشود، سقف ۱۰۰۰۰۰ تکرار، برای خاتمه آموزش، در نظر گرفته شده است. تابع create_word2vec، که کد آن در ادامه این بخش، آورده شده است، یکی از متدهای کلاس DataHelper است که در زمان ایجاد نمونه جدید و در انتهای Constructor این کلاس، فراخوانی می‌شود.

```

def create_word2vec(self):
    print("creating word embeddings structure")
    sentences = [re.split("[\W]+", (self.anns[i]["caption"]).lower()) for i in self.anns]
    iteration_count = 1
    self.word2vec = Word2Vec(iterator=iteration_count, min_count=0, size=512, workers=8)
    self.word2vec_vocab = self.word2vec.build_vocab(sentences=sentences)
    print("training word embeddings")
    loss_value = 1000000
    i = 0
    while loss_value >= 540000 and i < iteration_count:
        self.word2vec.train(sentences=sentences, total_examples=len(sentences), epochs=1,
                             compute_loss=True)
        loss_value = self.word2vec.get_latest_training_loss()
        print("iteration:", i, ", loss value:", loss_value)
        i += 1

```

واژه‌نامه‌ی فارسی به انگلیسی

Visual Attention	توجه بصری	آ
Embedding	جاسازی	ج
Framework	چارچوب کاری	ج
Batch	دسته	د
Precision	دقت	
Bidirectional	دوطرفه	ر
Encoder	رمزگذار	
Decoder	رمزگشای	
Process	فرایند	ف
Meaning Space	فضای معنایی	
Hyperparameter	فوق پارامتر	
	Detector	آ
	Supersegment	ابرقطعه
	Max Aposteriori	احتمال بیشینه پسین
	Probability	بروگردان
	Confidence	اطمینان
	Validation	اعتبارسنجی
	Design Pattern	الگوی طراحی
	Transfer Learning	انتقال یادگیری
	One-Hot Vector	بردار تکفعال
	Package	بسته
	Optimizer	بهینه‌ساز
	Recall	پوشش
		ت

قطعه‌بندی قطعه‌بندی

م

متحرک متحرک

Latent مخفی

ن

Facade نما

ه

Alignment هم‌ترازسازی

واژه‌نامه‌ی انگلیسی به فارسی

A

Alignment هم‌ترازسازی

Animated متحرک

Framework چارچوب کاری

H

Hyperparameter فوق پارامتر

B

Batch دسته

Bidirectional دوطرفه

L

Latent مخفی

C

Confidence اطمینان

Max Aposteriori احتمال بیشینه پسین
Probability

D

Decoder رمزگشای

Design Pattern الگوی طراحی

Detector آشکارکننده

Meaning Space فضای معنایی

O

One-Hot Vector بردار تک‌فعال

Optimizer بهینه‌ساز

E

Embedding جاسازی

Encoder رمزگذار

P

Package بسته

Precision دقیق

F

Facade نما

Process فرایند

R

Recall	پوشش	Transfer Learning	انتقال یادگیری
S		V	
Segmentation	قطعه‌بندی		اعتبارسنجی
Supersegment	ابرقطعه	Validation	
T		Visual Attention	توجه بصری

Abstract

This page is accurate translation from Persian abstract into English.

Key Words:

Write a 3 to 5 KeyWords is essential. Example: AUT, M.Sc., Ph. D.,..



**Amirkabir University of Technology
(Tehran Polytechnic)**

Department of ...

MSc Thesis

Title of Thesis

By

Name Surname

Supervisor

Dr.

Advisor

Dr.

Month & Year