بسم الله الرحمن الرحيم

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد مهندسی هوش مصنوعی

# دسته‌بندی سؤالات کاربر در یک سیستم مکالمه خودکار با استفاده از روش‌های هوش مصنوعی

نگارنده: غزاله مقدم‌نژاد

استاد راهنما:

دکتر مرتضی زاهدی

بهمن‌ماه ۱۳۹۶

**این تحقیق علمی را کوچک تر از آن میبینم که به پدر و مادرم تقدیم کنم، پس آن را فقط به یک لبخند مهربان پدرم و یک نگاه نگران مادرم تقدیم میکنم...**

# تشکر و قدردانی

پروردگار مهربان را سپاس می‌گویم به خاطر تمامی نعمت‌هایی که به من داده، سپاس بی کران از استاد گرامی، **جناب آقای دکتر مرتضی زاهدی** که این پژوهش بدون راهنمایی‌های ارزنده ایشان راه به جایی نداشت.

بر خود لازم می‌بینم تا از **جناب آقای مهندس حسن طباطبایی** و **جناب آقای مهندس محمد مهدی حسینی** که در انجام این پژوهش از هیچ کمکی بر اینجانب دریغ نکردند، کمال تشکر و قدردانی را داشته باشم. از دوستان خوبم، خانوم‌ها **مولود آیت**، **فاطمه زکی‌زاده**، **سحر عموئی** و **پریسا انتظامی** که در تمامی مراحل این پژوهش یار و همراهم بودند، نهایت سپاس‌گزاری را دارم.

# تعهد نامه

اینجانب **غزاله مقدم‌نژاد** دانشجوی دوره کارشناسی ارشد رشته هوش مصنوعی دانشکده کامپیوتر و فناوری اطلاعات دانشگاه صنعتی شاهرود نویسنده پایان‌نامه **دسته‌بندی سؤالات کاربر در یک سیستم مکالمه خودکار با استفاده از روش‌های هوش مصنوعی** تحت راهنمایی **آقای دکتر مرتضی زاهدی** متعهد می‌شوم.

- تحقیقات در این پایان‌نامه توسط اینجانب انجام‌شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورداستفاده استناد شده است.
- مطالب مندرج در پایان‌نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود است و مقالات مستخرج با نام «دانشگاه صنعتی شاهرود» و یا «Shahrood University of Technology» به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که از موجود زنده (یا بافتهای آن‌ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری، ضوابط و اصول اخلاق انسانی رعایت شده است.

**امضای دانشجو**

**تاریخ**

## مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود است. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در پایان‌نامه بدون ذکر مرجع مجاز نمی‌باشد.

## چکیده

با افزایش روزافزون حجم اطلاعات ذخیره شده در منابع قابل دسترس، فرآیند بازیابی و استخراج اطلاعات اهمیت ویژه‌ای یافته است. سیستم‌های پرسش و پاسخ گونه‌ای از سیستم‌های بازیابی اطلاعات هستند که این امکان را می‌دهند تا کاربر پاسخ پرسش خود را به صورت مفید و کوتاه دریافت کند. سیستم‌های پرسش و پاسخ تعاملی با برقراری تعامل دو سویه با کاربر، ابهامات احتمالی موجود در پرسش را رفع نموده و دقت پاسخگویی را افزایش می‌دهند.

پس از دریافت پرسش کاربر، یکی از مراحل تحلیل پرسش، دسته‌بندی آن است. تاکنون روش‌های مختلف یادگیری ماشین برای استخراج ویژگی و طبقه‌بندی سؤالات مطرح شده است. از سوی دیگر، در حال حاضر یادگیری عمیق به عنوان ابزاری نوین و کارآمد، کیفیت یادگیری را نسبت به روش‌های پیشین بهبود بخشیده است. ما در این پژوهش با بهره‌گیری از ساختاری که از ترکیب دو روش یادگیری عمیق استفاده می‌کند، راهکاری جدید برای استخراج ویژگی‌های مناسب‌تر و دسته‌بندی سؤالات ارائه داده‌ایم.

در این روش، برای بازنمایی کلمات از شبکهٔ آموزش داده شدهٔ word2vec استفاده شده است. همچنین استخراج ویژگی و دسته‌بندی سؤالات در قالب یک شبکهٔ عمیق که ترکیب دو شبکهٔ کانولوشن و LSTM است، انجام می‌گیرد. شبکهٔ طراحی شده می‌تواند در صورت کافی نبودن میزان دقت پاسخ خود، از جملات قبلی موجود در مکالمه استفاده کند. آزمایشات انجام شده نشان می‌دهد، روش پیشنهادی ما منجر به بهبود ۴٫۲ درصدی روی مجموعه دادگان جمع‌آوری شده، نسبت به سایر روش‌های یادگیری ماشین اشاره شده، می‌شود.



**واژه‌های کلیدی**: بازیابی اطلاعات، سیستم پرسش و پاسخ تعاملی، طبقه‌بندی سؤالات، شبکه عمیق

# فهرست مطالب

## فصل ۱: مقدمه


مقدمه ..... ۲

۱-۱- سیستم‌های پرسش‌وپاسخ ..... ۲

۱-۲- تاریخچهٔ سیستم‌های پرسش‌وپاسخ ..... ۴

۱-۳- طبقه‌بندی سؤالات در سیستم‌های پرسش‌وپاسخ ..... ۸

۱-۴- هدف پژوهش ..... ۹

۱-۵- ساختار پایاننامه ..... ۹


## فصل ۲: پیشینه تحقیق


مقدمه ..... ۱۲

۲-۱- تعریف مسأله ..... ۱۲

۲-۲- ویژگی‌های مهم در طبقه‌بندی سؤالات ..... ۱۲

۲-۳-۱- مدل کیسه‌ای از کلمات ..... ۱۲

۲-۳-۲- مدل N-گرم ..... ۱۲

۲-۳-۳- مدل فضای برداری ..... ۱۴

۲-۳-۴- مدل احتمالاتی ..... ۱۵

۲-۳-۵- تحلیل ساختاری، گرامری و معنایی ..... ۱۶

۲-۳-۶- الگوریتم TF-IDF ..... ۱۸

۲-۳-۷- چارچوب کلمه به بردار ..... ۱۹

۲-۳- روش‌های مهم در طبقه‌بندی سؤالات ..... ۲۰

۲-۴-۱- دسته‌بندی مبتنی‌بر قوانین ..... ۲۱

۲-۴-۲- دسته‌بندی مبتنی‌بر یادگیری ماشین..... ۲۱
۲-۴- جمع‌بندی ..... ۳۲

## فصل ۳: یادگیری عمیق

۳-۱- مروری بر یادگیری عمیق ..... ۳۴
۳-۱-۱- یادگیری باناظر ..... ۳۶
۳-۱-۲- پس‌انتشار در معماری‌های چند لایه ..... ۳۸
۳-۱-۳- شبکه‌های کانولوشن..... ۴۲
۳-۱-۴- شبکه‌های عمیق بازگشتی..... ۴۵
۳-۱-۵- شبکه‌های LSTM ..... ۴۹
۳-۲- یادگیری عمیق در طبقه‌بندی سؤالات ..... ۵۵
۳-۳- جمع‌بندی ..... ۵۸

## فصل ۴: روش تحقیق

۴-۱- مقدمه ..... ۶۰
۴-۲- طرح مسأله ..... ۶۰
۴-۳- روش پیشنهادی ..... ۶۰
۴-۴- ارائه مدل کلی ..... ۶۱
۴-۴-۱- ماژول استخراج‌کننده ویژگی ..... ۶۲
۴-۴-۲- جزئیات شبکه عمیق ..... ۶۵
۴-۵- جمع‌بندی ..... ۶۸

## فصل ۵: نتایج و تحلیل آن‌ها

۵-۱- مقدمه ..... ۷۰
۵-۲- مجموعه دادگان مورد استفاده ..... ۷۰

۵-۳- پیاده‌سازی ..... ۷۲
۵-۳-۱- پیش‌پردازش دادگان ..... ۷۲
۵-۴- معرفی چارچوب TensorFlow ..... ۷۲
۵-۵- آزمون‌ها و پارامترهای مورد نظر ..... ۷۳
۵-۶- معیار ارزیابی ..... ۷۸
۵-۷- نتایج ..... ۷۹
۵-۷-۱- نتایج رده بندی ..... ۷۹
۵-۷-۲- نتایج بهبود با استفاده از دنبالهٔ مکالمه..... ۸۰
۵-۸- جمع‌بندی ..... ۸۲

## فصل ۶: نتیجه‌گیری

۶-۱- نتیجه‌گیری ..... ۸۴
۶-۲- کارهای آینده ..... ۸۵
مراجع ..... ۸۷

## فهرست اشکال


شکل (۲-۱) نمایش برداری یک درخواست و دو سند ..... ۱۵

شکل (۲-۲) یک نمونه درخت تجزیه جمله ..... ۱۷

شکل (۳-۱) مقایسه دو تابع فعالسازی با هم ..... ۳۹

شکل (۳-۲) شبکه‌های عصبی چندلایه و پس‌انتشار ..... ۴۰

شکل (۳-۳) شبکه‌های عصبی بازگشتی در ساختار خود دارای حلقه هستند. ..... ۴۳

شکل (۳-۴) شبکهٔ عصبی بازگشتی باز شده ..... ۴۵

شکل (۳-۵) شبکهٔ عصبی بازگشتی با یادگیری از اطلاعات گذشتهٔ نزدیک ..... ۴۶

شکل (۳-۶) شبکهٔ عصبی بازگشتی نیازمند یادگیری از اطلاعات گذشتهٔ دور ..... ۴۷

شکل (۳-۷) شبکه‌های عصبی بازگشتی استاندارد شامل یک لایه در هر ماژول هستند ..... ۴۸

شکل (۳-۸) ساختار کلی یک شبکهٔ عصبی بازگشتی LSTM ..... ۵۰

شکل (۳-۹) یک سلول حالت در شبکهٔ LSTM ..... ۵۰

شکل (۳-۱۰) نمایش یک دروازه در یک سلول از شبکهٔ عصبی LSTM ..... ۵۱

شکل (۳-۱۱) بخش حذف کننده اطلاعات از سلول حالت ..... ۵۲

شکل (۳-۱۲) بخش اضافه کردن اطلاعات جدید در سلول حالت ..... ۵۳

شکل (۳-۱۳) بروز شدن اطلاعات در سلول حالت ..... ۵۴

شکل (۳-۱۴) خروجی سلول حالت در شبکهٔ LSTM ..... ۵۴

شکل (۴-۱) ساختار کلی روش پیشنهادی ..... ۶۲

شکل (۴-۲) ساختار شبکه word2vec ..... ۶۳

شکل (۴-۳) بدست آوردن بردار ویژگی برای جملات ..... ۶۵

شکل (۴-۴) یک شبکه عادی LSTM ..... ۶۶

شکل (۴-۵) یک شبکه LSTM که روزنه ها ( به رنگ سیاه) به آن اضافه شده اند ..... ۶۷

شکل (۴-۶) شمای کلی شبکه ترکیبی cnn-lstm که در این پژوهش استفاده شده است ..... ۶۸

شکل (۵-۱) دو نمونه از مکالمات انتخاب شده از مجموعه دادگان ..... ۷۲

## فهرست جداول

جدول (۲-۱) نتایج دقت طبقه‌بند svm با هسته‌های مختلف روی مجموعه دادگان TREC.....۲۳

جدول (۲-۲) دقت ۵ طبقه‌بند مختلف با ویژگی کیسهٔ کلمات روی مجموعه دادگان TREC.....۳۱

جدول (۴-۱) مقایسه الگوریتم‌های بازنمایی word2vec با سایر روش‌های مدل‌سازی زبان.....۶۴

جدول (۵-۱) دسته‌بندی استاندارد برای طبقه‌بندی سؤالات در مجموعه‌دادگان TREC .....۷۰

جدول (۵-۲) کلاس‌های استفاده شده برای برچسب‌گذاری مجموعه دادگان .....۷۱

جدول (۵-۳) سؤال های مورد بررسی از مکالمه‌های انتخاب شده .....۷۴

جدول (۵-۴) امتیازات متعلق به هر کلاس برای سوال $Q_1$ توسط رده‌بند SVM.....۷۵

جدول (۵-۵) امتیازات متعلق به هر کلاس برای سوال $Q_2$ توسط رده‌بند SVM .....۷۵

جدول (۵-۶) امتیازات متعلق به هر کلاس برای سوال $Q_1$ با استفاده از ویژگی word2vec.....۷۶

جدول (۵-۷) امتیازات متعلق به هر کلاس برای سوال $Q_2$ با استفاده از ویژگی word2vec .....۷۶

جدول (۵-۸) امتیازات بدست آمده برای سوال $Q_2$ در دو شبکهٔ CNN-LSTM با ساختار ساده و تعاملی.....۷۷

جدول (۵-۹) نتایج استفاده از ویژگی‌های آماری بروی مجموعه دادگان .....۷۹

جدول (۵-۱۰) نتایج استفاده از شبکهٔ عصبی عمیق بروی مجموعه دادگان .....۷۹

جدول (۵-۱۱) مقایسه دقت روش CNN-LSTM معمولی و تعاملی ..... ۸۰

# فصل ۱:

## مقدمه

## ۱-۱- مقدمه

با افزایش روزافزون حجم اطلاعات ذخیره شده در منابع قابل دسترس و گوناگون، فرآیند بازیابی و استخراج اطلاعات اهمیت ویژه‌ای یافته است. اطلاعات مورد نظر ممکن است شامل هر نوع منبعی مانند متن، تصویر، صوت و ویدئو باشند. برخلاف پایگاه داده‌ها، اطلاعات ذخیره شده در منابع اطلاعاتی بزرگ مانند وب و زیرمجموعه‌های آن مانند شبکه‌های اجتماعی از ساختار مشخصی پیروی نمی‌کنند و عموماً دارای معانی تعریف شده و مشخصی نیستند. هدف بازیابی اطلاعات[1] در چنین شرایطی، کمک به کاربر برای یافتن اطلاعات مورد نظر در انبوهی از اطلاعات ساختارنیافته است.

بازیابی اطلاعات، علم جستجوی سند، جستجوی اطلاعات در سند، جستجوی فراداده‌هایی[2] که اسناد را توصیف می‌کنند و یا جستجو در پایگاه داده‌های رابطه‌ای یا پایگاه داده‌های شبکه‌ای نظیر اینترنت برای متن، صدا، تصویر یا داده است [۱].

## ۱-۲- سیستم‌های پرسش و پاسخ

در فرآیند بازیابی اطلاعات، موتورهای جستجو لیستی از اسناد مربوطه را در پاسخ به کلمات کلیدیِ فرموله شدهٔ کاربران ارائه می‌دهد. این جستجو براساس جنبه‌های مختلف مانند میزان محبوبیت، تطابق کلمات کلیدی، فرکانس دسترسی به اسناد و غیره ارائه است. با این حال، آن‌ها به درستی کار بازیابی اطلاعات را انجام نمی‌دهند، زیرا کاربران باید اسناد را یک به یک برای دریافت اطلاعات مورد نظر بررسی کنند [۳]. این امر باعث می‌شود که بازیابی اطلاعات یک فرایند وقت‌گیر باشد. به این منظور سیستم‌های پرسش و پاسخ (QA[3]) پدید آمدند که در حالت ایده‌آل، یک موتور جستجو باید تعدادی جملهٔ مختصر به عنوان پاسخ همراه با لینک‌های مرتبط خود را ارسال کند. تعداد زیادی از این سیستم‌ها از سال ۱۹۶۰ توسعه یافته است [۴].

[1] Information retrieval
[2] Meta data
[3] Question Answering

سیستم‌های پرسش و پاسخ تلاش می‌کنند تا سؤالات کاربران به زبان‌های طبیعی را پس از بازیابی و پردازش اطلاعات از منابع مختلف داده یا منابع وب معنایی، پاسخ دهند [۷]. این سیستم‌ها یک فرم پیچیده از بازیابی اطلاعات هستند که با اطلاعات مورد نیاز کاربر که به صورت عبارات یا سؤالات طبیعی بیان می‌شود، متمایز می‌گردند. سیستم‌های QA یکی از طبیعی‌ترین نوع ارتباط انسان با کامپیوتر هستند. در مقایسه با سیستم‌های بازیابی کلاسیک که در آن اسناد به صورت کامل در جواب درخواست اطلاعات داده می‌شوند، سیستم‌های QA بخش خاصی از اطلاعات را به عنوان پاسخ برمی‌گردانند. کاربران سیستم‌های QA علاقه‌مند به دریافت یک پاسخ مختصر، قابل درک و صحیح، بدون استفاده از کوئری‌ها یا قوانین خاص هستند که این پاسخ ممکن است یک کلمه، جمله، پاراگراف، تصویر، قطعه صوتی یا کل سند باشد [۵].

هدف اصلی سیستم‌های QA، پاسخ دادن به چه‌کسی (who)، چه‌چیزی (what)، چه‌کسی (whom)، کجا (where)، چه زمانی (when)، چگونه (how) و چرا (why) است. سیستم‌های QA روش‌های بازیابی و استخراج اطلاعات[۱] را برای شناسایی مجموعه‌ای از کاندیدهای محتمل با هم ادغام و سپس پاسخ نهایی را با استفاده از برنامه‌های رتبه‌بندی تولید می‌کنند [۱].

از بین روش‌های موجود در زمینه‌های مختلف پردازش متن از جمله روش‌های زبان‌شناختی، آماری و تطبیق‌الگو؛ روش‌های آماری برای پردازش داده‌های ساخت‌نیافته و با حجم بالا مناسب‌تر هستند. از سوی دیگر، روش‌های آماری به ساختار و گرامر زبان وابستگی نداشته، بنابراین کارهای انجام شده به کمک این روش‌ها قابل تعمیم به سایر زبان‌ها است. با توجه به اینکه سیستم‌های پرسش و پاسخ با حجم زیادی از داده‌ها سروکار دارند و در حالت ایده‌آل، مطلوب است که مستقل از زبان باشند، از این رو در طراحی این سیستم‌ها روش‌های آماری مورد توجه محققین قرار گرفته است.

---

[۱] Information Extraction

## ۱-۳- تاریخچه سیستم‌های پرسش و پاسخ

در این بخش ما سیر توسعه سیستم‌های پرسش و پاسخ را از دهه ۱۹۶۰ تا کنون را مورد بررسی قرار می‌دهیم. اولین بار طرح توسعه سیستم‌هایی که بتوانند پاسخ‌گوی سؤالات زبان طبیعی باشند، در نسل پنجم زبان برنامه‌نویسی کامپیوتری آغاز شد. وجود این سیستم‌ها کمک می‌کند تا کاربران برای تعاملات بین خود و کامپیوتر، نیازی به یادگیری زبان‌های رسمی مانند SQL، Prolog، Lisp و غیره برای ارسال ورودی نداشته باشند [۳].

سیستم NLIDB برای سهولت دریافت اطلاعات از پایگاه داده به کمک سؤالات در زبان طبیعی، برای کاربران طراحی شد [۶]. در سال ۱۹۶۱ گرین و همکاران سیستمی به نام BASEBALL را پیشنهاد می‌کنند. BASEBALL یک سیستم پرسش و پاسخ بوده که اطلاعات مربوط به لیگ بیس بال را در یک فصل خاص در آمریکا ارائه می‌دهد. سؤالات کاربران با استفاده از کارت پانچ‌هایی به زبان انگلیسی به سیستم داده می‌شد. پس از دریافت پرسش، سیستم با استفاده از لغت‌نامه در دسترس و با تحلیل ساختاری و گرامری پرسش مطرح شده، با توجه به کلمات موجود در آن، پاسخ مناسب را تعیین می‌نمود. دامنه اطلاعات این سیستم محدود به تاریخ، محل و نام تیم‌های شرکت کننده بود [۷].

در سال ۱۹۷۳ وودز سیستم پرسش و پاسخ LUNAR را پیشنهاد کرد که اطلاعاتی در مورد نمونه‌های خاک گرفته شده از اکتشاف ماه سفینه آپولو فراهم می‌کرد. این سیستم سؤالات کاربران را به واسطه قوانین ساده تطبیق الگو به کوئری‌های پایگاه داده تبدیل و در نهایت پاسخ‌ها را تولید می‌کردند. این قوانین ساده تطبیق‌پذیری الگو، از گرامرهای محدود، دانش سخت‌افزاری و قوانین نگاشت که به دامنه‌های برنامه بستگی دارند، استفاده می‌کنند [۸].

هر دو سیستم ذکر شده، به دلیل محدود بودن منابع اطلاعات مربوط به دامنهٔ برنامه، نتایج خوبی را تولید می‌کنند. در تحولات بعدی، طراحی سیستم‌های پرسش و پاسخ با هدف تجزیه و تحلیل زبان‌شناختی سؤالات به منظور دستیابی به نیازمندی‌های از پیش تعیین شده، در یک روش طبیعی

انجام شد.

یکی از این سیستم‌ها، سیستم MASQUE بود که سؤالات مطرح شده به زبان طبیعی را به صورت کوئری‌های منطقی نمایش می‌داد. سپس برای بازیابی اطلاعات مورد نظر در پایگاه داده، کوئری‌های منطقی را به کوئری‌های پایگاه داده ترجمه می‌کرد [۳].

FAQFinder با تطبیق دادن سؤال مطرح شده با لیستی از سؤالات جمع‌آوری شده در یک پایگاه داده از طریق شباهت آماری و معنایی، پاسخ را فراهم می‌کرد. در سال ۲۰۰۲ سیستم پرسش و پاسخ PRECISE پردازش زبان طبیعی سؤالات را انجام می‌داد. این سیستم کلاس سؤالات را تعیین کرده و سؤالات wh را به کوئری‌های پایگاه داده مرتبط با خود نگاشت می‌داد. سؤالات یک مجموعه ویژگی یا زوج مقادیری هستند که در آن هر ویژگی با مقدار wh- خود مرتبط است [۳].

در سال ۲۰۰۰ سیستم پرسش و پاسخی با نام QUARC سؤالات را براساس انواع مختلف سؤالات نوع what طبقه‌بندی می‌کرد و نوع پاسخ مورد انتظار را با استفاده از آثار لغوی و معنایی نتیجه می‌گرفت [۳].

مشکل ترجمه و تفسیر تاکنون حل نشده است. بعدها نیز، توسعه‌دهندگان سیستم‌های پرسش و پاسخ تمرکز خود را روی گونهٔ دامنه باز این سیستم‌ها قرار دادند. تحقیقات روی سیستم‌های پرسش و پاسخ دامنه باز با منابع داده بدون ساختار، بوسیله کمیته ارزیابی TREC انجام شد که هر ساله از سال ۱۹۹۹ به طور منظم انجام می‌شود [۹].

اولین کمیته ارزیابی TREC لیستی از ۲۰۰ سؤال و یک مجموعه سند را فراهم کرد. پاسخ‌ها در مجموعهٔ اسناد حضور داشتند و حداکثر طول مجاز برای آن‌ها ۵۰ تا ۲۵۰ کاراکتر بود. سیستم باید در مقابل درخواست انجام شده، یک لیست ۵تایی رتبه‌بندی شده از پاسخ ها را ارائه می‌داد. در کمپین بعدی، TREC-9 که در سال ۲۰۰۰ برگزار شد، تعداد سؤالات و اندازه مجموعه اسناد افزایش یافت.

در TREC-10 در سال ۲۰۰۱، یک پیچیدگی جدید در رابطه با پاسخ‌ها، یعنی وظیفه اعتبارسنجی پاسخ گنجانده شده بود، زیرا به تمام پاسخ‌هایی که در مجموعه اسناد بود، اطمینان وجود نداشت.

همچنین طول پاسخ ها به ۵۰ کلمه کاهش یافت. بنابراین در TREC-11 کـه در سـال ۲۰۰۲ برگـزار شد، انتظار می‌رفت سیستم‌ها پاسخ‌های دقیق و کوتاهی به سؤالات ارائه دهند. در TREC از ۲۰۰۲ تا ۲۰۰۷ لیستی از سؤالات، پرسش‌های تعریفی و پرسش‌های حقیقی مورد ارزیابی قرار گرفتند.

سؤالات موقتی به TREC-2005 و TREC-2006 اضافه شدند. مجموعه‌ای از ۷۵ موضوع وجود داشت که شامل انواع مختلفی از سؤالات بـود. در TREC-2007، مجموعـه اسـناد از مجموعـه‌هـای وبـلاگ جمع‌آوری شده بود. به طور خلاصـه، مسـابقات TREC بـا افـزایش تعـداد و پیچیـدگی اسـناد مـورد استفاده، افزایش پیچیدگی سؤالات و افزایش پیچیدگی راهبردهای ارزیابی پاسخ، پیشرفت می‌کنند.

کمیته TREC یک مجموعه داده‌های محلی را به عنوان منبع اطلاعات برای تولیـد پاسـخ هـا فـراهم می‌کند، اما با گسترش اطلاعات در سطح وب جهان گستر ، مجموعه‌های زیادی از داده در وب وجود دارد که می‌تواند اطلاعات مفیدی را برای کاربران فراهم کند. چنین مجموعهٔ بزرگی می‌تواند به عنوان یک پایگاه دانش برای پاسخ دادن به سؤالات کاربران مورد استفاده قرار گیرد [۱۰].

در این راستا، تعدادی سیستم پرسش و پاسخ مبتنی‌بر وب پدید آمدند که بـه دو دسـته دامنـه-بـاز و دامنه-محدود تقسیم‌بندی می‌شوند. از سیستم‌های پرسش و پاسخ دامنه-بسته می‌تـوان بـه سیسـتم Webclopedi، Mulder و Answerbus اشاره کرد. چند نمونه از سیستم‌های دامنـه بـاز نیـز Start، Naluri و Webcoop هستند. بیشتر سؤالات مطرح شده در این سیستم‌هـا سـؤالات حقیقـی هسـتند [۱۱].

انواع سیستم‌های پرسش و پاسخ از تکنیک‌های مختلفی نظیر ویژگی‌های تحمل کنندهٔ خطـا، تطـابق کلمات کلیدی و قوانین، برای سـازگاری بـا پاسـخ‌هـا از طریـق WordNet اسـتفاده مـی‌کننـد [۱۲]. پاسخ‌های تولید شده توسط این سیستم‌ها معمولاً به صورت متن، XML و یا اسناد ویکـی‌پـدیا اسـت. سیستم‌های پرسش و پاسخی نظیر start و QAS اطلاعـات قابـل تـوجهی را از داده‌هـای محلـی وب گرفته و به عنوان منبع تولید پاسخ برای پرسش‌های مطرح شده، با اسـتفاده از تکنیـک‌هـای زبـانی و مبتنی‌بر قانون استفاده می‌کنند [۱۳].

علاوه بر این منابع، محققان وب معنایی را نیز به عنوان منابع داده درنظر گرفتند. در مرجع [۱۴] از یک رویکرد تطبیق الگوی مبتنی‌بر الگو به روی منابع چارچوب داده‌های توصیفی استفاده شده است. نویسنده ادعا می‌کند که این روش می‌تواند بر روی وب معنایی اعمال شود.

در حال حاضر سیستم‌های پرسش و پاسخ تعاملی در حال تبدیل شدن به یک سرویس محبوب مبتنی‌بر وب هستند. تعاملی بودن این سیستم‌ها از این جهت است که یک کاربر می‌تواند در صورتی که پاسخ‌های بی‌شمار یا بسیار کمی وجود داشته باشد و یا پرسش مطرح شده دارای ابهام باشد، از طریق تعامل با سیستم کیفیت پاسخ را بهبود ببخشد. با توجه به دقت پایین سیستم‌های خودکار، سیستم‌های تعاملی می‌توانند بسیار مفید واقع شوند. به طور گسترده‌ای، تمامی سیستم‌های فروم آنلاین می‌توانند از پرسش و پاسخ تعاملی استفاده کنند. در حال حاضر چندین سیستم وجود دارند که برای سیستم پرسش و پاسخ تعاملی با کاربر طراحی شده‌اند که از مشهورترین آن‌ها می‌توان به Google-Answers اشاره کرد.

یکی دیگر از سیستم پرسش و پاسخ تعاملی موفق Sina-iAsk محسوب می‌شود که هدف اصلی آن ارتباط در جامعهٔ چینی‌ها است. ارسال سؤال در این سیستم برخلاف Google-Answers، هزینهٔ مستقیمی ندارد و در آن افراد داوطلبانه به دیگران کمک می‌کنند تا پاسخ خود را دریافت کنند [۱۵]. مشابه این سیستم می‌توان به Yahoo!Answers اشاره کرد که کاربران می‌توانند در آن با توجه به میزان فعالیت خود، در جامعهٔ سایبری اعتبار به دست آورند.

احتمالاً محبوب‌ترین سرویس پرسش و پاسخ تعاملی Baidu Zhidao است. این سرویس بعد از پیوستن به Sina-iAsk به عنوان یک موتور جستجوی بسیار خوب عمل کرده است [۱۶]. بسیاری از شرکت‌های دیگر، از جمله مایکروسافت، آمازون و Tencent، اخیراً سرویس‌های مشابه‌ای را راه‌اندازی کرده‌اند.

در حالت مرسوم، اغلب تحقیقات در حوزهٔ سیستم‌های پرسش و پاسخ، از تکنیک‌های مبتنی‌بر زبان در پردازش زبان طبیعی مانند تجزیه، برچسب زدن بخشی از جمله و غیره را مورد استفاده قرار دادند. با

پیشرفت‌های اخیر در یادگیری عمیق، مدل‌های شبکهٔ عصبی در این زمینه به نتـایج امیدوارکننـده‌ای رسیدند. اگرچه این سیستم‌ها به نظر دارای روند یادگیری آسـان‌تـری هسـتند، امـا ایـن روش‌هـا بـه آموزش قابل‌توجهی نیاز دارند. ساختارهایی نظیر GRU و LSTM در شبکه‌های عصبی بازگشتی این امکان را می‌دهند تا متون طولانی‌تر مورد نیاز برای سیستم‌های پرسش و پاسخ را بهتر مدیریت کنند. چنین شبکه‌هایی در حال حاضر عملکرد خوبی را برای سیستم‌های پرسش و پاسخ مبتنی بر یادگیری عمیق ارائه می‌کنند. در مرجع [۱۷]، از چندین مدل یادگیری عمیق برای پاسخگویی به سؤال مطـرح شده، استفاده شده است. پس از بررسی دو شبکه مبتنـی بـر RNN، بـروی شـبکه‌هـای end-to end memory networks تمرکز شده است که نتایج دقیق‌تری را در برخی از وظایف پرسش و پاسخ ارائـه می‌دهند و همچنین آموزش نسبتاً سریعی نیز دارند.

پژوهش‌های دیگری نیز در این زمینه به تازگی انجام شده است. در مرجع [۱۸] از یک شبکهٔ حافظه[1] رای بدست آوردن نتایج مطلوب با استفاده از یادگیری با نظارت روی مجموعه دادگان bAbI اسـتفاده شده است.

## ۱-۴- طبقه‌بندی سؤالات در سیستم‌های پرسش و پاسخ

طبقه‌بندی سؤالات نقشی کلیدی را در سیستم‌های پرسش و پاسخ ایفا می‌کند. اگرچـه سیسـتم‌هـای پرسش و پاسخ مختلف، معماری‌های متفاوتی دارند اما همگی به دنبال یک چهارچوب مناسب هستند که بتواند نقش مهم طبقه‌بندی سؤال را ایفا کند [۱۹].

در این سیستم‌ها هر پرسش به یک دستهٔ خاص که مشخص کننده نوع پاسخ احتمالی است، نگاشـت می‌شود. در سیستم‌هایی با این قابلیت، سیستم پاسخ مورد نظر را تنها در دسته مربوطـه جسـت‌وجـو می‌کند، بنابراین یافتن پاسخ بسیار سریع‌تر از سیستم‌های بدون دسته‌بنـدی سـؤال انجـام مـی‌شـود. همچنین دقت پاسخ‌دهی سیستم افزایش یافته و خطای آن کم خواهد شد. از آن‌جا که اگـر پرسشـی

[1] Memory Networks

در دستهٔ اشتباه قرار بگیرد توانایی دریافت پاسخ صحیح را از دست می‌دهد، انجام صحیح و دقیق این عملیات از اهمیت زیادی برخوردار است [۴][۲۰].

## ۱-۵- هدف پژوهش

امروزه فرآیند بازیابی و استخراج اطلاعات از داده‌های در دسترس دارای اهمیت ویژه‌ای است. با توجه به زمان‌بر بودن فرآیند بازیابی اطلاعات به روش سنتی، سیستم‌های پرسش و پاسخ به منظور کاهش زمان و پاسخ‌گویی به سؤالات کاربران به زبان طبیعی پدید آمدند. پس از آن به دلیل ابهام موجود در زبان طبیعی، سیستم‌های پرسش و پاسخ تعاملی پیشنهاد داده شدند. به منظور افزایش دقت در این سیستم‌ها از تکنیک‌های یادگیری ماشین برای طبقه‌بندی سؤالات استفاده می‌شود. همان‌طور که گفته شد در طبقه‌بندی سؤالات، پرسش کاربر بر حسب نوع پاسخ مورد انتظار آن دسته‌بندی می‌شود؛ از این طریق پاسخ‌های دریافت شده را می‌توان با دقت بیشتری بررسی نمود و در نهایت پاسخ مناسب‌تری به کاربر داد.

اخیراً تکنیک‌های یادگیری عمیق برای بالا بردن کیفیت یادگیری در سیستم‌های پرسش و پاسخ به کار گرفته شده است. ما در این پژوهش یک روش طبقه‌بندی سؤال در سیستم‌های پرسش و پاسخ تعاملی ارائه کرده‌ایم. یکی از محدودیت‌های موجود در این پژوهش، در دسترس نبودن مجموعه دادگان مناسب برای یک سیستم تعاملی بود، که به این منظور یک مجموعه از مکالمات جمع‌آوری و برچسب‌گذاری شد. پس از آن ما روشی پیشنهاد دادیم که از ترکیب دو شبکهٔ عصبی عمیق برای استخراج ویژگی از مجموعه داده‌ها و طبقه‌بندی آن‌ها استفاده می‌کند. در انتها در بخش نتایج با استفاده از تست‌های آماری نشان داده‌ایم که روش پیشنهادی، نسبت به روش‌های آماری متداول در دسته‌بندی سؤالات، نتایج بهتری داشته است.

## ۱-۶- ساختار پایان‌نامه

در ادامهٔ پایان‌نامه، در فصل دوم، مروری بر تکنیک‌های استخراج ویژگی و طبقه‌بندی سؤالات در

سیستم‌های پرسش و پاسخ داریم. همچنین در این فصل مروری اجمالی بر مبحث یادگیری عمیق داریم و مفاهیم پایه‌ای آن را بررسی می‌کنیم. در فصل سوم روش پیشنهادی خود را برای طبقه‌بندی سؤالات با استفاده از تکنیک یادگیری عمیق تشریح می‌کنیم. در فصل چهارم، سناریوهای پیاده‌سازی و نتایج را ارائه می‌دهیم و در پایان در فصل پنجم نتیجه‌گیری و کارهای آینده را ارائه می‌دهیم.

# فصل ۲:

# پیشینه تحقیق

## ۲-۱- مقدمه

در این فصل در ابتدا به بررسی ویژگی‌های مهم در طبقه‌بندی سؤالات می‌پردازیم و شاخه‌های کلی آن را بررسی می‌کنیم. سپس تمرکز خود را بر روی روش‌های طبقه‌بندی قرار می‌دهیم و کارهای مهم انجام شده در این زمینه را بیان می‌کنیم.

## ۲-۲- تعریف مسأله

همان‌طور که در فصل پیش گفته شد، طبقه‌بندی سؤالات نقشی کلیدی را در سیستم‌های پرسش و پاسخ ایفا می‌کند. هر سیستم پرسش و پاسخ با هر نوع معماری برای داشتن عملکرد مطلوب، نیازمند یک چهارچوب مناسب برای طبقه‌بندی سؤال است. در فرآیند طبقه‌بندی سؤال، هر پرسش به یک دستهٔ خاص که مشخصکننده نوع پاسخ احتمالی می‌باشد، نگاشت می‌شود، بنابراین یافتن پاسخ با سرعت و دقت بالاتری نسبت به سیستم‌های بدون دسته‌بندی سؤال انجام می‌شود.

## ۲-۳- ویژگی‌های مهم در طبقه‌بندی سؤالات

اگرچه طبقه‌بندی سؤالات به همان شیوهٔ دسته‌بندی اسناد کار می‌کند، اما دستیابی به دقت مناسب در طبقه‌بندی سؤالات بسیار چالش‌برانگیزتر است. دلیل آن نیز این است که برخلاف دسته‌بندی یک سند، اطلاعات موثر موجود در یک سؤال به مراتب کم‌تر است [۲۱]. این مسأله روی قدرت طبقه‌بندی سؤالات تأثیر می‌گذارد. به همین دلیل انتخاب روش مناسب برای استخراج ویژگی در این سیستم‌ها از اهمیت بالایی برخوردار است.

### ۲-۳-۱- مدل کیسه‌ای از کلمات

این مدل ساده‌ترین روش برای استخراج ویژگی در سیستم‌های بازیابی اطلاعات است که در آن مجموعهٔ کلمات موجود در پرسش مطرح شده و مجموعه اسناد موجود به عنوان ویژگی درنظر گرفته می‌شود. در این روش ترتیب قرار گرفتن کلمات در متن و ارتباط نحوی و یا معنایی آن‌ها در نظر

گرفته نمی‌شود. برای بهبود کارایی این روش، کلمات پرتکرار عمومی که فاقد ارزش معنایی هستند، از مجموعه کلمات حذف می‌شوند [۲۱]. مراحل بازیابی اطلاعات به کمک ویژگی کیسه‌ای از کلمات، به صورت زیر است:

۱- از مجموعه اسناد موجود، تمامی کلمات به جز کلمات عمومی و بدون ارزش معنایی در یک لغت‌نامه جمع‌آوری می‌شود.

۲- ماتریسی تعریف می‌شود که در آن هر سطر، معادل یک لغت از لغت‌نامه و هر ستون،۵۶۷ معادل نام یکی از اسناد است. همچنین هر درایه از ماتریس مقدار صفر یا یک می‌پذیرد که نشان می‌دهد هر کلمه در کدام اسناد موجود است.

۳- با بکارگیری یک تابع بازیابی که از مدل دودویی استفاده می‌کند، سیستم پرسش و پاسخ مجموعه‌ای از اسناد را در پاسخ به سؤال کاربر برمی‌گرداند.

### ۲-۳-۲- مدل N-گرم

یک N-گرم دنباله‌ای از n کلمه است که به صورت متوالی در یک متن ظاهر می‌شوند. به طور معمول جملات با موضوعات یکسان دارای N-گرم‌های مشابه هستند، بنابراین می‌توان برای یافتن میزان ارتباط یک متن و یک جمله که می‌تواند پرسش مطرح شده از سوی کاربر باشد، تعداد تکرار N-گرم‌های جمله مورد نظر را در هر یک از متون موجود در پایگاه داده محاسبه کرد. N-گرم با اندازه یک را ۱-گرم، با اندازه دو را ۲-گرم و با اندازه سه را ۳-گرم می‌گویند. از نقاط قوت مدل آماری N-گرم می‌توان به سادگی پیاده‌سازی و مقاومت نسبت به نویز اشاره کرد [۲۲].

در استفاده از مدل N-گرم، لازم است به نوع و اندازه مجموعه آموزشی دقت کرد. اگر این مجموعه مربوط به یک موضوع خاص باشد، نمی‌توان احتمالات بدست آمده را با دقت مناسب به جملات جدید تعمیم داد. در مقابل نیز در صورتی که مجموعه آموزشی شامل متن‌های بسیار عمومی باشد، امکان رسیدن به نتیجهٔ مطلوب روی دامنه‌های خاص از مفاهیم ضعیف است.

در کنار مسائل گفته شده، توجه به این نکته ضروری است که افزایش مقدار N لزوماً دقت مدل

استفاده شده را بالا نمی‌برد. برای بدست آوردن احتمالات مناسب در مرتبه‌های بالاتر نیاز به مجموعه متون بزرگ‌تری وجود دارد. بنابراین در بیشتر کارهای انجام گرفته، از مدل‌های ۲-گرم و یا ۳-گرم استفاده شده است [۲۳].

### ۲-۳-۳- مدل فضای برداری

در این مدل پرسش مطرح شده و اسناد مورد بررسی، هر کدام به یک بردار از کلمات تبدیل می‌شوند. سپس زاویهٔ بین بردار پرسش کاربر و هر یک از اسناد اندازه گرفته شده و بر اساس آن میزان شباهت اسناد و پرسش مشخص می‌شود.

در این روش ابتدا با در نظر گرفتن لغات سؤال و مجموعه اسناد، لغت‌نامه‌ای ساخته می‌شود. پس از آن برداری به طول لغت‌نامه ساخته می‌شود و این بردار به عنوان بردار ویژگی برای سؤالات و اسناد موجود پر می‌شود [۲۳]. می‌توان درایه‌های این بردار را به روش های مختلفی پر کرد. در این جا به چند روش پرکاربرد اشاره می‌شود:

۱- در ساده‌ترین حالت، می‌توان تنها به وجود و یا عدم وجود آن کلمه در سند یا پرسش توجه نمود و درایه منتاظر را با صفر یا یک مقداردهی کرد.

۲- تعداد دفعات تکرار لغت در سند یا پرسش، در درایه متناظر ذخیره شود.

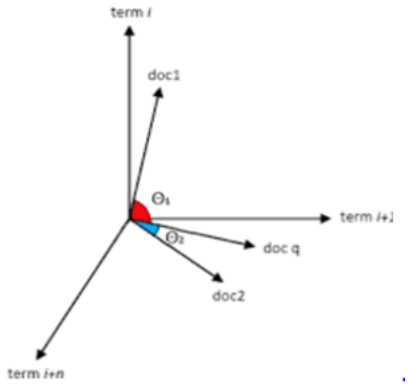
۳- استفاده از روش TF*IDF که در آن با توجه به تعداد دفعات تکرار یک لغت در سند مورد بررسی و معکوس فراوانی تکرار لغت در کل اسناد ، وزن لغت در درایه متناظر با آن ذخیره می‌شود. در این روش به کلمات مهم‌تر وزن بیشتر و به کلمات کم اهمیت‌تر وزن کمتری داده می‌شود. باید توجه داشت که هر چه تعداد دفعات تکرار یک لغت در یک متن بیشتر و تعداد اسناد دارای آن لغت کمتر باشد، لغت دارای وزن بیشتری خواهد شد.

در نهایت پس از بدست آوردن بردارهای مربوطه، میزان شباهت پرسش و اسناد موجود با کمک روش‌های تعیین شباهت دو بردار بدست می‌آید. از پرکاربردترین این روش‌ها می‌توان به روش مشابهت کسینوسی اشاره کرد. در این روش کسینوس زاویه بین دو بردار مورد نظر اندازه‌گیری

می‌شود. تشابه کسینوسی دو بردار q و d از رابطه ۲-۱ محاسبه می‌شود.

$$sim\ (d,q) = d * q/|d||q| \qquad (۱-۲)$$

در رابطهٔ بالا، d*q حاصلضرب برداری d و q است که با ضرب کردن فرآیندهای منتاظر در هم، محاسبه می‌شود. شکل ۲-۱ نمایش برداری دو سند و یک پرسش است.




شکل (۲-۱) نمایش برداری یک درخواست و دو سند

با توجه به این شکل، طبق رابطه ۲-۳ تشابه میان هر یک از اسناد $doc_1$ و $doc_2$ با جست‌وجوی Q، به ترتیب برابر با کسینوس زاویه بین دو بردار یعنی $\theta_1$ و $\theta_2$ است.

$$\begin{cases} sim(doc_1, Q) = Cos(\theta_1) \\ sim(doc_2, Q) = Cos(\theta_2) \end{cases} \qquad (۱-۲)$$

### ۲-۳-۴- مدل احتمالاتی

در این مدل پس از ارائهٔ پرسش، احتمال ارتباط اسناد موجود با درخواست کاربر محاسبه می‌شود و در نهایت فهرست مرتب شده‌ای از اسناد به کاربر ارائه می‌شود که اولین سند در آن، محتمل‌ترین سند برای پاسخ‌گویی به نیاز کاربر است. در این مدل، یک زبان برای سند در نظر گرفته شده و طبق رابطهٔ ۲-۳ روابط تعریف می‌شوند:

$$P(Q|D) = \prod_{i=1...M} P(q_i|D) \qquad (۲-۲)$$

$p(D)$ احتمال اولیهٔ سند و $P(Q \mid D)$ احتمال شرطی وجود سؤال در یک سند است.

فرمول فوق با فرض استقلال هر یک از کلمات کلیدی سؤال که با $q_i$ نمایش داده می‌شوند، محاسبه می‌گردد و M بیانگر تعداد کلمات کلیدی در سؤال است. آنگاه با توجه به فرمول بیز که در رابطهٔ ۲-۴ آورده شده است، احتمال مرتبط بودن هر سند با سؤال محاسبه شده و بر این اساس اسناد مرتب می‌شوند.

$$P(D|Q) \propto P(Q|D)P(D) \qquad (۳-۲)$$

این مدل اسناد و پرسش‌ها را به برداری از کلمات تبدیل می‌کند و از این نظر شبیه مدل برداری عمل می‌نماید. البته مدل احتمالاتی، برخلاف مدل برداری اسناد را نه براساس میزان مشابهت آن‌ها با پرسش، بلکه براساس میزان احتمال ارتباطشان با پرسش، بازیابی و مرتب می‌کند.

لازم به ذکر است که مدل‌های احتمالاتی هم می‌توانند از ویژگی‌های کاراکتری و هم از ویژگی‌های لغوی استفاده نمایند. یکی از نمونه‌های مدل احتمالاتی، روش بیزین است که معروفیت زیادی داشته و هموارسازی را به خوبی فراهم می‌کند [۲۳].

### ۲-۳-۵- تحلیل ساختاری، گرامری و معنایی

گرامر یک زبان، مجموعه‌ای از دستورات زبانی است که با استفاده از آن می‌توان یک جمله را به اجزای نحوی آن تفکیک نمود. تجزیه و تحلیل جمله و شکستن آن به اجزای تشکیل دهنده مانند اسم، فعل، صفت، قید و مضاف‌الیه توسط پارسرهای[1] زبان انجام می‌شود. باید توجه داشت که نادیده گرفتن ساختار جملات و نقش معنایی لغات می‌تواند فهم جمله را دچار مشکل نماید. برخی از اقداماتی که در تجزیه و تحلیل ساختار گرامری جملات انجام می‌شود، عبارتند از:

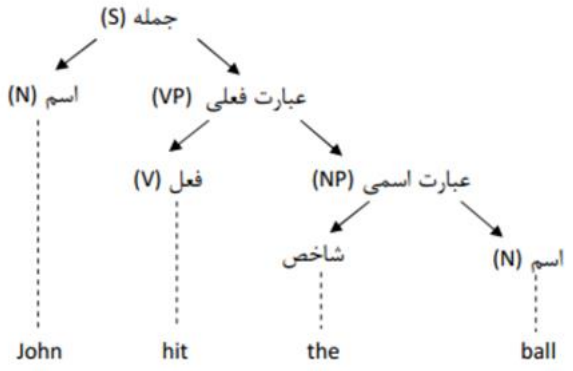
[1] parser

۱- ریشه‌یابی کلمات[۱] که در آن هر کلمه با ریشهٔ خود جایگزین می‌شود و در نتیجه بازیابی اطلاعات براساس ریشهٔ کلمه سبب افزایش کارایی سیستم‌های بازیابی اطلاعات خواهند شد.

۲- شناسایی و برچسب زدن اجزای واژگانی متن[۲]؛ که به این وسیله نقش کلمات در جمله مشخص شده و می‌توان جملات را به خوبی تحلیل کرد.

۳- شناسایی لغت اصلی[۳] پرسش؛ که بیانگر اطلاعاتی است که کاربر در جستجوی آن می‌باشد.

۴- استفاده از درخت نحوی اجزای جمله[۴]؛ که می‌توان به کمک آن نوع لغات تشکیل‌دهنده آن را تعیین نمود. در شکل ۲-۲ یک نمونه از درخت تجزیه را مشاهده می‌کنید.




شکل (۲-۲) یک نمونه درخت تجزیه جمله

۵- یکسان‌سازی متون با استفاده از کلمات مترادف؛ که بدین وسیله پرسش مطرح شده و اسناد همسان‌سازی شده و در نتیجه دقت پاسخ‌گویی افزایش می‌یابد.

۶- استفاده از سطوح عام‌تر کلمات به جای سطوح خاص‌تر آن‌ها؛ که به عنوان مثال می‌توان به کلمهٔ میوه اشاره کرد که نسبت به کلمهٔ سیب دارای معنای عام‌تری می‌باشد. استفاده از معنای عام‌تر کلمات، میزان کارایی سیستم‌ها در بازیابی اطلاعات را افزایش می‌دهد.

---

[۱] Stemming

[۲] Part Of Speech (POS)

[۳] Headword

[۴] Syntactic and semantic parse tree

به طور کلی می‌توان گفت که تجزیه و تحلیل ساختار گرامری و معنایی جملات می‌تواند سبب فهم بهتر پرسش مطرح شده و در نتیجه افزایش احتمال تطبیق آن با اسناد موجود شود. در کنار این مسأله باید به این نکته نیز توجه کرد که تجزیه و تحلیل نحوی یا معنایی می‌تواند پیچیدگی محاسباتی سیستم را به شدت افزایش دهد و افزایش زمان و هزینهٔ فرآیند بازیابی اطلاعات شود [۴].

### ۲-۳-۶- الگوریتم TF-IDF

همان‌طور که در مدل فضای برداری توضیح داده شد، به کمک الگوریتم TF-IDF می‌توان اهمیت لغات یک متن را با وزن‌دهی به آنان مشخص کرد. برای بازیابی اسنادی که دارای بیشترین احتمال داشتن پاسخ صحیح هستند، می‌توان از روش TF-IDF استفاده نمود. به این منظور با استفاده از این روش، وزن هر یک از لغات موجود در پرسش کاربر در هر یک از اسناد محاسبه می‌شود. سپس اسناد بر اساس مجموع وزن همه لغات، مرتب شده و اسناد دارای بیشترین امتیاز برگردانده می‌شوند. در این الگوریتم، TF[1] بردار فراوانی کلمه را مشخص می‌کند که از طریق رابطه ۲-۵ مشخص می‌شود:

$$TF_j = \frac{n_{i,j}}{\sum_{k=1}^{N} n_{i,k}} \qquad (۲-۴)$$

که در آن صورت کسر تعداد تکرار کلمهٔ jام در سند iام و مخرج کسر جمع تکرار همه کلمه‌ها در سند i را نشان می‌دهد. به این ترتیب می‌توان تعداد هر کلمه را نسبت به سند مورد بررسی نرمال کرد. از سویی، IDF[2] یا معکوس فراوانی سند به این نکته اشاره دارد که اگر یک کلمه در تعداد کمی از سندها حضور داشته باشد، تفکیک بیشتری بین سندها ایجاد می‌کند. برای کلمه jام مقدار IDF با رابطه ۲-۶ مشخص می‌شود که در آن صورت کسر تعداد کل سندها و مخرج آن تعداد سندهایی که حاوی کلمه $t_j$ هستند را نشان می‌دهد.

[1] Term-Frequency
[2] Inverse Document Frequency

$$IDF_j = \log\left(\frac{N}{i:t_j \epsilon d_i}\right) \qquad (۲-۵)$$

بنابراین برای یافتن میزان اهمیت یک کلمه در یک سند، می‌توان از ضرب تعداد وقوع آن کلمه در متن، در معکوس فراوانی سند استفاده کرد. که در رابطه ۲-۷ آورده شده است.

$$TFIDF_{i,j} = \quad TF_{i,j} \times IDF_j = \frac{n_{i,j}}{\sum_{k=1}^{N} n_{i,k}} \log\left(\frac{N}{i:t_j \epsilon d_i}\right) \qquad (۲-۶)$$

باید توجه داشت که در این روش، وزن یک لغت در متن تنها با توجه به تعداد تکرار آن در متن مشخص می‌شود و محل بکارگیری آن در متن تأثیری در انتخاب اسناد نمی‌گذارد. دلیل استفاده از این روش نسبت به سایر روش‌ها، سادگی استفاده و نتایج قابل قبول آن است [۴].

### ۲-۳-۷- چارچوب کلمه به بردار

بازنمایی کلمه به بردار[1] در سال ۲۰۱۳ توسط گوگل پیاده‌سازی شد که بر روی حجم بالای متون و اطلاعات، نتایج بسیار شگرفی را به همراه داشته است. در این روش هر کلمه تحت قالب یک بردار از اعداد نمایش داده می‌شود. این بازنمایی به گونه‌ای است که می‌تواند ساختار زبان را به خوبی مدل کند. الگوریتم این روش به صورت متن‌باز در دسترس است و کتابخانه‌های متعددی در زبان‌های برنامه‌نویسی مختلف برای کار با آن ایجاد شده است.

این نوع بازنمایی به نحوی طراحی شده است که برای مثال اگر بردار کلمهٔ «پادشاه» را از بردار کلمهٔ «مرد» کم کنیم، نتیجه به بردار کلمهٔ «ملکه» بسیار نزدیک است یا اگر اختلاف بردارهای «فرانسه» و «پاریس» را بدست آوریم و نتیجه را از «ایتالیا» کم کنیم، حاصل به بردار «رم» بیشترین نزدیکی را دارد.

به طور مختصر، این الگوریتم برای ساخت بردار کلمات از یکی از دو روش کیسهٔ کلمات پیوسته[2]

[1] word2vec

[2] continuous bag-of-words

(CBOW) و Skip-Gram استفاده می‌کند. این دو روش هر دو یک شبکهٔ عصبی ساده هستند که بدون وجود لایهٔ پنهان، به کمک چند قانون ساده، بردارهای مورد نیاز را تولید می‌کنند. در روش CBOW، ابتدا به ازای هر لغت یک بردار با طول مشخص و با اعداد تصادفی (بین صفر و یک) تولید می‌شود. سپس به ازای هر کلمه از یک سند یا متن، تعدادی مشخص از کلمات بعد و قبل آن را (به غیر از خود لغت فعلی) به شبکهٔ عصبی می‌دهیم. بردار لغت فعلی با یک عملیات سادهٔ ریاضی بدست می‌آید (یا به عبارتی از روی کلمات قبل و بعد یک لغت، آن را حدس می‌زنیم) در نهایت این بردار با بردار قبلی لغت جایگزین می‌شود. زمانی که این کار بر روی تمامی لغات در همهٔ متون انجام گیرد، بردارهای نهایی لغات همان بردارهای مطلوب ما هستند.

روش Skip-gram برخلاف روش قبل، براساس یک لغت داده شده، چند لغت قبل و بعد آن را تشخیص می‌دهد و با تغییر مداوم بردارهای لغات، نهایتاً به یک وضعیت باثبات می‌رسد.

## ۲-۴- روش‌های مهم در طبقه‌بندی سؤالات

همان‌طور که در فصل اول گفته شد، اولین اقدام پس از دریافت سؤال کاربر در سیستم‌های پرسش و پاسخ ، تحلیل پرسش است، که از مراحل مختلفی تشکیل شده است. از مهمترین این مراحل، تعیین نوع پرسش است که برای انتخاب و بازگرداندن پاسخ مناسب از اهمیت زیادی برخوردار است. به این منظور، سؤالات کاربران دسته بندی می‌شوند.

تحقیقات نشان داده است که دسته‌بندی سؤال کاربران بر مبنای پاسخ مورد انتظار می‌تواند کمک زیادی در بهبود عملکرد سیستم‌های پرسش و پاسخ داشته باشد و گامی بزرگ در پیش‌بینی نوع پاسخ می‌باشد. وظیفه دسته‌بندی کلاس، برچسب‌گذاری سؤال پرسیده شده به یک یا چند کلاس بر حسب استراتژی کلاس‌بندی مورد استفاده است. روش‌های موجود در دسته‌بندی سؤالات به دو گروه کلی تقسیم می‌شوند: ۱- روش‌های مبتنی بر قانون ۲- روش‌های مبتنی بر یادگیری ماشین.

### ۲-۴-۱- دسته‌بندی مبتنی بر قوانین

در روش نوع اول، نوع پرسش با توجه به قوانین از پیش نوشته شده تعیین می‌شود. این روش نیازمند قوانین از پیش تعریف شده زیادی برای هر مجموعه از سؤالات است و استفاده از آن‌ها در دادگان دیگر، کارایی الگوریتم را کاهش می‌دهد.

در مقاله [۲۴] یک رویکرد مبتنی بر قانون برای تجزیه و تحلیل و کلاس‌بندی سؤالات امتحانی بر مبنای روش‌های پردازش زبان طبیعی ارائه می‌شود. در این مقاله از طبقه‌بندی Bloom، با هدف طراحی ابزاری برای تعیین سطح ادراک مفاهیم محصلان بر اساس سؤالات امتحانی استفاده شده است. قوانینی به این منظور طراحی شده و با استفاده از یک روش پردازش متن، با استفاده از طبقه‌بندی Bloom برای سؤالات، سطح شناخت افراد مشخص می‌شود. در این بررسی، نشان داده می‌شود که قوانین می‌توانند همانند یک انسان در فرآیند تصمیم‌گیری در هر پردازشی عمل کنند. سؤالات متفاوت سطوح ادراکی مختلفی دارند که این سطوح می‌توانند سبک‌های مختلف سؤال و ساختارهای نحوی متفاوت را نتیجه بدهند. نتایج آزمایش‌ها یک رویکرد قابل رشد را برای کمک به گروه‌بندی سؤالات به صورت خودکار بر مبنای دسته‌بندی Bloom نشان می‌دهد.

### ۲-۴-۲- دسته‌بندی مبتنی بر یادگیری

روش‌های مبتنی بر یادگیری ماشین دسته‌بندی سؤالات را در ۳ مرحلهٔ استخراج ویژگی، آموزش طبقه‌بند و پیش‌بینی نوع سؤالات با استفاده از طبقه‌بند آموزش دیده، انجام می‌دهند. مسألهٔ مهم در دسته‌بندی سؤالات با این روش، استخراج ویژگی‌های مناسب جهت آموزش طبقه‌بند است. این ویژگی‌ها به ۳ دستهٔ کلی ویژگی‌های لغوی، ویژگی‌های نحوی و ویژگی‌های معنایی تقسیم می‌شوند.

همان‌طور که گفته شد، از ویژگی‌های لغوی می‌توان به ویژگی N-گرم، کلمات پرسشی، تعداد کلمات موجود در پرسش اشاره نمود. از ویژگی‌های نحوی می‌توان به ویژگی نوع کلمات بکار رفته در پرسش و نیز تعیین کلمهٔ اصلی پرسش (که دربردارنده موضوع سؤال است) اشاره نمود. این ویژگی‌ها عمدتاً بر

اساس درخت نحوی و بررسی ساختار جمله، بـا اسـتفاده از مجموعـه‌ای از قـوانین مبتنـی بـر گرامـر استخراج می‌شوند. از جمله ویژگی‌های معنایی می‌توان به ویژگی‌های تعیین کلمات با معنای عام‌تـر و تعیین کلمات هماهنگ اشاره نمود. کلمات هماهنگ کلماتی هستند که یا مانند روزهای هفته از نظـر معنایی با هم مرتبط هستند و یا مانند کلمات هم‌معنی به شکل‌های مختلفی ظـاهر مـی‌شـوند. بـرای آموزش طبقه‌بند از مجموعه‌ای از پرسش‌ها که به صورت دستی برچسب خورده‌اند و هر برچسب نـوع پاسخ مناسب برای هر پرسش را نشان می‌دهد، استفاده می‌گردد. تعدادی از کارهای انجام شده در این زمینه را در ادامه مشاهده می‌کنید.

اکثر آثار اخیر روی طبقه بندی سؤالات، براساس روش‌های یادگیری تحت نظارت انجـام شـده اسـت. روش‌های یادگیری تحت نظارت یـک طبقـه‌بنـدی را از یـک مجموعـه آموزشـی متشـکل از سـؤالات برچسب گذاری شده یاد می‌گیرند. روش‌های نظارت شده عمدتاً در مدل طبقه‌بندی و ویژگی‌هایی که از سؤالات استخراج می‌شوند، با سایر روش‌ها تفاوت دارند. انتخاب نوع طبقـه‌بنـد بـر سیسـتم نهـایی طبقه‌بندی سؤال تأثیر زیادی دارد.

مطالعات مختلف، طبقه‌بندهای مختلفی را انتخاب می‌کنند. از ماشین‌های بردار پشتیبانی ، مدل‌هـای حداکثر آنتروپی و شبکه‌های تنک Winnows می‌توان به عنوان مورد استفاده‌ترین طبقـه‌بنـدی‌هـای موجود در زمینه طبقه‌بندی سؤال نام برد. در برخی از کارهای انجام‌گرفته نیز از مدل‌سازی زبان برای طبقه‌بندی سؤال استفاده می‌شود. چند مطالعهٔ دیگر نیز انواع طبقه بندی‌ها ترکیب کرده است. در این بخش، کارهای انجام شده را براساس طبقه بندهـای مـورد اسـتفاده آن‌هـا، بـه طـور خلاصـه بررسـی می‌کنیم.

۱- ماشین بردار پشتیبان

در [۲۵] هوانگ و همکاران به دقت ۸۹,۲٪ در کلاس‌های ریزدانه و ۹۳,۴٪ در کلاس‌های درشت‌دانـه روی مجموعه داده TREC رسیدند. در [۲۶] سیلوا و همکاران دقت ۹۰,۸٪ در کلاس‌هـای ریزدانـه و ۹۵,۰٪ در کلاس‌های درشت‌دانه در همان مجموعه داده شده است که بالاترین دقت گزارش شـده در

این مجموعه داده است. این تفاوت در دقت از داشتن فضاهای ویژگی متفاوت نشأت می‌گیرد.

در [۲۷] دونالد متزلر و همکاران با تجزیه و تحلیل آماری، طبقه‌بندی سؤالات را بر مبنای حقیقت انجام داده‌اند. آن‌ها از svm با هسته پایه شعاعی (rbf) در کار خود استفاده کردند. طبقه‌بند ماشین بردار پشتیبان استفاده شده بروی چندین دیتاست مختلف، به عنوان مجموعه داده و بر اساس ویژگی‌های متفاوت، سبب رسیدن به نتایج مختلف شد. دیتاست‌های TREC و UIUC و MADSCI در این کار استفاده شد. آن‌ها دقت ۸۳,۶٪ در کلاس‌های ریزدانه و ۹۰,۲٪ در کلاس‌های درشت‌دانه از مجموعه داده TREC را گزارش داده‌اند. همچنین دقت ۷۳,۲٪ روی مجموعه داده کوچک MADSCIبا استفاده از ترکیبی از ویژگی‌ها گزارش شده است. برای وظیفه طبقه‌بندی سؤال با svm، نشان داده شده که هسته‌های خطی عملکرد بهتری نسبت به انواع دیگر آن دارند. هدف اصلی از این پژوهش بررسی نقش ویژگی‌های معنایی و نحوی مختلف روی کارایی طبقه‌بندی است. نتایج نشان می‌دهند که ویژگی‌های معنایی توانایی بیشتری در بالا بردن عملکرد، نسبت به ویژگی‌های کاملا نحوی دارند.

در [۲۸] شین و همکاران دقت SVM را بر اساس ۴ نوع مختلف توابع هسته مقایسه کردند. نتیجه در کلاس‌های ریزدانه مجموعه داده TREC در جدول ۲ آورده شده است. طبقه‌بندی‌ها در یک فضای مشخص ویژگی آموزش دیده است.

جدول (۲-۱)  نتایج دقت طبقه‌بند svm با هسته‌های مختلف روی مجموعه دادگان TREC

| هسته | خطی | چندجمله‌ای | RBF | سیگموئید |
|---|---|---|---|---|
| دقت | ٪۸۹,۲ | ٪۸۵,۲ | ٪۸۵,۰ | ٪۸۵,۲ |

در [۲۹] محققان یادگیری نیمه نظارتی را برای استفاده از سؤالات برچسب‌گذاری نشده، با انتظار پیشرفت عملکرد طبقه‌بندها در سلسله‌مراتب پیشنهاد کردند. در این کار محققان یک آموزش سه‌گانه

را اصلاح کرده‌اند. در این آزمایش‌ها، محققان ترکیب‌های مختلفی از یادگیری بانظارت و نیمه نظارتی را در یک معماری سلسله مراتبی بکار گرفتند. در یک روش از طبقه‌بندی سلسله‌مراتبی با یادگیری نیمه نظارتی MEM برای طبقه‌بندهای درشت‌دانه و یادگیری بانظارت SVM برای طبقه‌بندهای ریزدانه استفاده می‌کنند. نتایج نشان می‌دهد که بهبود یادگیری بانظارت به بهبود دقت نهایی کمک می‌کند. جالب است زمانی که اندازه مجموعه آموزشی افزایش می‌یابد، عملکرد طبقه‌بندهای ریزدانه با MEM نسبت به طبقه‌بندهای ریزدانه با SVM پیشرفت بهتری دارد. در نتیجه عملکرد کلی طبقه‌بندهای سلسله مراتبی با مجموعه آموزشی با اندازه ۴۰۰۰ در طبقه‌بندهای ریزدانه با استفاده از MEM به شکل قابل توجهی به مقدار دقت ۸۰,۴٪ بهبود یافته است که در مقایسه با آن استفاده از SVM به دقت ۷۹٪ رسیده است.

در [۳۰] هرچند پیش از آن در بسیاری از بررسی‌های انجام شده تمرکز بر روی طبقه‌بندی سؤال برای سؤالات واقعی گذاشته شده، نشان داده می‌شود که چگونه ماشین بردار پشتیبان به صورت موفقیت‌آمیز در طبقه‌بندی سؤالات open-ended استفاده می‌شود. در این آزمایش، کیسه کلمات به عنوان ویژگی‌ها استفاده شده و با انجام برخی کارهای پیش پردازش، دقت میانگین ۷۴,۶٪ ثبت شده است. در مجموع با استفاده از ۱۱ کلاس از پیش تعریف شده و برچسب‌ها، برخی کلاس‌ها دقت بیش از ۸۰٪ داشته درحالی که کارهای دیگران به دقت کمتر از ۵۰٪ رسیده‌اند.

۲- روش‌های هستهٔ پیشرفته

مقاله [۳۱] ژانگ و لی سیستم طبقه‌بندی سؤال را بر مبنای ماشین بردار پشتیبان ایجاد کردند. این سیستم از درخت کرنل و ساختار نحوی ساده استفاده می‌کند. در این رویکرد، سؤال داده شده در ابتدا به درخت نحوی آن تجزیه می‌شود و سپس این سؤال بر اساس تعدادی از بخش‌های درختی که زیرشاخه‌های درخت نحو اصلی هستند، نمایش داده می‌شود. در واقع آن‌ها یک تابع هسته سفارشی را تعریف می‌کنند که بردار ویژگی را به یک فضای با ابعاد بالاتر نگاشت می‌دهد.

این سیستم به دقت ۹۰٪ روی انواع سؤال درشت‌دانه رسیده که بهتر از روش‌هایی که بر مبنای

ماشین بردار پشتیبان روی کلمات و هسته‌های N-گرم کار می‌کنند، عمل کرده است. از این رو ثابت شده است که درخت کرنل قادر است ماشین بردار پشتیبان را برای ساختارهای نحوی سؤال بکار گیرد. علاوه بر این، نویسندگان همچنین طبقه‌بندی سؤال با استفاده از الگوریتم‌های یادگیری ماشین برمبنای کیسه‌ای از کلمات و ویژگی‌های N-گرم را آزمایش کردند و نتایج نشان می‌دهد که فقط با ویژگی‌های سطحی متن، ماشین بردار پشتیبان بهتر از سایر الگوریتم‌های یادگیری ماشین کار می‌کند و به دقت بالای ۷۹,۲٪ برای انواع سؤالات ریزدانه و دقت ۸۷,۴٪ برای انواع سؤالات درشت‌دانه می‌رسد.

در [۳۲] یک روش مشابه برای تعریف توابع هسته در کار پن و همکاران استفاده شده است (۲۰۰۸). آن‌ها یک هسته درخت معنایی را تعریف کردند که شباهت معنایی بخش‌های درخت را با استفاده از ویژگی‌های معنایی اندازه‌گیری می‌کند. آن‌ها دقت ۹۴,۰٪ را در کلاس‌های درشت‌دانه گزارش دادند، در حالی که ژانگ و لی (۲۰۰۳) دقت ۹۰,۰٪ را برای همان مجموعه داده به دست آوردند.

روش‌های کرنل نیز به صورت نیمه‌نظارتی اعمال شده‌اند.

در [۳۳] توماس و جولیانو (۲۰۰۹) یک هسته معنایی را برای طبقه‌بندی سؤالات تعریف کردند که از متن بدون برچسب استفاده می‌کند. آن‌ها از روش [۳۴] برای کاهش فضای ویژگی به فضای موثرتر با تعریف یک هسته پنهان معنایی استفاده کردند. در رویکرد آن‌ها، ماتریس مجاورت اصطلاحات با وقوع همزمان، در یک مجموعه متن بزرگ تعریف شده است.

۳- مدل آنتروپی بیشینه[1]

در [۳۵] Kocik آزمایش خود را روی مجموعه داده TREC انجام داد و به دقت ۸۵,۴٪ در کلاس‌های ریزدانه و ۸۹,۸٪ در کلاس‌های درشت‌دانه رسید. در [۳۶] با استخراج ویژگی‌های بهتر، Blunsom و همکاران به دقت ۸۶,۶٪ در کلاس‌های ریزدانه و ۹۲,۰٪ در کلاس‌های درشت‌دانه روی همان مجموعه داده رسیدند. در مرجع [۳۷]، هوانگ و همکاران (۲۰۰۸) با استفاده از تکنیک‌های

[1] Maximum Entropy

استخراج ویژگی بهتر، به نتایج بهتری رسیدند. آن‌ها به دقت ۸۹,۰٪ در کلاس‌های ریزدانـه و ۹۳,۶٪ در کلاس‌های درشت‌دانه روی همان مجموعه داده رسیدند.

در [۳۸] لونگوین و همکاران یک روش استخراج زیردرخت برای طبقه‌بندی سؤالات پیشـنهاد کردنـد. در رویکرد آن‌ها سؤال تجزیه می‌شود و زیردرخت‌های درخت تجزیه به عنوان ویژگـی‌هـای سـؤال در نظر گرفته می‌شوند. آن‌ها با استفاده از فضای ویژگی متراکم‌تر نسبت به سایر روش‌ها، به دقت ۸۳,۶٪ در کلاس‌های ریزدانه روی مجموعه داده TREC رسیدند. با همین فضای ویژگی، روش آن‌ها از SVM با هسته درختی بهتر عمل می‌کند.

۴- شبکهٔ تُنُک Winnows

شبکهٔ تنک winnows یک معماری یادگیری چندکلاسه است که به طور خاص برای یادگیری فضـای با ابعاد بالا مناسب است. این شبکه برای هر کلاس یک تابع خطی یاد می‌گیرد. توابع خطی می‌تواننـد توسط یکی از روش‌های به‌روز رسانی نظیر نیوی بیز[1]، پرسپترون[2] و Winnow یاد گرفته شوند.

در [۳۹] هدف دسته‌بندی سؤالات به کلاس‌های معنایی است که محدودیت‌ها را روی پاسخ‌های بالقوه تحمیل می‌کند، به صورتی که آن‌ها بتوانند در مراحل بعدی فرآیند QA مورد استفاده قرار بگیرند. در این مقاله یک طبقه‌بند سلسله‌مراتبی بر پایه معماری یادگیری Snow طراحی شده که بوسیلهٔ سلسله مراتب معنایی قادر است بر اساس انواع پاسخ‌ها، سؤالات را به کلاس‌های ریزدانه دسته‌بندی کنـد. بـر این اساس درنظر گرفتن طبقه‌بندی‌های چندبرچسبی مفید خواهد بود و می‌تواند به نتیجه بالای ۹۰٪ در یک طبقه‌بندی مناسب برسد. در این روند مشاهده شد که تنها ویژگی‌های محلی برای پشتیبانی از صحت پاسخ‌ها کافی نبوده و استفاده از ویژگـی‌هـای معنـایی بـرای یـک عملکـرد مناسـب ضـروری می‌باشد. در این روش تنها تعداد کمی از ویژگی‌ها تعریف شده و بر اساس روش‌های مبتنی بر داده به تعداد زیادی از ویژگی‌ها می‌رسیم.

[1] naïve Bayes
[2] Perceptron

انواع ویژگی‌های ابتدایی استخراج شده برای هر سؤال شامل کلمات واژگانی، بخش‌هایی از تگ‌های گفتار، چانک‌ها (بخش‌ها- عبارات بدون تداخل)، چانک‌های هدر (اولین چانک یا تکه اسم در یک سؤال) و موجودیت‌های نام‌گذاری شده است. در این پژوهش به دقت ۷۸,۸٪ برای ۵۰ کلاس ریزدانه دست یافته‌اند. با یک دیکشنری دست‌ساز از کلمات معنایی مرتبط، سیستم قادر به رسیدن به دقت ۸۴,۲٪ است. این آزمایش از طبقه‌بند سلسله مراتبی استفاده کرده و با وجود آن به یک دقت مناسب بر روی کلاس‌های ریزدانه رسیده است. نتایج نشان می‌دهند که یک کارایی مفید برای استفاده از یک سطح از کلاس‌های درشت‌دانه وجود ندارد و کلاس‌های معنایی برای بهبود عملکرد کمک زیادی نمی‌کند.

در این معماری طبقه‌بندی سؤالات با استفاده از دو طبقه‌بند با کمک الگوریتم winnow انجام شده که طبقه‌بند اول سؤالات را به کلاس‌های درشت‌دانه و طبقه‌بند دوم به کلاس‌های ریزدانه دسته‌بندی می‌کند. طبقه‌بند دوم برای تولید برچسب با توجه به کلاس‌های درشت‌دانه در طبقه‌بند اول عمل می‌کند. مراحل آموزش و تست طبقه‌بندها به صورت بالا به پایین انجام می‌شود. معماری Snow اجازه می‌دهد تا خروجی طبقه‌بند چند برچسب داشته باشد و سپس با استفاده از احتمال شرطی روی برچسب کلاس‌ها و با یک مقدار آستانه نتیجه را مشخص می‌کند.

۵- مدل‌سازی زبان

ایده اولیه مدل‌سازی زبان این است که هر قطعه از متن را می‌توان به عنوان یک زبان دید.

مدل‌سازی زبان به طور گسترده‌ای برای طبقه‌بندی اسناد استفاده شده است. ایده این است که یک سند D به عنوان دنباله‌ای $w_1$، ...، $w_N$ از کلمات مشاهده و احتمال تولید این دنباله برای هر کلاس محاسبه شود. برچسب کلاس با استفاده از قانون بیز[1] تعیین می‌شود.

در [۴۰] از روش اصلاح شده بیز افزایشی استفاده شده است. برای بهبود میزان صحت در طبقه‌بندی سؤالات، مجموعه آموزشی افزایشی برای اصلاح پارامترهای مجموعه آموزشی به صورت مرحله به

[1] Bayes

مرحله، بر مبنای آموزش آماری بکار می‌رود. در این جا از دو روش بیز اصلاح شده و بیز اصلاح شده افزایشی برای طبقه‌بندی سؤالات استفاده شده است. در روش اول فرض بر مستقل بودن توزیع کلمات در سؤال و عدم ارتباط میان کلمات و ترتیب آن‌ها در جمله می‌باشد. بر این مبنا حداکثر مقدار احتمال محاسبه می‌شود. در روش دوم طبق فرمول‌های ارائه شده در هر مرحله، از اطلاعات بدست آمده قبلی در مراحل بعدی استفاده می‌شود. به منظور آزمایش این روش‌ها طبقه‌بندی سؤالات روی دامنه گردشگری یونان انجام شده است. نتایج نشان‌دهنده افزایش ۳,۳٪ دقت متوسط بوده است. همچنین متوسط زمان آموزش نیز ۳۹,۱٪ نسبت به سایر روش‌های کارآمد بیز بهبود داشته است.

در [۴۱] نویسنده الگوریتم رتبه‌بندی مجدد عبارت احتمالی یا همان PPR را پیشنهاد کرده است که از مجاورت و ویژگی‌های انواع سؤال برای گسترش قابلیت موتورهای جستجو برای پشتیبانی پرسش و پاسخ‌های زبان طبیعی استفاده می‌کند. مراحل در PPR عبارتند از مدولاسیون کوئری، بازیابی سند، بازیابی جمله یا یک بند، استخراج عبارت (همان پاسخ) و رتبه‌بندی به پاسخ‌ها. قبل از بازیابی سند و فراهم آوردن پاسخ واقعی برای کوئری کاربر، شناسایی و کلاس‌بندی نوع سؤال اهمیت دارد. در این تحقیقات یک الگوریتم هیوریستیک rule-based برای شناسایی نوع سؤال استفاده شده‌است. در آن هر سؤال بوسیله ۱۳ ویژگی که شامل ۹ ویژگی معنایی برمبنای وردنت که Ripper برای شناسایی نوع سؤال از آن استفاده می‌کند، نشان داده شده است. دومین روش برای شناسایی نوع سؤال، استفاده از الگوریتم هیوریستیک است. در این روش از برچسب‌گذاری POS برای شناسایی نوع سؤالات برای سؤال‌هایی که حاوی هیچ WH کلمه‌ای نیست و همچنین کلماتی که دارای what/which هستند استفاده می‌شود. نتایج نشان می‌دهد که استفاده از هر دو ویژگی مجاورت و نوع سؤال، دقت بازیابی پاسخ واقعی به سؤالات را، بهبود می‌بخشند. همچنین در این آزمایش، سطوح مختلف محتوای سند، برای مشاهده چگونگی بهبود در یافتن جواب‌های خوب از نتایج جستجو مورد مطالعه قرار گرفته است. سطوح مختلف پاسخ‌ها عبارتند از سند، جمله و عبارت. در مقایسه کلی، نتایج نشان می‌دهند که استفاده از سطوح پاسخ جمله و عبارت با ویژگی مجاورت و نوع سؤال تأثیر قابل توجهی در رتبه‌بندی

پاسخ‌های واقعی دارد.

۶- سایر طبقه‌بندها

علاوه بر طبقه بندی‌های ذکر شده، طبقه‌بندی‌های دیگری نیز برای طبقه‌بندی سؤال مورد استفاده قرار گرفته است. در [۴۲] نویسندگان از QuASM که ابرداده نیمه ساختاریافته است در سیستم‌های پرسش و پاسخ استفاده کردند. هدف اصلی پاسخ دادن به سؤالات واقعی بوسیله اکتشاف در ساختار اینترنت در اسناد پیدا شده در دامنه وب و بهبود عملکرد جستجو بوسیله فراهم آوردن پاسخ خاص برای کاربران است. با توجه به اینکه اسناد وب نیمه ساخت یافته، یک نشانه برای آغاز و پایان بخش‌ها به عنوان موضوع مهم ارائه می‌کنند، این اسناد می‌توانند به قسمت‌های کوچک‌تر که معمولا همگن هستند، شکسته شوند. فرضیه‌ای وجود دارد که با ایندکس‌گذاری بخش‌های کوچک از اسناد برای بازیابی اطلاعات، جستجوی پاسخ می‌تواند موثرتر باشد.

آزمایش QuASM همچنین شامل طبقه‌بندی سؤالات نیز می‌باشد. در جهت پیدا کردن پاسخ، نیاز است تا یک ارتباط بین سؤال درخواست شده و پاسخ‌های ممکن در سند ایجاد شود. بنابراین نیاز است تا سؤالات برمبنای پاسخ مورد انتظار دسته‌بندی شوند. موجودیت‌های این پاسخ‌ها باید در اسناد بازیابی شده، شناسایی شود تا بتوان پاسخ‌ها را موقعیت‌یابی نمود. برای دستیابی به این مسأله، عبارات منظم و مدل‌های زبانی نظیر مدل‌های N-گرم بخصوص ۱-گرم و ۲-گرم برای تخمین احتمال مربوط بودن سؤال به کلاس پاسخ، استفاده می‌شوند. با وجود اینکه یافته‌ها برخی از پیشرفت‌ها را در استفاده از QuASM در بازیابی پاسخ‌های مرتبط با کوئری نشان می‌دهد، نویسنده به این نکته اشاره کرده است که بهبود زیادی برای رسیدن به نتایج امیدوارکننده نیاز است. اگرچه طبقه‌بند سؤال با استفاده از ۱-گرم و ۲-گرم بدون ایندکس‌گذاری در آزمایش QuASM، دقت بین ۷۳ و ۷۴٪ را در شرایطی که عبارات در حدود ۵۳٪ منظم هستند ثبت کرده است.

در [۴۳] لی و روث از منابع اطلاعاتی معنایی بیشتری شامل موجودیت‌های نام‌گذاری شده، مفاهیم وردنت، کلمات مرتبط با کلاس‌های خاص و شباهت توزیع‌پذیری برمبنای گروه‌ها در کار طبقه‌بندی

سؤال استفاده کردند. از طبقه‌بندی سلسله‌مراتبی بر مبنای SYN در این آزمایش استفاده شده است. با تمام این ویژگی‌های معنایی به علاوه انواع نحوی آن، مدل خود را با ۲۱۵۰۰ سؤال آموزش داده و قادر به رسیدن به دقت ۸۹,۳٪ روی مجموعه تست با ۱۰۰۰ سؤال گرفته شده از TREC10-11 برای ۵۰ کلاس ریزدانه و دقت ۹۲,۳٪ برای کلاس‌های درشت‌دانه شدند.

در [۴۴] نویسندگان از یک توالی کوتاه از توکن‌های سؤال (معمولا یک تا سه کلمه) یا در واقع از یک محدوده اطلاعات به عنوان یک ویژگی برای طبقه‌بندی سؤال استفاده کردند. مدل آن‌ها می‌تواند به دقت بالای ۸۶,۲٪ با استفاده از دیتاست UIUC برای گروه‌های سؤال ریزدانه برسد که بالاترین دقت گزارش شده روی این دیتاست است. همچنین برای گروه‌های درشت‌دانه دقت بالای ۹۳,۴٪ ثبت شده است. این مدل برمبنای متاکلاسیفایر استفاده شده روی بردار ماشین پشتیبان خطی روی خروجی CRF است.

در [۴۵] برای بهبود طبقه‌بند سلسله مراتبی[۲۹]، لی و همکاران استفاده از طبقه‌بند تصادفی وابسته شرطی طولانی (LDCRFها) را پیشنهاد دادند. CRFS یک نوع مدل احتمالی است که برای برچسب‌گذاری اطلاعات سریالی sequential استفاده می‌شود. در مدل پیشنهاد شده توسط لی و همکاران. (۲۰۰۸)، یک سؤال به عنوان دنباله‌ای از کلمات به هم مرتبط از نظر معنایی، در نظر گرفته شده است.

آن‌ها از CRFS برای برچسب زدن تمامی کلمات یک سؤال استفاده می‌کنند و برچسب کلمهٔ اصلی به عنوان کلاس سؤال در نظر گرفته می‌شود. این رویکرد از جهت درنظر گرفتن سؤال به عنوان داده‌های متوالی، با سایر روش‌های طبقه‌بندی سؤال متفاوت است. این محققان ادعا کردند که این طبقه‌بندی می‌تواند ویژگی‌هایی غنی را جمع‌آوری کند. ویژگی‌های استفاده شده در این کار کلمات، تگ‌های POS، چانکرها، اطلاعات پارسر، طول سؤال، نام موجودیت، کلمات هماهنگ هستند. نتایج دقت ۸۵,۶٪ در کلاس‌های ریزدانه را روی مجموعه داده TREC نشان می‌دهند که کمی بهتر از دقت بدست آمده در[۴۲] و دقت کلی در [۳۲] است.

در [۴۶] از رویکردهای یادگیری ماشین مانند استفاده از طبقه‌بندهای متفاوت و ترکیب آن‌ها برای طبقه‌بندی سؤالات استفاده می‌کند. در این مقاله رویکردهای مختلف نظیر وابستگی ساختار، وردنت، کیسهٔ کلمات، ۲-گرم و همچنین تأثیر تعداد توابع کرنل و روش‌های مختلف ترکیب طبقه‌بند روی دقت طبقه‌بندی سؤالات بررسی شده است. در این مقاله رویکردهای مختلف ذکر شده با استفاده از روش‌های مختلف مانند TBL، ANN، SVM و VOTE با هم ترکیب شده و نتایج ترکیب‌ها با داده‌های آموزشی مختلف بدست آمده است. در این میان ترکیب TBL از طبقه‌بندها بهتر از سایر ترکیب‌ها عمل کرده و دقت بالاتری را در طبقه‌بندی تولید می‌کند. ترکیب TBL می‌تواند به راحتی برای طبقه‌بندی‌های جدید و یا مجموعه ویژگی‌های جدید یافت شده تعمیم یابد که می‌توانند به توابع کلاس‌بندی بیشتری منجر شوند.

در [۴۷] ژانگ و لی دقت طبقه‌بندی سؤال را با ۵ طبقه‌بند متفاوت در همان فضای ویژگی مقایسه کردند. آن‌ها ماشین بردار پشتیبان را با نزدیک‌ترین همسایه، نیوی بیز، درخت تصمیم‌گیری و snow مقایسه کردند که بین آن‌ها ماشین بردار پشتیبان، بهترین عملکرد را داشت. نتایج آن‌ها در کلاس‌های ریزدانه مجموعه دادهٔ TREC در جدول ۲-۲ آورده شده است.

جدول (۲-۲) دقت ۵ طبقه‌بند مختلف با ویژگی کیسهٔ کلمات روی مجموعه دادگان TREC

| رویکرد | دقت (ریزدانه) | دقت (درشت‌دانه) |
|---|---|---|
| NN | ٪۶۸,۴ | ٪۷۵,۶ |
| NB | ٪۵۸,۴ | ٪۷۷,۴ |
| DT | ٪۷۷,۰ | ٪۸۴,۲ |
| SNOW | ٪۷۴,۰ | ٪۶۶,۸ |
| SVM | ٪۸۰,۲ | ٪۸۵,۸ |

در [۴۸] به منظور بهبود عملیات در ترم‌های طبقه‌بندهای سلسله مراتبی ویژه، در این مقاله محققان طبقه‌بندهای گوناگون را (حداکثر آنتروپی، نیوی بیز ، ماشین بردار پشتیبان، درخت تصمیم) برای

طبقه‌بندی اولیه و ثانویه آزمایش کردند و یافته‌ها نشان می‌دهد که یک ترکیب از طبقه‌بند حداکثر آنتروپی درشت‌دانه و طبقه‌بند بیز ریزدانه بهترین ترکیب است. مشاهدات نشان می‌دهد که ترکیب طبقه‌بندها، نسبت به استفاده از یک نوع آن برای طبقه‌بندی اولیه و ثانویه مناسب‌تر است. در نتیجه دقت ۸۰٪ بدون وردنت با استفاده از ۲-گرم و درخت تجزیه بدست آمده که بیش تر از عملکرد بدست آمده در پژوهش‌های قبلی  با اضافه کردن کلمات مرتبط است، بنابراین محققان انتظار امکان افزایش عملکرد را در صورت کار با اطلاعات معنایی بیشتر دارند.

نتایج جدول ۲-۲ نشان می‌دهند که svm در مقایسه با سایر طبقه‌بندها، زمانی که از فضای ویژگی یکسان استفاده شود، بهتر عمل می‌کند. با این حال، بسته به ویژگی‌های استخراج شده، ممکن است سایر طبقه‌بندهای دیگر بهتر عمل کنند. بنابراین با توجه به فضای ویژگی و پارامترهای دیگر، طبقه‌بندی بهینه می‌تواند متفاوت باشد.

## ۲-۵- جمع‌بندی

در این فصل ویژگی‌های مورد استفاده در دسته‌بندی سؤالات و متون را بررسی نمودیم و شیوه به کارگیری آن‌ها در زمینه طبقه‌بندی سؤالات را با بررسی تعدادی از مقالات مهم در این حوزه نشان دادیم. در فصل بعدی مروری بر ساختار شبکه‌های عمیق و پژوهش‌های انجام شده در زمینهٔ پردازش متن به کمک این شبکه‌ها را مطرح کرده‌ایم.

# فصل ۳:

# یادگیری عمیق

## ۳-۱- مروری‌بر یادگیری عمیق

رویکردهای یادگیری ماشین، در بسیاری از جنبه‌های جامعه امروزی مانند جستجوهای وب تا فیلتر کردن محتوا در شبکه‌های اجتماعی، در وب‌سایت‌های مربوط به پیشنهادهای تجاری و در حال حاضر در محصولات مصرفی مانند دوربین‌ها و تلفن‌های هوشمند،کاربرد داشته و قدرت می‌بخشد. سیستم‌های یادگیری ماشین در مواردی همچون شناسایی اشیاء در تصویر، تبدیل گفتار به متن، تطبیق عناوین خبری، یافتن محصولات مورد علاقه کاربر و انتخاب نتایج درست جستجو کاربرد دارند. به طور فزاینده، این کاربردها از دسته‌ای از تکنیک‌ها به نام یادگیری عمیق استفاده می‌کنند.

روش‌های معمول یادگیری ماشین در توانایی پردازش داده‌های طبیعی به صورت خام، دارای محدودیت هستند. برای ده‌ها سال، ساخت یک سیستم تشخیص الگو یا یادگیری ماشین، نیاز به مهندسی دقیق و تخصص ویژه در زمینه طراحی یک سیستم استخراج ویژگی داشت. این سیستم باید بتواند یک نمایش داخلی مناسب از بردار ویژگی‌ها برای داده‌های خام (مانند مقادیر پیکسل‌های یک تصویر) ارائه کرده تا به وسیله آن، زیر سیستم یادگیری(که معمولاً یک رده‌بند است) بتواند الگوهای ورودی را به درستی دسته‌بندی کند [۴۹].

یادگیری بازنمایی شامل دسته‌ای از روش‌ها است که داده‌های خام را به عنوان ورودی دریافت و به طور خودکار بازنمایی مناسب را برای سیستم تشخیص یا رده‌بندی تولید می‌کند. روش‌های یادگیری عمیق، نوعی از روش‌های یادگیری بازنمایی هستند که از چندین سطح بازنمایی تشکیل شده‌اند. هر سطح به وسیله ترکیب ماژول‌های ساده اما غیر خطی به دست می‌آید که در هر کدام، این بازنمایی‌ها از یک سطح (که از داده خام شروع می‌شود) به سطح بالاتر (و انتزاعی تر) انتقال داده می‌شوند. با ترکیب تعداد کافی از این انتقال‌ها، توابع بسیار پیچیده، قابل یادگیری خواهند بود.

در کارهای رده بندی، لایه‌های بالاتر از بازنمایی، جنبه‌هایی از داده‌های ورودی را که برای تشخیص مهم‌تر هستند، تقویت کرده و متغیرهای بی‌ربط را از بین می‌برد. به عنوان مثال یک عکس به وسیله

آرایه‌ای از مقادیر پیکسل‌ها نمایش داده می‌شود. معمولاً ویژگی‌های یاد گرفته شده در لایه اول بازنمایی، مواردی مانند بودن یا نبودن لبه‌ها در یک جهت‌گیری و یا مکان خاص در عکس را نمایش می‌دهند. لایه دوم نیز معمولاً موارد خاصی را با کنار هم قرار دادن لبه‌ها، بدون توجه به تغییرات کوچک در مکان لبه‌ها تشخیص می‌دهد. سومین لایه این موارد خاص را به یک ترکیب بزرگ‌تر که مربوط به قسمت‌هایی از یک شی خاص است، تبدیل می‌کند. در نهایت لایه‌های بعدی، با ترکیب این قسمت‌ها، این شی را تشخیص می‌دهند. جنبه کلیدی یادگیری عمیق این است که این لایه‌های ویژگی، به وسیله مهندسان انسانی طراحی نشده است، بلکه این لایه‌ها به وسیله روش‌های یادگیری عام منظوره، از روی داده‌ها یادگرفته شده‌اند [۴۹].

یادگیری عمیق، در حل مسائلی که بهترین رویکردهای هوش مصنوعی سال‌ها در حل آن‌ها ناموفق بوده‌اند، پیشرفت فراوانی به دست آورده است. این روش برای کشف ساختارهای پیچیده در داده‌های با ابعاد بالا بسیار خوب عمل می‌کند و از این‌رو در دامنه‌های بسیاری از علوم و تجارت قابل استفاده است.

یادگیری عمیق، علاوه بر بهبود نتایج در زمینه پردازش تصویر و تشخیص گفتار، نتایج روش‌های دیگر یادگیری ماشین در زمینه‌هایی همچون فعالیت بالقوه مولکول‌های دارویی، تحلیل اطلاعات شتاب‌دهنده ذره‌ای، بازسازی مدارهای مغزی و پیش‌بینی تأثیر جهش در DNA رمزگذاری نشده در حالات ژن و بیماری، را نیز بهبود بخشیده است. یادگیری عمیق برای سرفصل‌های گوناگون در پردازش و درک زبان طبیعی مخصوصاً طبقه‌بندی موضوعات، تجزیه و تحلیل احساسات، پاسخ به سؤالات و ترجمه زبان نتایج بسیار نویدبخشی را تولید کرده‌است.

این دیدگاه وجود دارد که یادگیری عمیق موفقیت‌های زیادی در آینده نزدیک دارد، زیرا نیازمند نظارت انسانی بسیار کمی است. بنابراین در این روش می‌تواند به سادگی از افزایش مقدار محاسبات در دسترس و داده‌ها سود برد. الگوریتم‌ها و معماری‌های یادگیری جدید که در حال حاضر برای شبکه های عصبی عمیق توسعه داده می شوند، این پیشرفت را تسریع خواهند کرد [۴۹].

### ۳-۱-۱- یادگیری با ناظر

رایج‌ترین گونه از یادگیری ماشین (عمیق یا غیر عمیق) یادگیری با ناظر است. تصور کنید می‌خواهیم سیستمی بسازیم که تصاویر را بر اساس وجود خانه، ماشین، انسان و حیوان دسته‌بندی کند. ابتدا مجموعه داده بزرگی از تصاویر خانه، ماشین، مردم و حیوان جمع می‌کنیم که هر کدام به وسیله دستهٔ خود برچسب‌گذاری شده‌اند. در اینجا ما قصد داریم تا دسته مورد نظرمان امتیاز بیشتری نسبت به سایر دسته‌ها داشته باشد، اما این کار پیش از یادگیری اتفاق نخواهد افتاد. بنابراین نیاز به تعریف یک تابع هدف داریم تا میزان خطا (یا فاصله) بین امتیازات خروجی و الگوی خروجی مطلوب ما را محاسبه کند. سپس ماشین پارامترهای داخلی قابل تنظیم خود را تغییر می‌دهد تا این خطا را کاهش دهد. این پارامترهای قابل تنظیم که معمولاً «وزن» نامیده می‌شوند، اعدادی حقیقی هستند که می‌توانیم به آن‌ها مانند یک «اهرم» نگاه کنیم که تابع ورودی-خروجی ماشین را تعریف می‌کنند. در یک سیستم معمولی یادگیری عمیق، ممکن است صدها میلیون عدد از این وزن‌های قابل تنظیم و صدها میلیون نمونه برچسب خورده وجود داشته باشد که به کمک آن‌ها ماشین را آموزش می‌دهیم [۴۹].

به منظور تنظیم درست بردار وزن‌ها، الگوریتم یادگیری یک بردار گرادیان را محاسبه می‌کند که در آن برای هر وزن یک بردار گرادیان تعیین می‌شود، در صورتی‌که این وزن به میزان کوچکی زیاد شود، خطا افزایش یا کاهش می‌یابد. پس از آن، بردار وزن در جهت مخالف بردار گرادیان حرکت می‌کند.

تابع هدف که روی تمامی مثال‌های یادگیری، میانگین گرفته می‌شود، می‌تواند به عنوان نوعی فضای تپه‌مانند از وزن‌ها در نظر گرفته شود. بردار منفی گرادیان نشان‌دهنده جهت بیش‌ترین کاهش در این فضای تپه‌مانند است که خطا را به مقدار کمینه آن نزدیک می‌کند و خطای خروجی را به طور میانگین کاهش می‌دهد.

در عمل، بسیاری از متخصصان از فرآیندی به نام کاهش تصادفی گرادیان استفاده می‌کنند. این فرآیند شامل بخش‌هایی نظیر نمایش بردار ورودی برای تعدادی از نمونه‌ها، محاسبه خروجی‌ها، محاسبه خطا و محاسبه میانگین گرادیان برای نمونه‌های انتخابی می‌باشد. در نهایت تنظیم وزن‌ها بر

این اساس انجام می‌گیرد. این فرآیند برای تعداد زیادی از مجموعه‌های کوچک از مجموعه یادگیری، تکرار می‌شود تا زمانی که دیگر میانگین تابع هدف کاهش پیدا نکند. به این دلیل به آن اتفاقی می‌گوییم که هر مجموعه کوچکی از این نمونه‌ها، یک تخمین نویزی از گرادیان متوسط کل مثال‌ها می‌دهد. این فرآیند ساده در مقایسه با روش‌های عالی بهینه‌سازی، معمولاً به شکل عجیبی، به سرعت مجموعه خوبی از وزن‌ها را پیدا می‌کند. پس از یادگیری، کارایی سیستم روی مجموعه متفاوتی از مثال‌ها به نام مجموعه آزمون اندازه‌گیری می‌شود. این کار کمک می‌کند تا بتوانیم توانایی تعمیم ماشین را به درستی بسنجیم. به بیان بهتر، در این مرحله توانایی تولید جواب‌های معقول روی ورودی‌های جدیدی که ماشین در زمان یادگیری ندیده است، بررسی می‌شود.

در حال حاضر، بسیاری از کاربردهای عملی یادگیری ماشین، از رده‌بندهای خطی با ویژگی‌های استخراج شده به صورت دستی، استفاده می‌کنند. یک رده‌بند خطی دوکلاسه، یک جمع وزن‌دار را از اعضای بردار ویژگی محاسبه می‌کند. اگر این جمع وزن‌دار بیشتر از یک مقدار آستانه باشد، ورودی به یکی از کلاس‌ها رده‌بندی می‌شود [۴۹].

گزینه مرسوم طراحی، یک استخراج‌کننده ویژگی خوب می‌باشد که این امر مستلزم میزان قابل توجهی از دانش و مهارت مهندسی در زمینه مورد نظر است. اما در صورتی که بتوانیم به وسیله یک رویه عمومی، ویژگی‌های خوبی را به طور خودکار استخراج کنیم، دیگر نیازی به استفاده از طراحی دستی استخراج‌کننده نداریم. این مزیت کلیدی یادگیری عمیق است.

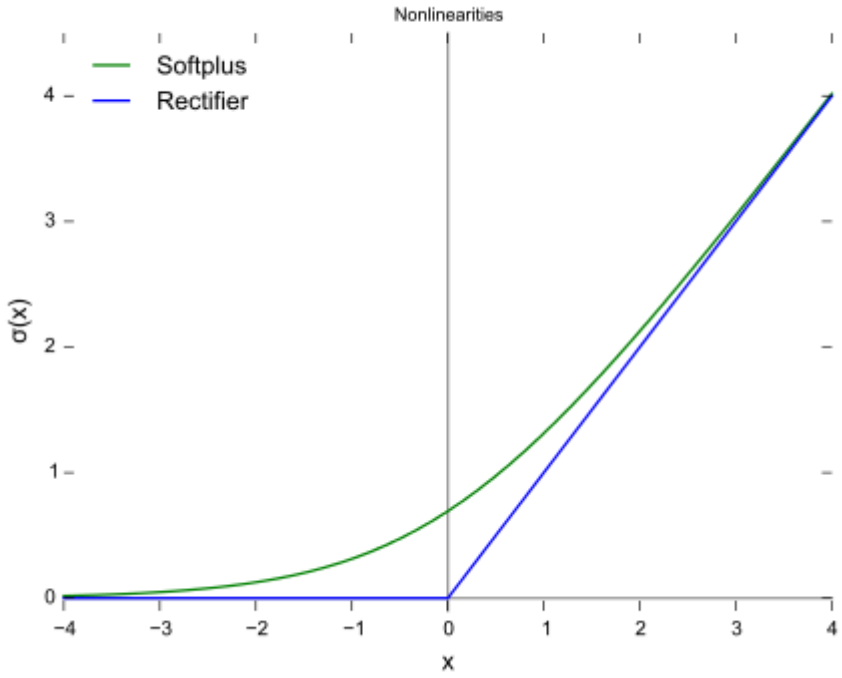
معماری یادگیری عمیق عبارت است از پشته‌ای از ماژول‌های ساده، که همه یا بیش‌تر آن‌ها مربوط به یادگیری است و بسیاری از آن‌ها نگاشت‌های ورودی-خروجی غیرخطی را یاد می‌گیرند. هر ماژول در این پشته، ورودی خود را به گونه‌ای تغییر می‌دهد که هر دو معیار «به‌گزینی» و «ثابت بودن» را افزایش دهد. با استفاده از چندین لایه غیرخطی، مثلاً به عمق ۵ تا ۲۰ عدد، سیستم می‌تواند توابع به شدت پیچیده از ورودی خود را پیاده کند که به طور هم‌زمان هم نسبت به جزئیات دقیق حساس باشد و هم نسبت به تغییرات نامربوط بزرگ، حساس نباشد [۴۹].

### ۳-۱-۲- پس‌انتشار در معماری‌های چند لایه

از ابتدای شروع شناسایی الگو، هدف محققان جایگزین کردن ویژگی‌هایی که به صورت مهندسی به دست آمده‌اند با شبکه‌های چندلایه قابل یادگیری بوده است؛ اما بر خلاف سادگی این کار، راه‌حل آن تا میانه دهه ۱۹۸۰ پیدا نشد. همانطور که گفته شد، معماری‌های چند لایه می‌توانند با کاهش گرادیان ساده آموزش داده شوند. هنگامی که ماژول‌ها توابع نسبتاً همواری از ورودی و وزن‌های داخلی آن‌ها باشند، می‌توان گرادیان‌ها را به روش پی حل کرد. ایده قابل انجام بودن این روش، توسط چند گروه مستقل در بین سال‌های ۱۹۷۰ تا ۱۹۸۰ کشف شد.

روش پس‌انتشار برای محاسبه گرادیان یک تابع هدف با توجه به وزن‌های پشته چند لایه ماژول‌ها، چیزی بیش از یک استفاده عملی از قوانین زنجیره‌ای از مشتق‌ها نیست. نکته کلیدی این است که مشتق (گرادیان) یک تابع هدف نسبت به ورودی‌های یک ماژول، می‌تواند با توجه به خروجی آن ماژول یا ورودی ماژول بعدی و حرکت خلاف جهت گرادیان محاسبه شود (شکل ۱). معادله پس‌انتشار می‌تواند به دفعات بسیار استفاده شود تا خطا را در بین همه ماژول‌ها منتشر کند. این فرآیند از خروجی در بالاترین سطح (جایی که شبکه خروجی خود را تولید می‌کند) آغاز شده و به پایین حرکت می‌کند (جایی که ورودی به شبکه داده می‌شود). هنگامی که این گرادیان‌ها محاسبه شد، محاسبه گرادیان‌ها نسبت به وزن‌های ماژول‌ها (مشتق نسبت به وزن‌ها) کار آسانی است [۴۹].

بسیاری از کاربردهای یادگیری عمیق از معماری شبکه عصبی روبه جلو استفاده می‌کنند که در آن، یاد می‌گیرند تا یک ورودی با اندازه ثابت را به یک خروجی با اندازه ثابت تبدیل کنند. برای رفتن از یک لایه به لایه بعد، مجموعه ای از گره‌ها (node ها) جمع وزن‌دار ورودی‌های خود را محاسبه می‌کنند و نتیجهٔ بدست آمده را به یک تابع می‌دهند؛ که اصطلاحاً به آن تابع فعال‌ساز می‌گویند و نتیجهٔ این تابع را به عنوان ورودی به مرحلهٔ بعد می‌فرستند. انواع مختلفی از این توابع غیر خطی وجود دارد که در حال حاضر تابع ReLU محبوب‌ترین نوع آن‌ها است. این تابع که یک اصلاح‌کننده نیم موج بوده، با ورودی z به شکل $F(z) = \max(z, 0)$ تعریف می‌شود.

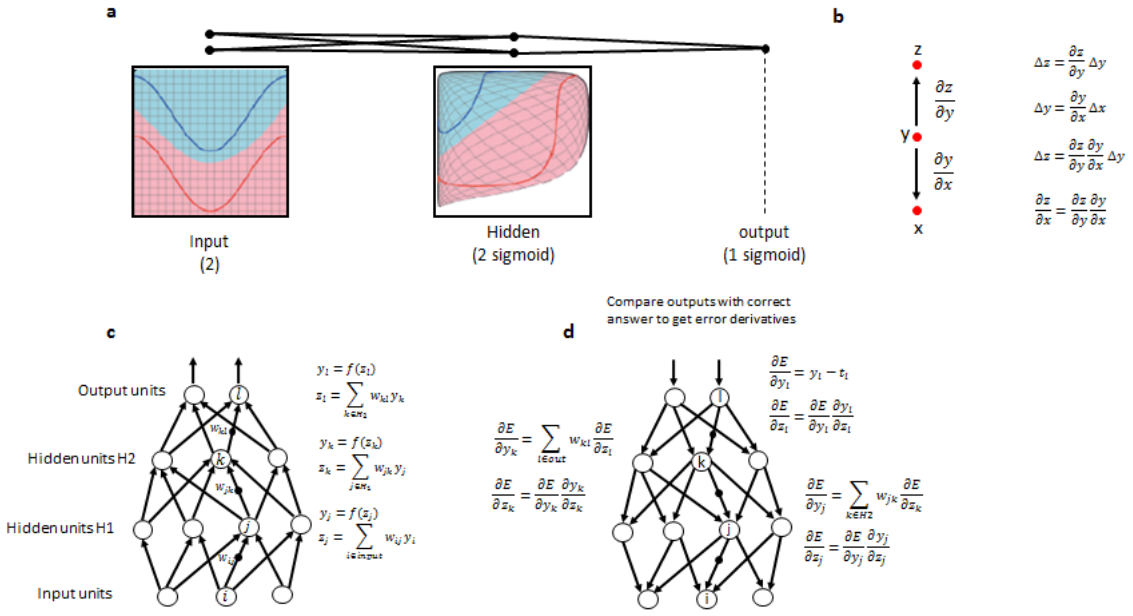



شکل (۳-۱) مقایسه دو تابع فعال‌سازی با هم [۴۹].

در شکل۲-۳ خط آبی نشان دهنده تابع ReLU است، که ورودی های قبل از صفر را به خروجی صفر تبدیل کرده و برای مقادیر ورودی بزرگتر از صفر، آن خروجی را برابر خود ورودی قرار میدهد.

در دهه های گذشته شبکه های عصبی از توابع غیر خطی هموارتری مانند Tanh(z) یا $1/(1+\exp(-z))$، با نام softplus (که در تصویر با رنگ سبز آمده است) استفاده می‌کردند اما Relu معمولاً در شبکه‌هایی با تعداد لایهٔ زیاد، منجر به یادگیری سریع‌تری می‌شود که این امر سبب می‌شود تا شبکه‌های باناظر عمیق بدون استفاده از پیش یادگیری بی‌ناظر، به خوبی آموزش داده شوند.

واحدهایی که در لایهٔ ورودی یا خروجی نیستند، عموماً لایه‌های مخفی نامیده می‌شوند. لایه‌های مخفی می‌توانند ورودی‌ها را از فضای ورودی به صورت غیرخطی به فضای خروجی ببرند که در لایهٔ آخر، بتوانیم کلاس‌های مختلف را به راحتی و با یک تابع خطی از هم جدا کنیم.




شکل (۳-۲) شبکه‌های عصبی چندلایه و پس‌انتشار [۴۹].

در شکل ۲-۴ مشاهده می‌کنیم که در قسمت a، یک شبکه عصبی چندلایه (که با نقاط پیوسته مشخص شده است) می‌تواند فضای ورودی را اعوجاج دهد تا کلاس‌های مختلف داده (که مثال‌های مختلف آن روی خطهای آبی و قرمز هستند) را به صورت خطی، جداپذیر کند. دقت کنید چگونه یک توری منظم در فضای ورودی (که در سمت چپ می‌بینید) به وسیله واحدهای مخفی (که در وسط نشان داده شده است) تبدیل می‌شود. این یک مثال گویا با تنها دو ورودی، دو لایه مخفی و یک واحد خروجی می‌باشد؛ اما شبکه‌هایی که برای تشخیص اشیا و یا پردازش زبان طبیعی استفاده می‌شوند، از ده‌ها یا صدها هزار واحد تشکیل شده‌اند.

درقسمت B، قانون زنجیره‌ای مشتقات به ما می‌گوید که چگونه دو اثر کوچک (اثر یک تغییر کوچک x روی y و اثر کوچک y روی z) با هم ترکیب می‌شوند. یک تغییر کوچک $\Delta x$ در x، در ابتدا به وسیله ضرب شدن در $\partial y/\partial x$ به تغییر کوچک $\Delta y$ در y تبدیل می‌شود(که این تعریف مشتق جزیی می‌باشد). به طور م-شابه تغییر $\Delta y$ باعث تغییر $\Delta z$ در z می‌شود. جایگزین کردن یک معادله در

دیگری، قانون زنجیره‌ای مشتقات را می‌دهد: یعنی چگونه $\Delta x$ از طریق ضرب در $\partial y/\partial x$ و $\partial x/\partial z$ به $\Delta z$ تبدیل می‌شود. این معادلات زمانی که x و y و z به شکل برداری باشند هم کار می‌کند (و مشتقات همان ماتریس‌های ژاکوبین هستند).

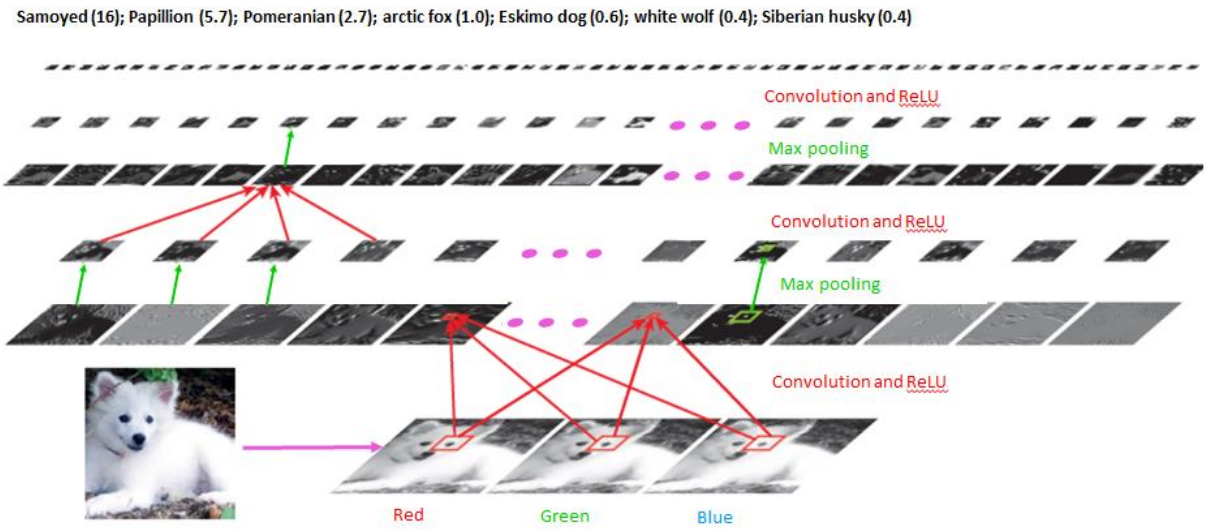
در قسمت c، معادلاتی که برای محاسبه گذر رو به جلو در شبکه عصبی با دو لایه مخفی و یک لایه خروجی صورت می‌گیرد را مشاهده می‌کنیم. هر لایه مدلی را تشکیل می‌دهد که از طریق آن می‌تواند گرادیان‌ها را پس‌انتشار کند. در هر لایه ابتدا ورودی‌های z برای هر واحد محاسبه می‌شود که یک جمع وزن‌دار از خروجی‌های واحدهای لایه‌های پایین‌تر می‌باشد. برای سادگی، عبارات بایاس را حذف کرده‌ایم. توابع غیرخطی که در شبکه‌های عصبی استفاده می‌شود مانند: واحد اصلاح‌کننده خطی (ReLU) $f(z) = \max(0, z)$ (که عموماً در سال‌های اخیر استفاده شده) و توابع کانولوشن سیگموئید مانند تانژانت هیپربولیک، $f(z) = \frac{exp(z)-exp(-z)}{exp(z)-exp(-z)}$ و تابع لجستیک $f(z) = 1/(1 + \exp(-z))$ است [۴۹].

در قسمت D، معادلات استفاده شده برای محاسبه گذر روبه عقب را می‌بینیم. در هر لایهٔ مخفی، مشتق خطا نسبت به خروجی هر واحد محاسبه می‌شود که برابر با جمع وزن‌دار مشتقات خطا نسبت به همه ورودی‌های وارد شده به آن واحد، در لایه مخفی است. سپس مشتقات خطا نسبت به خروجی را با ضرب کردن آن در گرادیان f(z)، به مشتقات خطا نسبت به ورودی تبدیل می‌کنیم. در لایه خروجی، مشتق خطا نسبت به خروجی یک واحد، به وسیله مشتق‌گیری از تابع هدف به دست می‌آید. به طور مثال فرض کنید در طبقه $l$، تابع هدف ما، برابر $0.5(y_l - t_l)^2$ باشد، که در آن $t_l$ مقدار مطلوب باشد. در اینصورت، مشتق خطا در این لایه، برابر $y_l - t_l$ هست. به این ترتیب، هنگامی که $\partial E/\partial z_k$ معلوم باشد، مشتق خطا برای وزن $w_{jk}$ که روی اتصال از واحد j از لایه زیرین است برابر است با $y_j \partial E/\partial z_k$ .

### ۳-۱-۳- شبکه‌های کانولوشن

شبکه‌های کانولوشن به منظور پردازش داده‌های به شکل آرایه‌های چندگانه، طراحی شده‌اند. برای مثال می‌توان به یک تصویر رنگی اشاره نمود که از سه آرایه ۲ بعدی تشکیل شده است، که هر کدام از آن‌ها شامل شدت آن پیکسل در هر یک از کانال‌های سه‌گانه است. بسیاری از شکل‌های داده در فرم ارائه‌های چندتایی هستند: ۱بعدی برای سیگنال‌ها و دنباله‌ها مانند زبان، ۲بعدی برای عکس‌ها یا پرتونگاری رادیویی و ۳ بعدی برای تصاویر ویدیویی یا حجمی. چهار ایده کلیدی در پشت شبکه‌های کانولوشن وجود دارد که از فواید ویژگی‌های سیگنال‌های طبیعی استفاده می‌کند که عبارتند از اتصالات محلی، وزن‌های مشترک، ادغام و استفاده از لایه‌های متعدد [۴۹].

معماری یک شبکه کانولوشن معمولی (شکل ۲-۵) به شکل تعدادی طبقه می‌باشد. چند طبقه ابتدایی از دو نوع لایه تشکیل شده‌اند: لایه‌های کانولوشن و لایه‌های ادغام(pooling). در لایه‌های کانولوشن، واحدها در نقشه‌های ویژگی سازماندهی شده‌اند که در درون هر یک از این نقشه‌ها، هر واحد به پچ‌های محلی در نقشه ویژگی لایه قبل متصل شده است که این اتصال از طریق مجموعه‌ای از وزن‌ها به نام بانک فیلتر انجام می‌پذیرد. در واقع ورودی لایهٔ قبل، از طریق این فیلترها (به وسیله جمع وزن‌دار)، به تعدادی خروجی تبدیل می‌شوند. سپس نتیجه این جمع وزن‌دار محلی، از طریق یک تابع غیر خطی مانند ReLU ارسال می‌شود.




شکل (۳-۳) ساختار درونی یک شبکه کانولوشن. خروجی‌های هر لایه (نه فیلترها) از یک شبکه کانولوشن معمولی، که به تصویر یک سگ Samoyed اعمال شده است. (پایین راست و ورودی‌های RGB پایین راست). هر تصویر مستطیلی یک نقشه ویژگی است [۴۹].

همهٔ واحدها در نقشه ویژگی، یک بانک فیلتر مشترک دارند. استفاده از این معماری دو دلیل دارد: ابتدا اینکه در داده‌های آرایه‌ای مانند تصویر، مقادیری که از لحاظ محلی به هم نزدیک هستند، معمولاً به شدت نیز به یکدیگر وابستگی دارند و یک موضوع (موتیف) متمایز محلی را تشکیل می‌دهند که به راحتی تشخیص داده می‌شود. مثلاً گروهی از پیکسل‌های نزدیک به هم در تصویر، یک لبه را نشان می‌دهند. با بررسی این پیکسل‌ها به طور همزمان، می‌توان آن لبه را به راحتی تشخیص داد.

دوم اینکه آماره‌های محلی تصاویر و سیگنال‌های دیگر، نسبت به مکان تغییر ناپذیرند. به عبارت دیگر، اگر یک موضوع بتواند در یک قسمت اصلی از یک تصویر ظاهر شود، می‌تواند در همه جا ظاهر شود. به طور مثال، با یافتن شیوه قرارگیری پیکسل‌هایی که لبه را نشان می‌دهند می‌توان آن را به همهٔ لبه‌ها تعمیم داد. پس می‌توانیم این ایده را نتیجه بگیریم که واحدها در مکان‌های مختلف وزن‌های مشترکی دارند، پس می‌توانند الگوهای یکسان را در مکان‌های مختلف آرایه تشخیص دهند. به همین دلیل است که از لحاظ ریاضی، عملیات فیلتر کردن که توسط یک نقشه ویژگی انجام می‌شود را

کانولوشن گسسته می‌نامند.

همانطور که نقش یک لایه کانولوشن، تشخیص پیوستگی‌های محلی ویژگی‌های لایه قبلی است، نقش لایه ادغام، ترکیب معنایی ویژگی‌های مشابه با هم می‌باشد. به این دلیل که مکان نسبی ویژگی‌هایی که یک موضوع(موتیف) را تشکیل می‌دهند، می‌تواند تا حدی تغییر کند، تشخیص مطمئن یک موضوع(موتیف) به وسیله Coarse-Graining مکان هر ویژگی انجام شود [۴۹].

یک واحد ادغام معمولی، بیشینه یک پچ محلی از واحدها را در یک یا چند نقشه ویژگی محاسبه می‌کند. واحدهای ادغامی نزدیک به هم، ورودی‌هایشان را از پچ‌هایی که به اندازه بیش از یک ردیف یا یک ستون انتقال داده شده‌اند، گرفته و به این ترتیب، ابعاد بازنمایی را کاهش می‌دهد. از این طریق سیستمی تغییر ناپذیر نسبت به جابه‌جایی‌ها و اعوجاج‌های کوچک ساخته می‌شود.

دو یا سه طبقه کانولوشن، طبقه غیرخطی و طبقه ادغام، پشت سر هم قرار دارند، که بعد از آن لایه‌های کانولوشن با اتصالات کامل، قرار می‌گیرند. انتشار رو به عقب گرادیان، از طریق شبکه کانولوشن، می‌تواند به همان سادگی انتشار رو به عقب از طریق یک شبکه عمیق معمولی، همه وزن‌ها را در همه بانک‌های فیلتر آموزش دهد.

از ویژگی سلسله مراتبی بودن سیگنال‌های طبیعی که در آن ویژگی‌های سطح بالاتر، از ترکیب ویژگی‌های سطوح پایین‌تر ساخته می‌شوند، در شبکه‌های عصبی عمیق استفاده می‌شود. در تصاویر، ترکیب محلی لبه‌ها، شکل‌ها را تشکیل می‌دهد. شکل‌ها با هم ترکیب می‌شوند و اجزاء را تشکیل می‌دهند و در نهایت اجزاء با هم ترکیب شده و اشیاء را به وجود می‌آورند. سلسله مراتب مشابهی در صوت و متن نیز وجود دارد؛ از اصوات گرفته تا آواها، واج‌ها، سیلاب‌ها، کلمه‌ها و جمله‌ها. ادغام باعث می‌شود تا هنگامی که المان‌های لایه قبلی از نظر مکانی و ظاهری تغییر کردند، بازنمایی‌ها به مقدار بسیار کمی تغییر کنند.

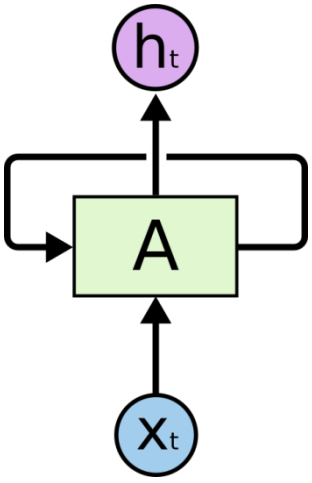
لایه‌های کانولوشن و ادغام در شبکه‌های کانولوشن به صورت مستقیم، تحت تأثیر مفهوم سلول‌های ساده و سلول‌های پیچیده در علوم عصب‌شناسی بصری قرار دارند [۴۹].

### ۳-۱-۴- شبکه‌های عمیق بازگشتی

اگر به نحوه تفکر در انسان‌ها دقت کنیم، متوجه می‌شویم که روند فکری در انسان پیوسته و دنباله‌دار است و در هر ثانیه از ابتدا شروع نمی‌شود. برای مثال در همین لحظه که این مطلب را می‌خوانید، معنی هر کلمه را با توجه به دانشی که از خواندن کلمه‌های قبلی کسب کرده‌اید متوجه می‌شوید. به عبارت دیگر موقع خواندن یک متن، درکی را که از آن متن با خواندن کلمات قبل کسب کرده‌اید، دور نمی‌ریزید و به صورت پیوسته با خواندن هر کلمهٔ جدید، نسبت به متن درک و فهم پیدا می‌کنید و به عبارتی معنی آن متن را متوجه می‌شوید.

شبکه‌های عصبی متداولی که تا کنون متخصصان یادگیری ماشین از آن‌ها استفاده می‌کردند، نمی‌توانستند به این صورت مشابه انسان عمل کنند. این یک نقصان بزرگ برای این شبکه‌ها در پیش‌بینی گام‌های بعدی با توجه به گام‌های گذشته محسوب می‌شود [۵۰].

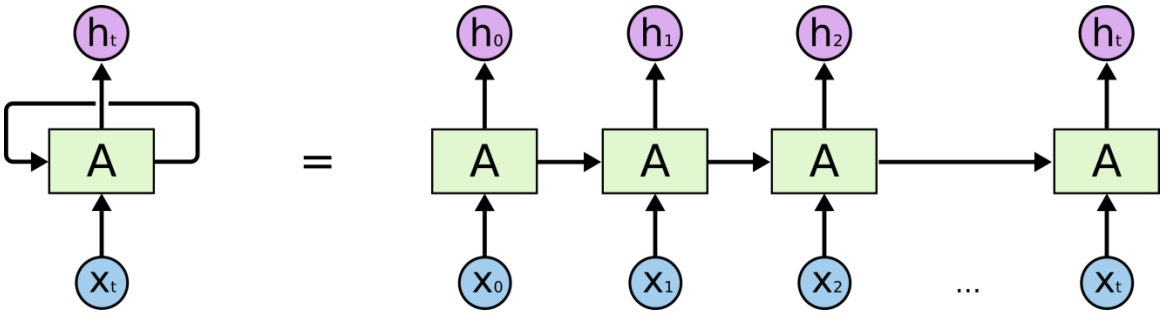
شبکه‌های عصبی بازگشتی[1] برای برطرف کردن این مشکل طراحی شدند. این شبکه‌ها در ساختار خود شامل یک حلقه بازگشتی هستند که منجر به حفظ و ماندگاری اطلاعات بدست آمده از گام‌های گذشته، در شبکه می‌شود. در شکل ۲-۶ ساختار کلی این شبکه را مشاهده می‌کنید.




شکل (۳-۴) شبکه‌های عصبی بازگشتی در ساختار خود دارای حلقه هستند [۵۰].

[1] Recurrent Neural Network (RNN)

در شکل بالا، بخش A، به عنوان ورودی مقدار xt را دریافت و مقدار ht را به خروجی مـی‌بـرد. حلقـهٔ موجود در شبکه سبب می‌شود تا اطلاعات بدست آمده از یک مرحله به مرحلهٔ بعد ارسـال شـود. ایـن شبکه‌ها عملاً تفاوت خاصی با شبکه‌های عصبی معمولی ندارند. شبکه‌های عصبی بازگشتی را می‌توان به صورت چندین کپیِ یکسان از یک شبکهٔ عصبی در نظر گرفت که هر کدام اطلاعاتش را بـه شـبکه بعدی منتقل می‌کند. در شکل ۲-۷ وضعیت شبکهٔ عصبی بازگشتی در صورت باز کردن حلقه نمایش داده شده است.




شکل (۳-۵) شبکهٔ عصبی بازگشتی باز شده [۵۰].

با توجه به ذات زنجیره‌مانند شبکه‌های عصبی بازگشتی، می‌توان تشخیص داد کـه ایـن شـبکه‌هـا بـه مقدار زیادی به دنباله‌ها و لیست‌ها مرتبط هستند. در حقیقـت شـبکه‌هـای عصـبی بازگشـتی اولـین انتخاب برای کار با چنین داده‌هایی هستند. در سال‌های اخیر به صورت مکرر از این شبکه‌ها اسـتفاده شده است که موفقیت‌های بسیار چشم‌گیری در حوزه‌های مختلف از جمله مدل کردن زبان، تشخیص صدا، ترجمه، درج خودکار توضیح برای تصویر و غیره شده است.

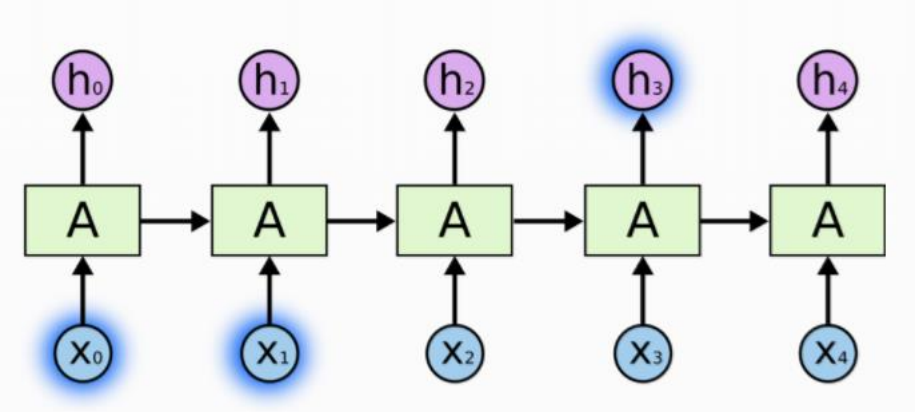
بیشتر این موفقیت‌ها مدیون استفاده از شبکه‌های حافظهٔ بلند-کوتاه مدت است که به اختصار به آن‌ها "LSTM" گفته می‌شود. این شبکه‌ها نوع خاصی از شبکه‌های عصبی بازگشتی هستند که در بیشتر موارد عملکرد بهتری از شبکه‌های عصبی بازگشتی استاندارد دارند. با تقریب نسبتاً خوبی

می‌توان گفت که اکثر موفقیت‌های شبکه‌های عصبی بازگشتی پس از معرفی و استفاده از شبکه‌های LSTM بدست آمده است. در ادامه شبکه‌های LSTM را به تفسیر شرح می‌دهیم [۵۰].

### ◘ مشکلی به نام وابستگی‌های بلندمدت

یکی از جذابیت‌های شبکه‌های عصبی بازگشتی، مرتبط ساختن اطلاعاتی که از قبل توسط شبکه مشاهده شده با کاری که در حال حاضر قصد انجام آن را دارند، است. برای مثال استفاده از فریم‌های قبلی یک ویدئو می‌تواند در فهم فریم کنونی آن کمک‌کننده باشد. این که تا چه میزان یک شبکه عصبی بازگشتی می‌تواند در انجام این کار موفق باشد، به پاره‌ای از مسائل بستگی دارد.

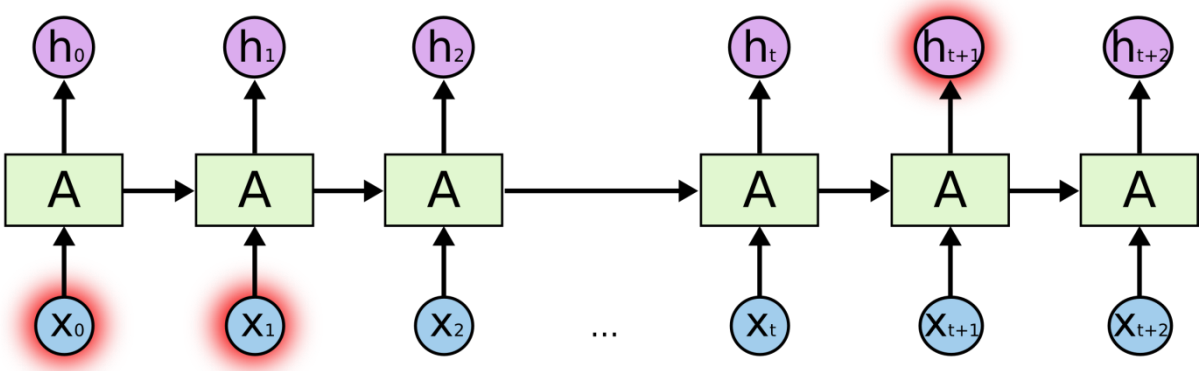
گاهی تنها ما نیاز داریم فقط به اطلاعات گذشتهٔ نزدیک نگاه کنیم تا اطلاعات حال حاضر را درک کنیم. برای مثال، فرض کنید ما یک مدل زبانی ساخته‌ایم که تلاش می‌کند کلمهٔ بعدی را با توجه به کلمات قبلی که در اختیار دارد، پیش‌بینی کند. اگه ما بخواهیم آخرین کلمه در جملهٔ «ابرها هستند در آسمان» را پیش‌بینی کنیم، به اطلاعات اضافی دیگری نیاز نداریم و تقریباً واضح است که کلمهٔ بعدی «آسمان» است. در موارد مشابه این مثال، که فاصله بین اطلاعات مرتبط و جایی که به این اطلاعات نیاز داریم بسیار کوتاه باشد، شبکه‌های عصبی بازگشتی می‌توانند استفاده از این اطلاعات را یاد بگیرند.




شکل (۳-۶) شبکهٔ عصبی بازگشتی با یادگیری از اطلاعات گذشتهٔ نزدیک [۵۰].

در مقابل این حالت شرایطی وجود دارند که شبکه به اطلاعات بیشتری برای یادگیری نیاز دارد. به عنوان مثال درنظر بگیرید که قصد داریم تا کلمهٔ آخر در جملهٔ «من زبان فرانسه را خیلی راحت صحبت می‌کنم... من به دنیا آمدم در فرانسه.» را با توجه به اطلاعات گذشتهٔ نزدیک (یعنی چهار تا پنج کلمه قبل از آخرین کلمه) پیش‌بینی کنیم. با توجه به ساختار جمله می‌توان پیش‌بینی کرد که کلمهٔ آخر به احتمال زیاد نام یک کشور است، اما اگر بخواهیم به طور دقیق نام کشور را حدس بزنیم، به اطلاعات دورتر (یعنی ده یا بیست کلمه قبل از آخرین کلمه) نیاز داریم. پس به صورت کلی ممکن است فاصلهٔ بین اطلاعات مرتبط و مکانی که به این اطلاعات نیاز است، زیاد باشد. شکل ۲-۹ نمونه‌ای از این مسأله را نشان می‌دهد.

متأسفانه، هر چه این فاصله افزایش پیدا می‌کند، شبکه‌های عصبی بازگشتی قدرت خود را در به یادآوردن و استفاده از اطلاعاتی که در گذشتهٔ دورتر یاد گرفته‌اند، از دست می‌دهند. به عبارت دیگر این شبکه‌ها توانایی استفاده از اطلاعات گذشتهٔ دورتر را ندارند.




شکل (۳-۷) شبکهٔ عصبی بازگشتی نیازمند یادگیری از اطلاعات گذشتهٔ دور [۵۰].

از نظر تئوری، شبکه‌های عصبی بازگشتی باید توانایی مدیریت وابستگی‌های بلندمدت را از طریق تنظیم دقیق پارامترهای شبکهٔ عصبی داشته باشند؛ اما متأسفانه در عمل شبکه‌های عصبی بازگشتی

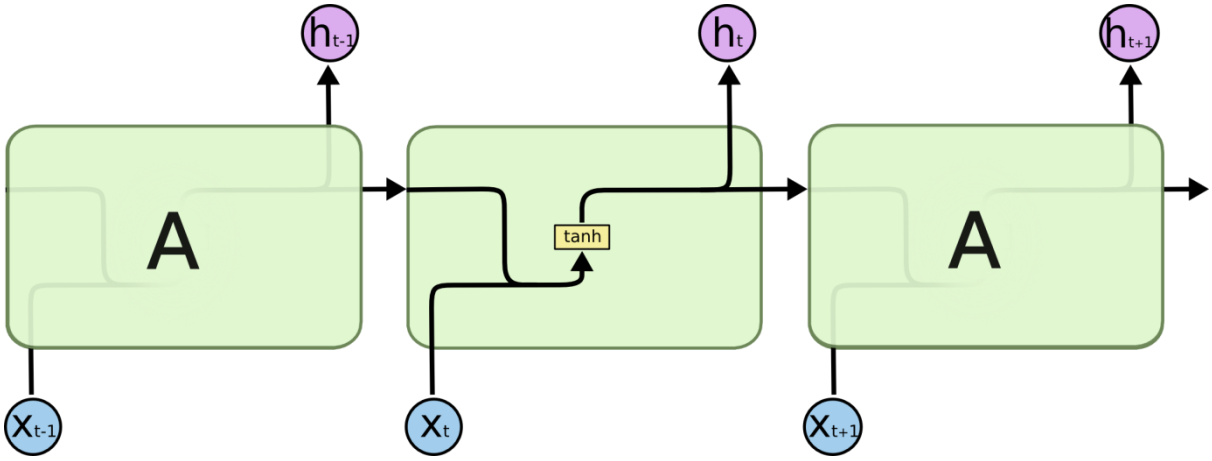
توانایی یادگیری وابستگی بلندمدت را ندارند. این مسأله از دو مشکل اصلی در این شبکه‌ها با نام Vanishing وExploding Gradient نشأت می‌گیرد. برای حل این مشکلات، شبکه‌های حافظهٔ بلند-کوتاه مدت یا همان LSTM پدید آمدند [۵۰].

### ۳-۱-۵- شبکه‌های LSTM

شبکه‌های LSTM که خلاصه شدهٔ عبارت "Long Short Term Memory" هستند، نوع خاصی از شبکه‌های عصبی بازگشتی هستند که توانایی یادگیری وابستگی‌های بلندمدت را دارند. این شبکه‌ها برای اولین بار توسط Hochreiter و Schmidhuber در سال ۱۹۹۷ معرفی شدند. پس از آن نیز تعداد زیادی از محققان در بهبود این شبکه‌ها نقش داشته‌اند.

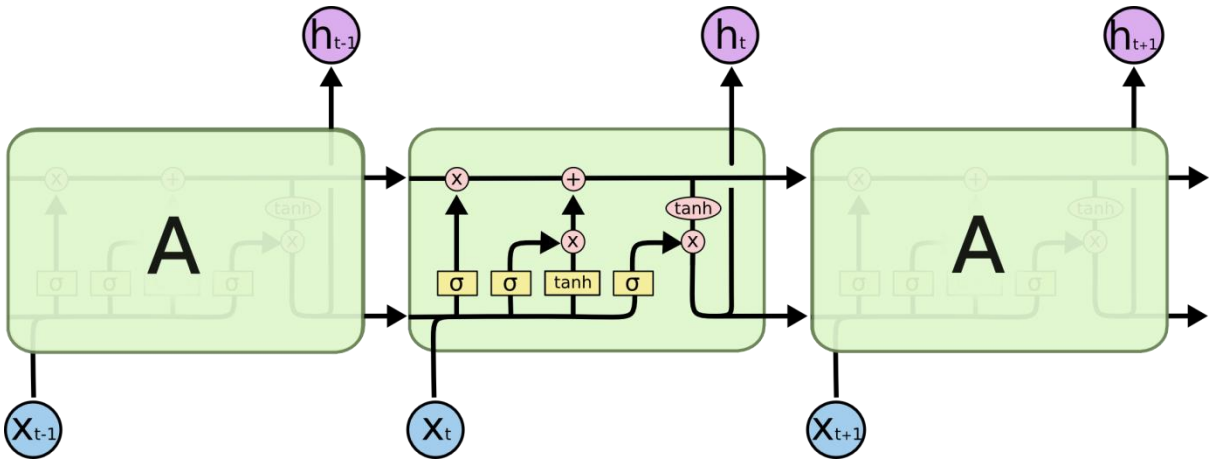
همان‌طور که گفته شد، هدف از طراحی شبکه‌های LSTM حل مشکل وابستگی بلندمدت بود. به یاد سپردن اطلاعات برای بازه‌های زمانی بلند مدت، رفتار پیش‌فرض و عادی شبکه‌های LSTM است. در حقیقت ساختار این شبکه‌ها به صورتی است که می‌توانند اطلاعات خیلی دور را به خوبی یاد بگیرند [۵۰].

تمامی شبکه‌های عصبی بازگشتی به شکل دنباله‌ای تکرارشونده از ماژول‌های شبکه‌های عصبی هستند. در شبکه‌های عصبی بازگشتی استاندارد، این ماژول‌های تکرار شونده ساختار ساده‌ای دارند. به عنوان مثال تنها شامل یک لایهٔ تانژانت هایپربولیک هستند. شکل ۲-۱۰ ساختار کلی این شبکه‌ها را نشان می‌دهد.




شکل (۳-۸) شبکه‌های عصبی بازگشتی استاندارد شامل یک لایه در هر ماژول هستند [۵۰].

شبکه‌های LSTM نیز از چنین ساختار زنجیره‌مانندی پیروی می‌کنند با این تفاوت که ماژولِ تکرار شوندهٔ آن‌ها ساختار متفاوتی دارد. این شبکه‌ها به جای داشتن تنها یک لایه، چهار لایهٔ شبکه عصبی دارند که طبق ساختار ویژه‌ای با یکدیگر در تعامل و ارتباط هستند. ساختار شبکه‌های LSTM را در شکل ۲-۱۱ می‌بینید.

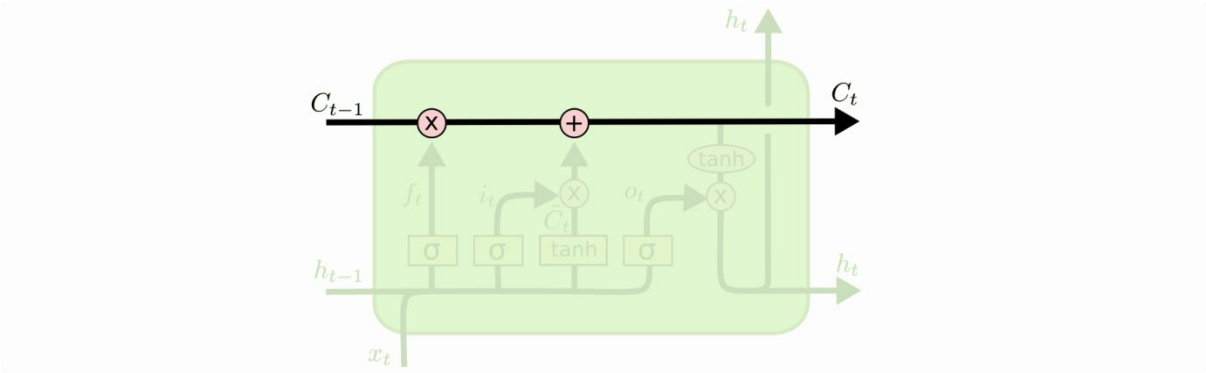



شکل (۳-۹) ساختار کلی یک شبکهٔ عصبی بازگشتی LSTM [۵۰].

در در شکل ۲-۱۱، هر خط یک بردار را به صورت کامل از خروجی یک گره به ورودی گره دیگر انتقال می‌دهد. دایره‌های صورتی نمایش دهنده عملیات‌های نقطه به نقطه مانند «جمع دو بردار» هستند. مستطیل‌های زرد، لایه‌های شبکه‌های عصبی هستند که شبکه پارامترهای آن‌ها را یاد می‌گیرد. خط‌هایی که با هم ادغام می‌شوند، نشان‌دهنده الحاق و خط‌هایی که چند شاخه می‌شوند، نشان دهندهٔ این است که محتوای آن‌ها کپی و به بخش‌های مختلف ارسال می‌شود [۵۰].

### ◘ ایده اصلیِ پشت LSTMها

عنصر اصلی LSTMها سلول حالت[1] است که در حقیقت یک خط افقی است که در بالای شکل قرار دارد. شکل ۲-۱۲ یک سلول حالت را نشان می‌دهد. سلول حالت یک ساختار ساده با تغییرات کم است که می‌توان آن را به صورت یک تسمهٔ نقاله تصور کرد که از اول تا آخر زنجیره با تعاملات خطیِ جزئی در حرکت است.

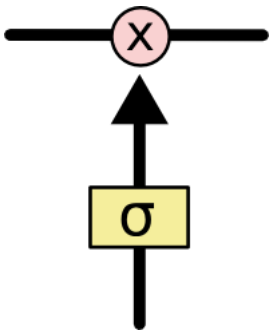



شکل (۳-۱۰) یک سلول حالت در شبکهٔ LSTM [۵۰].

شبکهٔ عصبی LSTM توانایی اضافه کردن یا حذف اطلاعات جدید را به سلول حالت دارد. این کار توسط ساختارهای دقیقی به نام دروازه[2] انجام می‌شود. دروازه‌ها راهی برای ورود اختیاری اطلاعات هستند. آن‌ها از یک لایه شبکهٔ عصبیِ سیگموئید[3] به همراه یک عملگر ضرب نقطه به نقطه تشکیل

[1] cell state
[2] Gate
[3] Sigmoid

شده‌اند. در شکل ۲-۱۳ یک دروازه را در یک شبکهٔ عصبی LSTM نشان می‌دهد.




شکل (۳-۱۱) نمایش یک دروازه در یک سلول از شبکهٔ عصبی LSTM [۵۰].

خروجی لایهٔ سیگموئید عددی بین صفر و یک است. این عدد نشان‌گر آن است که چه مقدار از وروی باید به خروجی ارسال شود. مقدار یک به معنی آن است که تمام ورودی باید به خروجی ارسال شود و در مقابل مقدار صفر نشان‌دهندهٔ آن است که هیچ اطلاعاتی از ورودی به خروجی ارسال نشود. شبکهٔ عصبی LSTM دارای سه دروازهٔ مشابه برای کنترل مقدار سلول حالت است [۵۰].

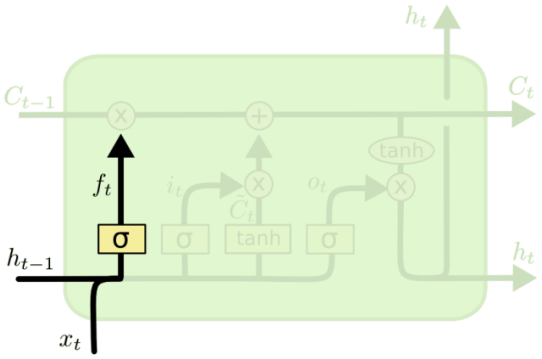
### ◘ بررسی گام به گام ورود و خروج اطلاعات به شبکهٔ LSTM

در شبکهٔ LSTM گام اول، در مورد اطلاعاتی است که می‌خواهیم آن‌ها را از سلول حالت پاک کنیم. این تصمیم توسط یک لایهٔ سیگموئید به نام دروازهٔ فراموشی[1] انجام می‌شود. این دروازه با توجه به مقادیر $h_{t-1}$ و $x_t$، برای هر عدد مقدار صفر یا یک را در سلول حالتِ $C_{t-1}$ به خروجی می‌برد. مقدار یک به صورت کامل مقدار حال حاضر سلول حالت ($C_{t-1}$) را به $C_t$ می‌برد و مقدار صفر به صورت کامل اطلاعات سلول حالت کنونی را پاک می‌کند و هیچ مقداری از آن را به $C_t$ نمی‌برد. شکل ۲-۱۴ این بخش از شبکهٔ LSTM را نشان می‌دهد.

اگر به مثال ابتدای این بخش برای پیش‌بینی کلمات برگردیم، در چنین مسأله‌ای سلول حالت ممکن است دربردارنده جنسیت فاعل کنونی باشد. با توجه به جنسیت فاعل کنونی می‌توان تشخیص داد که

[1] Forget Gate

از چه ضمیری باید استفاده کرد. حال در صورتی که یک فاعل جدید در جمله ظاهر شود، لازم است تا جنسیت فاعل قبلی حذف شود.

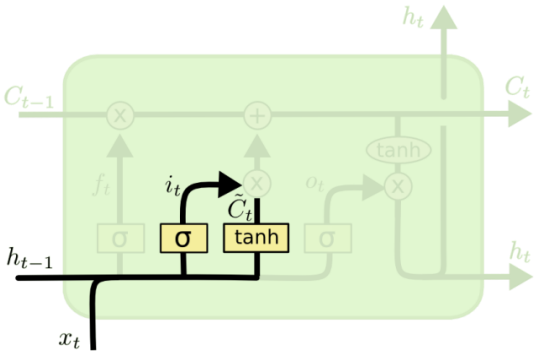



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

شکل (۳-۱۲) بخش حذف کننده اطلاعات از سلول حالت [۵۰].

در قدم بعدی تصمیم گرفته می‌شود که چه اطلاعات جدیدی در سلول حالت ذخیره شود. این تصمیم شامل دو قسمت است. ابتدا یک لایهٔ سیگموئید به نام دروازه ورودی[1] برای تصمیم‌گیری در مورد مقادیری که باید به‌روز شوند، وجود دارد. در قسمت بعدی یک لایهٔ تانژانت هایپربولیک وجود دارد که برداری از مقادیر به نام $\sim C_t$ می‌سازد که می‌توان آن‌ها را به سلول حالت اضافه کرد. در مرحلهٔ بعد، این دوبخش با هم ترکیب می‌شوند تا مقدار سلول حالت را به‌روز کنند. شکل ۲-۱۴ این بخش را نشان می‌دهد.

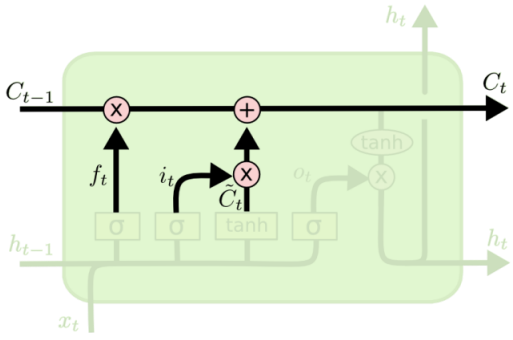
[1] Input Gate




$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

شکل (۳-۱۳) بخش اضافه کردن اطلاعات جدید در سلول حالت [۵۰].

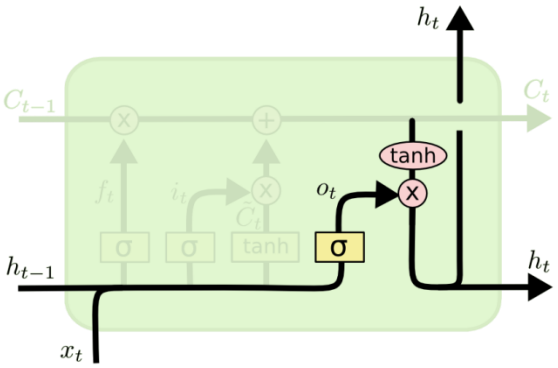
در مثال مدل زبانی که پیش‌تر بررسی کردیم، قصد داریم تا در سلول حالت، جنسیت فاعل جدید را جایگزین جنسیت فاعل قبلی کنیم. به این منظور باید مقدار سلول حالت قدیمی یعنی $C_{t-1}$ را با مقدار سلول حالت جدید، یعنی $C_t$ به‌روز کنیم. در این مرحله، مقدار قبلی سلول حالت را به منظور فراموش کردن اطلاعات قبلی، در $f_t$ ضرب می‌کنیم. سپس $i_t * \sim C_t$ را به آن اضافه می‌کنیم. به این صورت مقادیر جدید سلول حالت با توجه به تصمیمات از قبل گرفته شده، بدست آمده‌اند. در مثال مدل زبانی، شکل ۲-۱۵ دقیقاً جایی را نشان می‌دهد که اطلاعات جنسیت قبلی دور ریخته و اطلاعات جدید به سلول حالت اضافه می‌شود.




$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

شکل (۳-۱۴) به‌روز شدن اطلاعات در سلول حالت [۵۰].

در نهایت باید تصمیم گرفته شود که چه اطلاعاتی به خروجی برده شود. در واقع مقدار سلول حالت با عبور از فیلتر مشخصی تبدیل به مقدار خروجی خواهد شد. در ابتدا، یک لایهٔ سیگموئید وجود دارد که مشخص می‌کند چه بخشی از سلول حالت، به خروجی برده شود. سپس مقدار سلول حالت به‌روز شده، به یک لایهٔ تانژانت هایپر بولیک (تا مقادیر) داده می‌شود تا مقادیر آن به بازهٔ بین ۱- و ۱+ نرمال شود. سپس مقدار بدست آمده در خروجی لایهٔ سیگموئید قبلی ضرب می‌شود تا تنها بخش‌های مد نظر به خروجی برده شود. شکل ۲-۱۷ نحوهٔ بدست آمدن خروجی سلول حالت را در این شبکه نشان می‌دهد.




$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

شکل (۳-۱۵) خروجی سلول حالت در شبکهٔ LSTM [۵۰].

در مثال مدل زبانی، اگر تنها فاعل را دیده باشیم، در صورتی که بخواهیم کلمهٔ بعدی را حدس بزنیم، شبکه ممکن است نیاز داشته باشد تا اطلاعاتی در ارتباط با فعل را به خروجی ببرد. برای مثال، ممکن است مفرد یا جمع بودن فاعل را به خروجی ببرد، تا از طریق آن فرم فعل مشخص شود [۵۰].

## ۳-۲- یادگیری عمیق در طبقه‌بندی سؤالات

همانطور که گفتیم، اخیراً استفاده از شبکه‌های عمیق در دسته‌بندی متون و جملات شروع شده است. در این زمینه تعدادی مقاله وجود دارد که به بررسی آن‌ها می‌پردازیم. در [۲۵] تعدادی آزمایش با شبکه‌های عصبی کانولوشن (CNN) که با بردارهای کلمات از پیش آموزش یافته، آموزش داده

شده‌اند، برای انجام وظایف طبقه‌بندی در سطح جمله انجام شده است. در این پژوهش نشان داده شد که یک CNN ساده با تنظیمات کم ابرپارامترها و بردارهای استاتیک می‌تواند نتایج بسیار خوبی روی چندین معیار ارزیابی داشته باشد. یادگیری بردارهای با وظیفه خاص از طریق تنظیم دقیق، بهبود بیشتری را در کارایی حاصل می‌کند. در این پژوهش علاوه بر این، یک اصلاح ساده برای معماری پیشنهاد می‌شود تا امکان استفاده از هر دو بردار وظیفه خاص و استاتیک وجود داشته باشد.

در [۲۶] یک معماری شبکه عصبی عمیق برای پردازش زبان طبیعی پیشنهاد شده است. این معماری کمک می‌کند تا بتوان روی پایگاه داده‌های بزرگ با سرعت بالایی نتیجه گرفت. شبکه عصبی عمیق پیشنهاد شده می‌تواند در کارهای مختلفی مانند قطعه‌بندی، برچسب‌گذاری و مدل‌سازی زبان استفاده شود. همچنین در یادگیری هم‌زمان می‌تواند عملکرد کلی را بهبود ببخشد. به طور خاص، هنگام آموزش وظیفهٔ مدل‌سازی به طور مشترک با زبان این مدل، این معماری بدون هیچ گونه ویژگی صریح نحوی به عملکرد مطلوبی رسیده است.

در [۲۷] یک شبکه عصبی کانولوشنال داینامیک برای مدل‌سازی معنایی جمله پیشنهاد شده است. این شبکه وابسته به درخت تجزیه نبوده و به راحتی قابل اعمال روی سایر زبان‌ها است. این شبکه جملات ورودی با طول‌های مختلف را پردازش می‌کند و یک گراف ویژگی روی جملات تشکیل می‌دهد که قادر است به طور صریح بازه کوتاه تا بلند ارتباطات بدست آورد. DCNN در چهار آزمایش مورد بررسی قرار گرفته است: پیش‌بینی مقیاس دو بعدی و چندطبقه‌ای، طبقه‌بندی سؤال شش راهه و پیش‌بینی نظرسنجی توئیتر توسط نظارت از راه دور. این شبکه در سه آزمایش اول عملکرد عالی و در آخرین آزمایش کاهش بیش از ۲۵٪ خطا با توجه به قوی‌ترین مبنای قبلی را ارائه می‌دهد.

با توجه به اینکه ساختن کامپیوترهایی که قادر به پاسخ‌گویی به سؤالات کاربران در هر زمینه‌ای باشند، از اهداف دائمی هوش مصنوعی است، به تازگی پیشرفت‌های چشم‌گیری به وسیله روش‌های یادگیری ماشین برای نگاشت سؤالات به فرم منطقی یا کوئری‌های پایگاه داده شده است. در این مقاله ما یک رویکرد برای یادگیری نگاشت سؤالات به فرم نمایش برداری از ویژگی‌ها ارائه کرده‌ایم. با

نگاشت پاسخ‌ها به همان فضا، می‌توان هر کوئری از پایگاه داده را مستقل از طرح آن و بدون نیاز به گرامر و واژگان، بدست آورد.

در [۲۸] یک چهارچوب جدید برای آموزش پاسخ‌گویی در سیستم‌های پرسش و پاسخ دامنه باز با نظارت بسیار کم معرفی می‌شود. با استفاده از جانشینی به عنوان محور اصلی کار، این رویکرد می‌تواند روی داده‌های به صورت ناقص برچسب‌گذاری شده با موفقیت آموزش داده شود و این روش به طور قابل توجهی از روش‌های قبلی برای پاسخگویی به سؤالات حقیقی ساده بهتر عمل می‌کند. این روش تنها روش موجود است که قادر است روی داده‌های مشابهی که ضعیف برچسب‌گذاری شده‌اند، آموزش ببیند. با وجود این نتایج امیدوارکننده، برخی چالش‌ها مانند تعمیم روش به مدل سؤالات با معنای پیچیده‌تر، همچنان بی‌پاسخ مانده است. با توجه به استفاده از آموزش با نظارت بسیار کم، کار ما تنها بروی سؤالات حقیقی بسیار ساده، دارای پاسخ رضایت‌بخش است.

در [۲۹] یک سیستم ارائه می‌شود که یاد می‌گیرد به سؤالات در طیف گسترده‌ای از موضوعات در یک پایگاه دانش پاسخ دهد. این مدل از بازنمایی کلمات با ابعاد پایین و مفاهیم پایه یاد می‌گیرد. این بازنمایی برای امتیازدهی سؤالات زبان طبیعی برای یافتن پاسخ کاندید، استفاده می‌شود.

در این پژوهش یک مدل نمایش کلمات ارائه شده است که یاد می‌گیرد تا یک سیستم پرسش‌و پاسخ دامنه باز را با استفاده از داده‌های آموزش داده شده از زوج سؤال و پاسخ آن‌ها و از یک پایگاه دانش که ساختار پاسخ‌ها را ارائه دهد می‌دهد، به عملکرد قابل قبولی روی دیتاست معیار WebQuestions برساند.

روش‌های طبقه‌بندی متن برای کار پاسخ‌گویی به سؤالات حقیقی، عموما از تعریف قوانین تطبیق رشته یا نمایش کیسه کلمات استفاده می‌کنند. این روش‌ها هنگامی که سؤالات شامل تعداد کمی کلمه کلیدی که نشان‌دهنده پاسخ هستند، باشد ناکارآمد است. در [۳۰] یک مدل شبکه عصبی بازگشتی (rnn) معرفی شده است که می‌تواند روی چنین دادگانی، با استفاده از مدل ترکیبی متنی استنتاج کند. برخلاف مدل‌های قبلی، مدل rnn نمایش را در سطح کلمات و عبارات می‌آموزد که

جملات را با توجه به موجودیت‌های آن‌ها ترکیب می‌کند. این مدل بهتر از روش‌های پایه کار می‌کند و هنگامی که با روش‌های بازیابی اطلاعات ترکیب می‌شود با بهترین کاربران انسانی رقابت می‌کند. در نهایت، نشان داده شده است که نمایش در سطح جمله می‌تواند به صورت آسان و موثر ترکیب شده و نمایش در سطح پاراگراف را با قدرت پیش‌بینی بیشتری نسبت به جملات مجزا تولید کند.

## ۳-۳- جمع‌بندی

در این فصل مروری بر ساختار شبکه‌های عمیق داشتیم و پژوهش‌های انجام شده در زمینه طبقه‌بندی سؤالات به کمک شبکه‌های عمیق را بررسی کردیم. در فصل بعد روش پیشنهادی خود را که مبتنی بر شبکهٔ عصبی عمیق است، ارائه خواهیم نمود.

# فصل ۴:

## روش تحقیق

## ۴-۱- مقدمه

در فصل پیش، مروری بر روش‌های طبقه‌بندی پرسش‌ها و شبکه‌های عمیق و کاربرد آن‌ها در سیستم‌های پرسش و پاسخ داشتیم. دیدیم که استفاده از روش‌های عمیق در این حوزه باعث پیشرفت چشم‌گیری در این زمینه شده است. در این فصل روش پیشنهادی خود را بر مبنای این رویکرد ارائه می‌دهیم، راهکار پیشنهادی ما با استفاده از روش بازنمایی word2vec در قالب یک شبکه عمیق، طراحی شده است. دراین فصل ابتدا به تعریف دقیق مسأله پرداخته سپس با بررسی دقیق هر یک از اجزاء روش پیشنهادی، مدل خود را به تفصیل ارائه می‌دهیم.

## ۴-۲- طرح مسأله

در فصل پیش، به بررسی روش‌های استخراج ویژگی و طبقه‌بندی سؤال در سیستم‌های پرسش و پاسخ پرداختیم. با توجه به امکان وجود ابهام در سؤالات کاربران، پایین بودن دقت در سیستم‌های خودکار پرسش و پاسخ و همچنین وجود تعداد زیاد و یا بسیار کم پاسخ، بهره بردن از تعامل با کاربران می‌تواند بسیار مفید واقع شود.

از سوی دیگر، امروزه استفاده از تکنیک‌های یادگیری عمیق در حوزه‌های مختلف، از جمله پردازش زبان‌های طبیعی، نمود فراوانی پیدا کرده است. در پژوهش‌های اخیر در سیستم‌های پرسش و پاسخ از این ابزار برای طبقه‌بندی سؤالات استفاده شده است. با توجه به مطالب گفته شده، در نظر داریم یک شبکه عصبی عمیق برای طبقه‌بندی سؤالات در سیستم پرسش‌و پاسخ تعاملی طراحی کنیم.

## ۴-۳- روش پیشنهادی

همان‌طور که پیش از این اشاره شد، یکی از چالش‌های مهم در طبقه‌بندی سؤال، استخراج ویژگی است. روش‌های موجود در این زمینه، به دلیل کوتاه بودن سؤالات و در نتیجه کم بودن اطلاعات موجود در آن‌ها از دقت خوبی برخوردار نیستند. طبق پژوهش‌های انجام شده در حوزه‌های مختلف مانند پردازش متن و تصویر، یادگیری عمیق ابزاری کارآمد برای استخراج ویژگی است. همچنین در

زمینه دسته‌بندی سؤالات نیز این ابزار به نتایج خوبی دست پیدا کرده است. از این‌رو درنظر گرفتیم تا یک شبکه عصبی عمیق برای استخراج ویژگی و دسته‌بندی سؤالات طراحی کنیم.

برای بسیاری از روش‌های پردازش متن، نیاز به نمایش عددی کلمات موجود در متون داریم تا بتوانیم از انواع روش‌های عددی حوزه یادگیری ماشین مانند اکثر الگوریتم‌های دسته‌بندی روی لغات و اسناد استفاده کنیم. یکی از روش‌هایی که در این حوزه بسیار رایج شده است، نمایش برداری کلمات و جملات است .

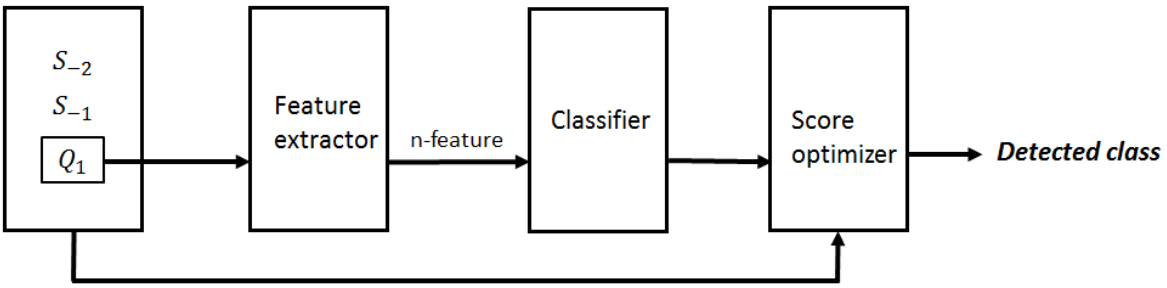
یکی از روش‌های کارآمد نمایش اسناد و پردازش آن‌ها، روش Word2Vec است. این روش بازنمایی متن در بسیاری از کاربردهای نوین پردازش متن مانند سنجش احساسات، جستجوی متون مشابه و یا پیشنهاد اخبار یا کالای مشابه استفاده شود.

در این روش به کمک شبکه عصبی یک بردار با اندازه کوچک و ثابت برای نمایش تمام لغات و متون در نظرگرفته شده و با اعداد مناسب در فاز آموزش، برای هر لغت این بردار محاسبه می‌شود. بعد از ایجاد بردارهای مرتبط با هر لغت، برای نمایش برداری هر متن، می‌توان بردار تک تک کلمات به کار رفته در آنرا یافته و میانگین اعداد هر ستون را به دست آورد که نتیجه آن یک بردار برای هر متن یا سند خواهد بود. سرعت این آموزش بسیار بالا است و در عرض چند ساعت و یا چند دقیقه (بسته به این‌که کدام یک از دو الگوریتم موجود برای آموزش شبکه را استفاده می‌کنیم) می‌توان حجم عظیمی از داده‌ها را به این الگوریتم داده و بردارهای لغات متناظر را ایجاد کرد.

## ۴-۴- ارائه مدل کلی

همان‌طور که در شکل ۳-۱ مشاهده می‌کنیم، مدل کلی ما شامل ۴ ماژول اصلی است. در بخش اول، پیش‌پردازش و آماده‌سازی روی داده‌ها انجام می‌گیرد. پس از آن یک سؤال برای دسته‌بندی به سیستم داده می‌شود. سؤال مطرح شده برای استخراج ویژگی به ماژول استخراج‌کننده ویژگی می‌رود. در این قسمت با استفاده از شبکه عصبی از پیش آموزش داده شده یک بردار ویژگی برای سؤال

موردنظر، بدست می‌آید. پس از آن این بردار ویژگی به ماژول طبقه‌بندی داده می‌شود. این ماژول با استفاده از یک شبکهٔ عصبی عمیق، طبقه‌بندی سؤالات را انجام می‌دهد. در این قسمت شبکهٔ مذکور، با توجه به سؤال ارائه شده برای هر کلاس تعریف شده در مجموعهٔ دادگان، یک امتیاز تعیین می‌کند. پس از آن بردار امتیاز بدست آمده به ماژول score optimizer رفته و امتیازات بدست آمده بررسی می‌شوند. در صورتی که امتیازها به صورت قابل قبولی تمییز دهندهٔ کلاس‌ها نباشند، نیاز به استفاده از جملات قبلی برای تعیین کلاس این سؤال وجود خواهد داشت. به این منظور به ماژول اول در مدل برگشت می‌کنیم و جمله قبل را برای تعیین امتیاز در همین پروسه انتخاب می‌کنیم. پس از تعیین امتیاز در صورتی که مجموع امتیازات سؤال و جمله قبلی مشخص کننده کلاس باشد، کلاس سؤال را به عنوان خروجی سیستم مشخص می‌کنیم؛ در غیر این صورت این پروسه تکرار می‌شود و از جملات قبلی سؤال مطرح شده تا جایی استفاده می‌شود که بتوان با توجه به حد آستانه مشخص شده برای تفاوت امتیازات، یک کلاس را به صورت قطعی تعیین کرد.



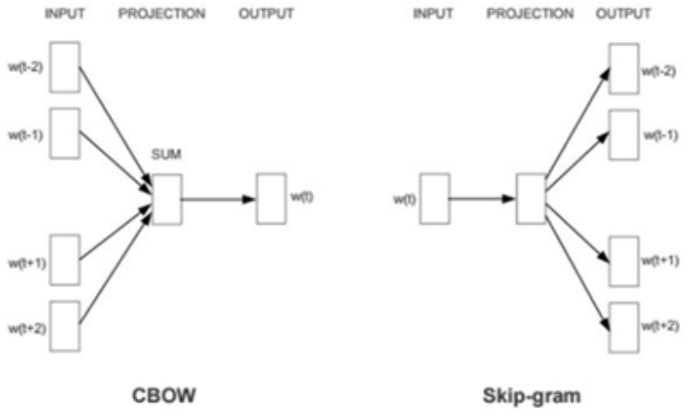

شکل (۴-۱) ساختار کلی روش پیشنهادی

## ۴-۴-۲- ماژول استخراج‌کننده ویژگی

همان‌طور که گفته شد، در این مدل برای استخراج ویژگی از ابزار word2vec استفاده می‌کنیم. این ابزار یک شبکه عصبی از قبل آموزش دیده شده است که در این بخش به صورت مختصر آن را شرح می‌دهیم.

word2vec یک مدل پیش‌بینی کننده به منظور یادگیری بازنمایی لغات از متن خام است که از لحاظ پیچیدگی محاسباتی بسیار کارا است. به بیان ساده در این مدل، روابط بین لغات از نحوه قرارگیری آن‌ها در متون استخراج می‌شود. همان‌طور که در فصل پیش به آن اشاره شد، word2vec به دو روش پیاده‌سازی می‌شود. یک روش استفاده از الگوریتم کیسهٔ لغات پیوسته[1] و دیگری الگوریتم اسکیپ-گرام[2] است.

از لحاظ الگوریتمی این دو روش شبیه هم هستند، با این تفاوت که CBOW لغات هدف را از روی لغات متن ورودی پیش‌بینی می‌کند ولی اسکیپ-گرام به صورت برعکس از روی لغات مرجوعه هدف، لغات ورودی را پیش‌بینی می‌کند. نحوهٔ کار این دو روش را می‌توانید در شکل ۳-۲ ببینید.




شکل (۴-۲) ساختار شبکه word2vec با الگوریتم‌های CBOW و Skip-gram

از لحاظ آماری CBOW، با رفتاری شبیه به یک مشاهده بر روی کل متن، تأثیر نرمی بر روی همه اطلاعات توزیعی دارد و درکل می‌تواند روشی مفید برای استفاده در مجموعه دادگان کوچک‌تر باشد. اما اسکیپ-گرام با هر زوج محتوا-هدف به‌صورت یک مشاهدهٔ جدید رفتار می‌کند و در مجموعه دادگان بزرگ‌تر، جواب مناسب‌تری می‌دهد. در حقیقت از این ترفند در استخراج بدون ناظر ویژگی‌ها

[1] Continuous Bag-Of-Words (CBOW)
[2] Skip-Gram

استفاده می‌شود جایی که شما یک اتوانکودر را آموزش می‌دهید تا یک بـردار ورودی را در لایـه‌هـای مخفی فشرده کند و مجدداً در لایه خروجی از فشرده‌سازی خارج نماید. پس از آموزش ایـن شـبکه ، بدون داشتن داده‌های برچسب خورده شما ویژگی‌های خوبی را استخراج کردید.

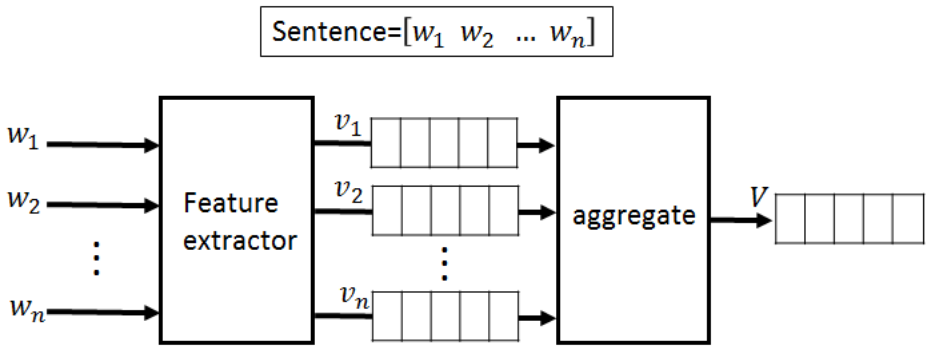
در جدول ۳-۱ نمونه‌ای از مقایسه این الگوریتم با دو روش مختلف برای استخراج ویژگی آمـده اسـت. زمان لازم برای ایجاد بردارها، دقت و ابعـاد بردارهـا در زیـر آورده شـده اسـت. در ایـن جـدول، روش Word2Vec با دو الگوریتم ایجاد بردار CBOW و Skip-gram با روش‌های مدل‌سازی زبان با شـبکه عصبی مقایسه و ارزیابی شده است.

جدول (۴-۱) مقایسه الگوریتم‌های بازنمایی word2vec با سایر روش‌های مدل‌سازی زبان

| مدل | ابعاد بردار | تعداد کلمات مجموعهٔ آموزش | زمان آموزش | دقت [%] |
|---|---|---|---|---|
| Collobert NNLM | ۵۰ | ۶۶۰ میلیون | ۲ ماه | ۱۱ |
| Turian NNLM | ۲۰۰ | ۳۷ میلیون | کمتر از یک هفته | ۲ |
| Mnih NNLM | ۱۰۰ | ۳۷ میلیون | ۷ روز | ۹ |
| Mikolov RNNLM | ۶۴۰ | ۳۲۰ میلیون | یک هفته | ۲۵ |
| Huang NNLM | ۵۰ | ۹۹۰ میلیون | یک هفته | ۱۳ |
| Our NNLM | ۱۰۰ | ۶ میلیارد | ۲,۵ روز | ۵۱ |
| Skip-gram | ۱۰۰۰ | ۶ میلیارد | چند ساعت | ۶۶ |
| CBOW (negative) | ۳۰۰ | ۱,۵ میلیارد | چند دقیقه | ۷۲ |

ما در مدل خود از یک شبکه عصبی آموزش داده شده با مجموعه لغات GoogleNews برای استخراج ویژگی استفاده کردیم. از آن‌جا که ورودی مورد نیاز ما در این شبکه جملات هستند، طبق شکل ۳-۴ کلمات مربوط به جمله مورد نظر اسپارس شده و به شبکه داده می‌شود. بردار ویژگی بدست آمده برای هر جمله، از جمع بردارهای ویژگی متعلق به کلمات آن جمله بدست مـی‌آیـد. از ایـن بـردار ویژگـی

می‌توان به عنوان ورودی برای طبقه‌بندها استفاده کرد.

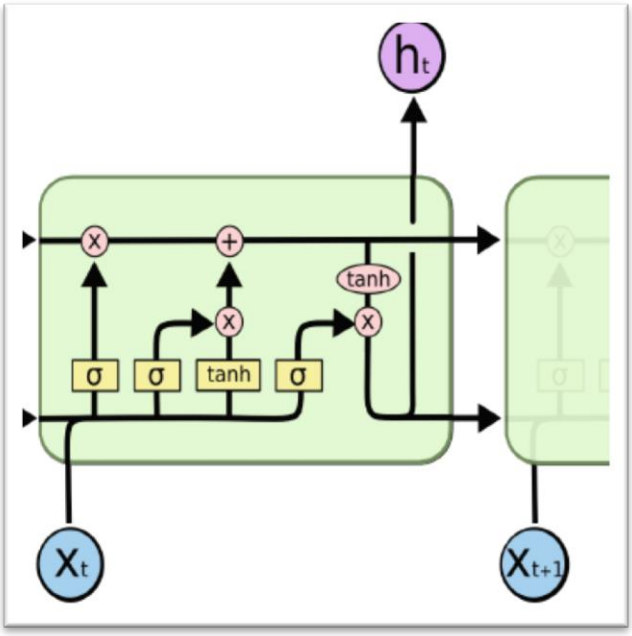



شکل (۴-۳) بدست آوردن بردار ویژگی برای جملات

### ۴-۴-۳- جزئیات شبکه عمیق

برای تشخیص برچسب هر جمله ورودی، ویژگی‌های استخراج شده در مرحله قبل را به یک رده‌بند می‌دهیم تا کلاس آن را برای ما تعیین کند. این رده‌بند می‌تواند از یک naïve Bayes ساده تا یک شبکهٔ کانولوشنال عمیق باشد. در اینجا به دلیل کارایی بالای شبکه‌های عمیق، تصمیم گرفتیم از آن‌ها در این بخش استفاده کنیم. شبکه‌های مختلفی مانند شبکه‌های کانولوشن، بازگشتی و حافظه بلند-کوتاه مدت را می‌توانیم برای این کار استفاده کنیم. از بین شبکه‌های عمیقی که کارایی بالایی دارند، شبکهٔ حافظه کوتاه-بلند[1] اخیراً در مباحث پردازش طبیعی زبان به خوبی استفاده شده است. شکل ۴-۳ شبکهٔ پایه LSTM که در فصل قبل معرفی شد را نشان می‌دهد.

[1] LSTM

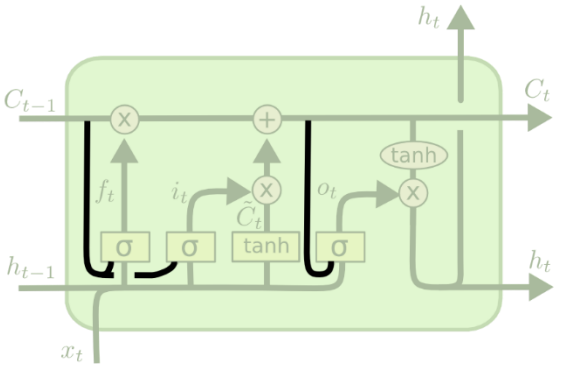



شکل (۴-۴) یک شبکه عادی LSTM [۵۰]

این شبکه، یکی از ساده‌ترین شکل‌های ممکن شبکه LSTM می‌باشد، اما عملاً در کاربردهای آکادمیک از نسخه‌های جدیدتر این شبکه استفاده می‌شود. یکی از راهکارهای جدیدی که در شبکه های LSTM استفاده می‌شود، استفاده از مفهوم روزنه[۱] در شبکه است.

این نوع شبکه در سال ۲۰۰۰ توسط Gers و Schmidhuber ارائه شده است که از اضافه کردن اتصال‌های روزنه به شبکهٔ پایه به وجود آمده است. به این معنی که لایه های دروازه[۲] می‌توانند به سلول‌های حالت[۳] دسترسی داشته باشند.

[11] Peephole
[2] Gate
[3] Cell State

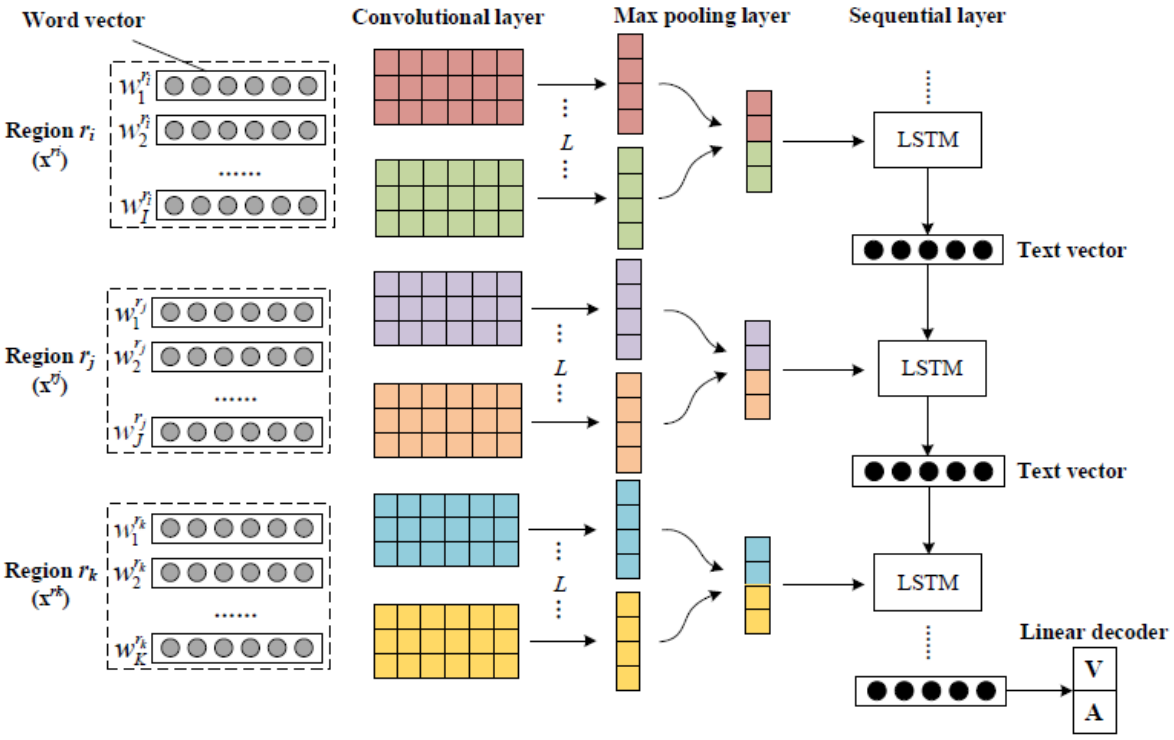



$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \; + \; b_o\right)$$

شکل (۴-۵) یک شبکه LSTM که روزنه ها ( به رنگ سیاه) به آن اضافه شده اند [۵۰].

نمونه‌ای از این شبکه که ما در این پژوهش از آن استفاده کرده‌ایم در شکل ۳-۶ دیده می‌شود. در این شبکه به همه دروازه‌ها، روزنه متصل شده است؛ هر چند در برخی مقالات، این اتصالات فقط به برخی از دروازه‌ها متصل می‌شود.

از سویی دیگر، علاوه بر استفاده از شبکهٔ LSTM، به تازگی از یک شبکه ترکیبی استفاده می‌شود که قسمت اصلی آن همان شبکهٔ LSTM معرفی شده است، با این تفاوت که در ابتدا و قبل از ورود ویژگی‌ها، از یک شبکه CNN برای پیش پردازش داده‌ها استفاده می‌شود [۵۱]. در این شبکه، یک فیلتر بر روی ویژگی‌های ورودی کلمات مختلف حرکت می‌کند. این کار باعث در نظر گرفتن توالی مکانی کلمه‌ها در جمله می‌شود که به نوعی یک ویژگی مشابه ویژگی N-Gram را پیاده می‌کند. شمای کلی این شبکه را در شکل ۳-۶ می‌بینیم. در این شبکه پس از اعمال فیلتر بر روی ورودی‌ها، با استفاده از یک لایهٔ max-pooling ، ویژگی‌های جدید با هم ترکیب شده و وارد شبکه LSTM می‌شوند.




شکل (۴-۶) شمای کلی شبکه ترکیبی cnn-lstm که در این پژوهش استفاده شده است [۵۱].

## ۴-۵- جمع‌بندی

در این فصل شبکه مورد نظر خود را هم به صورت کلی و هم به صورت جزء به جزء ارائه دادیم. در فصل بعد این روش را با استفاده از سناریوهای ارائه شده می‌آزماییم و نتایج را برای هر مرحله ارائه می‌دهیم.

# فصل ۵:

## نتایج و تحلیل آن‌ها

## ۵-۱- مقدمه

در فصل پیش به معرفی روش پیشنهادی خود پرداختیم، در این فصل نتایج عملی این روش را شرح می‌دهیم. در ابتدا مجموعه دادگان مورد استفاده را معرفی می‌کنیم و پس از آن به مرور جزئیات پیاده‌سازی می‌پردازیم. در ادامه نتایج رویکرد خود را برای سناریوهای مختلف با برخی از روش‌های پیشنهادی مهم در این حوزه، مقایسه می‌کنیم.

## ۵-۲- مجموعه دادگان مورد استفاده

در این پژوهش به دلیل ماهیت تعاملی سیستم، نیاز به یک مجموعه از مکالمات به عنوان مجموعه دادگان آموزشی وجود داشت. به این منظور یک مجموعه از مکالمات جمع آوری گشت که در مجموع شامل ۲۴۸۵ مکالمه با میانگین ۱۰ جمله است. این مجموعه جمع‌آوری شده با توجه به استاندارد استفاده شده در مجموعه دادگان مرجع TREC10-11 که شامل ۶ کلاس درشت‌دانه است، توسط چند عامل انسانی برچسب‌گذاری شده است. در جدول ۴-۱ این دسته‌ها را مشاهده می‌کنید.

جدول (۵-۱) دسته‌بندی استاندارد برای طبقه‌بندی سؤالات در مجموعه‌دادگان TREC

| کلاس درشت‌دانه | کلاس ریزدانه |
|---|---|
| ABBR | abbreviation, expression |
| DESCRIPTION | definition, description, manner, reason |
| ENTITY | animal, color, creation, currency, disease/medicine, event, food, instrument, language, letter, other, plant, product, religion, sport, sub-stance, symbol, technique, term, vehicle, word |
| HUMAN | Description, group, individual, title |
| LOCATION | City, country, mountain, other, state |
| NUMERIC | Code, count, date, distance, money, order, other, percent, period, speed, temperature, size, weight |

لازم به ذکر است که به دلیل ماهیت مکالمات، از کلاس ABBR در برچسب‌گذاری مجموعه دادگان

استفاده نشد. جدول ۴-۲ دسته‌های موجود در مکالمات را که برای این پروژه استفاده شده است، نشان می‌دهد.

جدول (۵-۲) کلاس‌های استفاده شده برای برچسب‌گذاری مجموعه دادگان

| کلاس‌ها |
| --- |
| DESCRIPTION |
| ENTITY |
| HUMAN |
| LOCATION |
| NUMERIC |

در فرآیند برچسب‌گذاری، جملات مکالمه را به سه دستهٔ sentence، wh-question و non-wh-question تقسیم کردیم. جملات با برچسب sentence جملات غیرپرسشی مکالمات هستند. جملات پرسشی با برچسب wh-question حاوی کلمات پرسشی whدار نظیر what، where و غیره بوده و جملات پرسشی با برچسب non-wh-question فاقد کلمات پرسشی whدار هستند. جملاتی که ما در مکالمه قصد دسته‌بندی آن‌ها را داریم، جملات پرسشی با برچسب wh-question است که آن‌ها را به ۵ دسته درشت دانه، دسته‌بندی می‌کنیم. در این پژوهش ۲/۳ از داده‌ها را برای مجموعهٔ آموزش و ۱/۳ باقی‌مانده را برای مجموعهٔ تست درنظر می‌گیریم و از این دو مجموعه برای آزمودن روش پیشنهادی و مقایسهٔ آن با سایر روش‌ها استفاده می‌کنیم. در شکل ۴-۱ دو نمونه از این مکالمات برچسب‌گذاری شده را مشاهده می‌کنید.

```
<conversation id="766" subject="Where to Buy a Pass">
<personA act="question" category="non-wh-question">Did you get your bus pass?</personA>
<personB act="sentence">I haven't gone to get it yet.</personB>
<personA act="question" category="wh-question" classname="DESCRIPTION:description">Why haven't you got it?</personA>
<personB act="sentence">I don't know where to go to get one.</personB>
<personA act="sentence">I know where to get them from.</personA>
<personB act="question" category="wh-question" classname="LOCATION:other">Where do I get it?</personB>
<personA act="sentence">They sell them at the business office on campus.</personA>
<personB act="sentence">I had no idea.</personB>
<personA act="sentence">I'm sure you didn't.</personA>
<personB act="sentence">Thank you.</personB>
<personA act="sentence">Don't mention it.</personA>
<personB act="sentence">I'll buy it later on today.</personB>
</conversation>
<conversation id="1132" subject="Where to Buy Lunch">
<personA act="question" category="non-wh-question">Let's have some lunch; do you know of a great restaurant somewhere?</personA>
<personB act="question" category="wh-question" classname="ENTITY:food">I'm hungry, too. What types of food sound good?</personB>
<personA act="sentence">Maybe just a quick sandwich, salad, or something.</personA>
<personB act="sentence">I have a great idea of where you might like to go.</personB>
<personA act="question" category="non-wh-question">Can we walk there?</personA>
<personB act="question" category="non-wh-question">It's on the second floor. May I join you?</personB>
<personA act="sentence">Of course! Let's go.</personA>
<personB act="sentence">Let's hurry. I'm starving!</personB>
</conversation>
```

شکل (۵-۱)  دو نمونه از مکالمات انتخاب شده از مجموعه دادگان

## ۵-۳- پیاده‌سازی

در این بخش به معرفی جزئیات پیاده‌سازی و پیش پردازش‌های انجام شده می‌پردازیم.

### ۵-۳-۱- پیش‌پردازش دادگان

مجموعه دادگان در فرمت xml طبق فرآیندی که توضیح داده شد، برچسب‌گذاری شده است. با استفاده از برچسب‌های زده شده روی داده‌ها، سؤالات حاوی کلمات پرسشی wh جدا شده و برای دسته‌بندی به سیستم داده می‌شوند. از مجموعه دادگان آموزش و تست، ایست واژه‌ها[1] با استفاده از یک تابع حذف می‌شوند. ایست واژه‌ها لغاتی هستند که تعداد رخداد بالا و بار معنایی کمی دارند بنابراین حذف این واژگان از پایگاه‌داده حجم محاسبات را کاهش و کارایی را بهبود می‌دهد.

همچنین به کمک یک تابع، علائم نگارشی و نیز فضای خالی اضافی موجود در جملات حذف می‌شوند. پس از آن سؤال به ماژول استخراج‌کننده ویژگی منتقل می‌گردد.

## ۵-۴- معرفی چارچوب TensorFlow

در این پژوهش برای پیاده‌سازی شبکه‌های عمیق پیشنهادی که در فصل قبل توضیح داده شد، از

[1] Stop Word

چارچوب TensorFlow استفاده کرده‌ایم. TensorFlow یکی از سریع‌ترین چارچوب‌ها برای توسعه برنامه‌هایی با وظایف مفهومی و مرتبط با پردازش زبان، در حوزه یادگیری عمیق است. تنسورفلو در آغاز توسط تیم برین در مرکز تحقیقاتی گوگل، به صورت داخلی استفاده می‌شد اما بعدها، در ۹ نوامبر ۲۰۱۵ تحت مجوز آپاچی منتشر شد.

## ۵-۵- آزمون‌ها و پارامترهای مورد نظر

در ابتدا برای مقایسهٔ روش پیشنهادی خود با سایر روش‌های متداول در این زمینه، تعدادی از روش‌های آماری طبقه‌بندی سؤال را که در فصل دوم بررسی شدند، پیاده‌سازی و پرسش‌های موجود در پایگاه دادهٔ خود را با استفاده از آن‌ها دسته‌بندی کردیم. همان‌طور که گفته شد، تمرکز ما برای انجام کار و همچنین مقایسه‌ها در انجام این پژوهش، بر روی روش‌های آماری بوده است. دلیل انتخاب این روش‌ها مزایایی است  که پیش‌تر مورد اشاره قرار گرفت.

بهترین ویژگی‌های آماری استخراج شده برای مجموعه دادگان مورد استفاده، سه ویژگی TFIDF، N-گرم و کیسه‌ای از کلمات است. برای دسته‌بندی به کمک این ویژگی‌ها، از تعدادی از الگوریتم‌های طبقه بندی نظیر ماشین بردار پشتیبان، بیزین ساده و درخت تصمیم استفاده کردیم. از میان روش‌های طبقه‌بندی استفاده شده، الگوریتم ماشین بردار پشتیبان، بهترین نتیجه را روی مجموعه دادگان مورد استفادهٔ ما در این پژوهش داشته است؛ در نتیجه برای مقایسهٔ روش پیشنهادی این الگوریتم را برگزیده‌ایم. بنابراین ویژگی‌ها و روش دسته‌بندی انتخاب شده برای مقایسه عبارتند از:

- ویژگی TFIDF با رده‌بند SVM
- ویژگی N-گرم (N-Gram) با رده‌بند SVM
- ویژگی کیسه کلمات (BOW) با رده بند SVM

دو مکالمه را به عنوان نمونه از مجموعه دادگان درنظر می‌گیریم و امتیازات بدست آمده برای یک سؤال انتخاب شده از هر کدام را به کمک ویژگی‌های آماری ذکر شده، بررسی می‌کنیم. به این منظور

امتیازات جملهٔ اول از مکالمهٔ شماره ۱۰۱۸ و جملهٔ هشتم از مکالمهٔ شماره ۹ را استخراج می‌کنیم.

سؤالات انتخاب شده از این مکالمات را در جدول ۴-۲ می‌بینید. همچنین امتیاز بدست آمده به وسیلهٔ طبقه‌بند svm را برای این سؤالات، به ترتیب در جدول ۴-۳ و ۴-۴ مشاهده می‌کنید.

```
<conversation id="1018" subject="Ordering Food at the Cafeteria 2">
<personA act="question" category="wh-question" classname="ENTITY:food">What did you get for lunch today?</personA>
<personB act="sentence">All I had was a sandwich, chips, and soda.</personB>
<personA act="question" category="non-wh-question">Where'd you get your food from?</personA>
<personB act="sentence">I went to the cafeteria and bought it.</personB>
<personA act="question" category="wh-question" classname="ENTITY:food">What sandwich did you order?</personA>
<personB act="sentence">I ordered a ham sandwich, but they gave me a bologna sandwich instead.</personB>
<personA act="question" category="non-wh-question">Was it any good?</personA>
<personB act="sentence">I enjoyed it, even though I had not asked for it.</personB>
<personA act="sentence">I ordered a sandwich there before.</personA>
<personB act="question" category="non-wh-question">Is that right?</personB>
<personA act="sentence">Yes, and they messed my order up too.</personA>
<personB act="sentence">That may be true, but I'm sure you enjoyed your sandwich.</personB>
</conversation>
```

شکل (۵-۲) مکالمه شمارهٔ ۱۰۱۸، سؤال انتخاب شده از دستهٔ ENTITY

```
<conversation id="9" subject="Pre-Qualifying for a Loan">
<personA act="sentence">I want to go house shopping, and I want to know what I can afford to buy.</personA>
<personB act="question" category="wh-question" classname="NUMERIC:money">There is a formula that can help us
 figure that out.What is your annual salary?</personB>
<personA act="sentence">I earn sixty-five thousand dollars per year.</personA>
<personB act="question" category="wh-question" classname="NUMERIC:time">What is the length of time that you
have worked at the job you have now?</personB>
<personA act="sentence">I have been at the job I now have for 2 years, and I have been with the company for 6
years total.</personA>
<personB act="question" category="non-wh-question">Do you receive any other income?</personB>
<personA act="sentence">I own many unit a partner on sixty-five percent.</personA>
<personB act="question" category="wh-question" classname="NUMERIC:money">Do you know what your credit score
is?</personB>
<personA act="sentence">I dont know, but the last time I checked, it was very good!</personA>
<personB act="sentence">You are in good shape. Figuring in your other expenses, you can easily afford a house
costing seven hundred thousand or maybe even a little more!</personB>
</conversation>
```

شکل (۵-۳) مکالمهٔ شمارهٔ ۹، سؤال انتخاب شده از دستهٔ NUMERIC

جدول (۵-۳) سؤال های مورد بررسی از مکالمه‌های انتخاب شده

| | |
|---|---|
| “What did you get for lunch today?” | $Q_1$ |
| “Do you know what your credit score is?” | Q2 |

جدول (۵-۴) امتیازات متعلق به هر کلاس برای سوال $Q_1$ توسط رده‌بند SVM

| کلاس | ویژگی BOW | ویژگی N-Gram | ویژگی TFIDF |
|---|---|---|---|
| NUMERIC | ۰,۰۵۰۵۸ | ۰,۰۰۰۶۶ | ۰,۰۷۴۳۳ |
| LOCATION | ۰,۰۰۸۵۰ | ۰,۰۰۰۸۸ | ۰,۰۳۲۳۶ |
| DESCRIPTION | ۰,۵۵۱۲۸ | ۰,۵۵۴۹۵ | ۰,۶۶۷۷۹ |
| ENTITY | ۰,۳۷۱۲۵ | ۰,۴۴۱۰۴ | ۰,۲۲۲۹۰ |
| HUMAN | ۰,۰۱۸۳۷ | ۰,۰۰۲۴۴ | ۰,۰۰۲۶۰ |

جدول (۵-۵) امتیازات متعلق به هر کلاس برای سوال $Q_2$ توسط رده‌بند SVM

| کلاس | ویژگی BOW | ویژگی N-Gram | ویژگی TFIDF |
|---|---|---|---|
| NUMERIC | ۰,۰۵۴۱۳ | ۰,۰۰۱۹۴ | ۰,۱۵۱۹۳ |
| LOCATION | ۰,۰۱۵۱۱ | ۰,۰۱۸۲۲ | ۰,۰۲۸۲۵ |
| DESCRIPTION | ۰,۸۱۲۶۴ | ۰,۱۱۲۶۱ | ۰,۶۵۵۳۲ |
| ENTITY | ۰,۱۱۴۸۸ | ۰,۸۶۰۴۹ | ۰,۱۲۸۴۴ |
| HUMAN | ۰,۰۰۳۲۲ | ۰,۰۰۶۷۲ | ۰,۰۳۶۰۴ |

طبق آنچه که در فصل پیش توضیح داده شد، در این پژوهش، روش پیشنهادی ما برای عملیات دسته‌بندی سؤال از استخراج ویژگی به روش Word2Vec، به همراه روش‌های یادگیری عمیق استفاده می‌کند. پس از بررسی ساختارهای متداول موجود در پردازش زبان‌های طبیعی، به ویژه دسته‌بندی متون و جملات، ما به طور خاص از شبکهٔ ترکیبی کانولوشن و LSTM به عنوان روش یادگیری عمیق استفاده کردیم. برای مقایسه، نتایج هر کدام از این دو شبکه را نیز به طور جداگانه بررسی می‌کنیم. پس از آن، میزان بهبود دقت را با روش استفاده از دنبالهٔ مکالمات که در فصل پیش، در بخش مدل کلی سیستم شرح دادیم، می‌سنجیم. بنابراین روش‌های مورد بررسی به کمک یادگیری عمیق در این پژوهش به شرح زیر است:

- ویژگی W2Vec به همراه شبکه عصبی CNN

- ویژگی W2Vec به همراه شبکه عصبی LSTM
- ویژگی W2Vec به همراه شبکه عصبی CNN-LSTM
- استفاده از دنباله مکالمه برای بهبود نتیجه به همراه CNN-LSTM

هر یک از آزمایشات با ۵ بار تقسیم مجموعه دادگان به دو قسمت یادگیری و آموزش انجام شده است و در پایان بازه اطمینان نتایج نیز ارائه شده است. به عنوان نمونه، امتیازات مربوط به دو سؤال از مکالمه‌های انتخاب شده برای مقایسهٔ روش‌های آماری را برای روش‌های گفته شده در یادگیری عمیق نیز بررسی می‌کنیم.

جدول (۵-۶) امتیازات متعلق به هر کلاس برای سوال $Q_1$ با استفاده از ویژگی word2vec

| کلاس | شبکهٔ CNN | شبکهٔ LSTM | شبکهٔ CNN-LSTM |
|---|---|---|---|
| NUMERIC | ۷,۴۲۶۰۴ | ۰,۱۹۰۲۷ | -۳,۷۵۶۹۹ |
| LOCATION | -۷,۷۱۷۷۶ | -۱۲,۲۹۶۵۳ | ۰,۰۶۴۹۶ |
| DESCRIPTION | -۵,۲۷۳۵۵ | ۰,۹۰۵۵۳ | ۱,۹۱۱۰۳ |
| ENTITY | -۵,۰۸۳۸۲ | ۱۵,۵۴۲۹۸ | ۱۸,۲۲۸۶۱ |
| HUMAN | -۴,۱۰۵۹۲ | -۷,۴۲۶۰۴ | ۲,۷۹۶۹۴- |

جدول (۵-۷) امتیازات متعلق به هر کلاس برای سوال $Q_2$ با استفاده از ویژگی word2vec

| کلاس | شبکهٔ CNN | شبکهٔ LSTM | شبکهٔ CNN-LSTM |
|---|---|---|---|
| NUMERIC | -۲,۸۱۸۱۲ | ۱۵,۱۸۶۳۴ | -۲,۳۱۴۵۷ |
| LOCATION | -۳,۰۰۸۴۰ | -۸,۶۲۹۹۵ | ۲,۹۰۴۵۰ |
| DESCRIPTION | ۱,۹۲۴۸۶ | ۱,۲۶۵۶۳ | -۱,۵۶۱۲۷ |
| ENTITY | -۱,۶۸۰۸۷ | ۱۱,۵۹۷۱۵ | -۳,۷۹۴۹۲ |
| HUMAN | -۵,۹۰۵۷۰ | -۸,۹۰۳۵۵ | ۱,۹۲۴۱۶ |

طبق نتایج جدول ۴-۵، بالاترین امتیاز بدست آمده توسط دو شبکهٔ LSTM و CNN-LSTM مطابق

با برچسب سؤال $Q_1$ است و این دو شبکه به درستی کلاس این سؤال را تشخیص داده‌اند. با توجه به امتیازهای بدست آمده برای سؤال دوم در جدول ۴-۶، کلاس تشخیص داده شده با بهترین شبکهٔ انتخاب شده یعنی CNN-LSTM نادرست است. همان‌طور که در بخش روش پیشنهادی توضیح داده شد، در صورتی که بازهٔ امتیازات بدست آمده برای انتخاب کلاس از یک حد آستانه، که با توجه به نوع شبکه تعیین می‌شود، کمتر باشد، برای کاهش خطا از جملات قبلی موجود در مکالمه استفاده می‌کنیم. در این حالت میانگین امتیاز استخراج شده برای پرسش مورد بررسی و جملهٔ قبل از آن را به عنوان امتیاز نهایی در نظر می‌گیریم. استفاده از جملات موجود قبلی در مکالمه تا جایی انجام می‌شود که اختلاف امتیازهای بدست آمده برای کلاس‌ها از میزان آستانهٔ تعیین شده، بیشتر شود. به این منظور با تغییر دادن ساختار شبکهٔ CNN-LSTM مجموعه دادگان تست را دوباره آزمایش کردیم. به کمک روش گفته شده، دقت دسته‌بندی سؤالات با ویژگی word2vec در شبکهٔ CNN-LSTM از ۸۰,۲ به ۸۵,۳ افزایش پیدا کرد. به عنوان نمونه، امتیاز سؤال مطرح شدهٔ $Q_2$ را مجدداً در دو حالت ساختار ساده و ساختار تعاملی سیستم در جدول ۴-۷ بررسی می‌کنیم.

جدول (۵-۸) امتیازات بدست آمده برای سوال $Q_2$ در دو شبکهٔ CNN-LSTM با ساختار ساده و تعاملی

| کلاس | شبکهٔ CNN-LSTM | شبکهٔ CNN-LSTM تعاملی |
|---|---|---|
| NUMERIC | -۲,۳۱۴۵۷ | ۱,۲۱۲۳۶ |
| LOCATION | ۲,۹۰۴۵۰ | -۰,۵۱۴۷۸ |
| DESCRIPTION | -۱,۵۶۱۲۷ | -۰,۸۱۶۶۵ |
| ENTITY | -۳,۷۹۴۹۲ | -۶,۰۲۷۷ |

## ۵-۶- معیار ارزیابی

برای ارزیابی روش پیشنهاد شده و مقایسه با سایر روش‌ها، از معیار-F استفاده می‌شود. معیارهای صحت[۱]، حساسیت[۲]، معیار-F[۳] و دقت[۴] از مهم‌ترین معیارهای مورد استفاده برای ارزیابی هستند. از بین معیارهای ذکر شده، معیار-F پارامتر مناسبی برای ارزیابی کیفیت کلاس‌بندی است و همچنین توصیف میانگین وزن‌دار بین دو کمیت صحت و حساسیت است. برای یک الگوریتم طبقه‌بند در شرایط ایده‌آل، مقدار این کمیت برابر با ۱ و در بدترین وضعیت برابر با صفر است. مقدار معیار-F از طریق رابطهٔ ۴-۱ محاسبه می‌شود.

$$f-measure = 2 \times \frac{percision \times recall}{percision + recall} \qquad (۱-۴)$$

در این رابطه، مقادیر precision و recall به ترتیب طبق روابط ۴-۲ و ۴-۳ بدست می‌آید.

$$percision = \frac{t_p}{t_p + f_p} \qquad (۲-۴)$$

$$recall = \frac{t_p}{t_p + f_n} \qquad (۳-۴)$$

که در این روابط، tp تعداد پاسخ مثبت درست، tn تعداد پاسخ منفی درست، fp تعداد پاسخ مثبت غلط و fn تعداد پاسخ منفی غلط است.

[1] Precision
[2] Recall
[3] F-measure
[4] Accuracy

## ۵-۷- نتایج

### ۵-۷-۱- نتایج رده بندی

در جدول ۴-۸ نتایج ابتدایی رده‌بندی سؤالات را به کمک روش‌های آماری مورد مقایسه آورده‌ایم.

جدول (۵-۹) نتایج استفاده از ویژگی‌های آماری بروی مجموعه دادگان

| روش | دقت (بازه اطمینان) |
|---|---|
| TFIDF + SVM | ۷۷,۲ (۷۷,۶ و ۷۶,۹) |
| N-Gram + SVM | ۷۸,۴ (۷۸,۸ و ۷۸,۱) |
| BOW + SVM | ۷۷,۶ (۷۸,۱ و ۷۷,۲) |
| TFIDF + N-Gram + BOW + SVM | ۷۸,۵ ( ۷۸,۹ و ۷۸,۲) |

نتایج شبکه‌های یادگیری عمیق استفاده شده، به کمک بـردار ویژگـی word2vec را در جـدول ۴-۹ می‌بینیم.

جدول (۵-۱۰) نتایج استفاده از شبکهٔ عصبی عمیق بروی مجموعه دادگان

| روش | دقت (بازه اطمینان) |
|---|---|
| Word2vec + CNN | ۷۹,۷ (۷۹,۴ و ۸۰,۱) |
| Word2vec + LSTM | ۸۰,۴ (۸۰,۷ و ۸۰,۲) |
| Word2vec + CNN-LSTM | ۸۲,۲ (۸۲,۴ و ۸۲,۰۱) |

### ۵-۷-۲- نتایج بهبود با استفاده از دنبالهٔ مکالمه

پس از مشاهدهٔ مطلوب بودن نتایج روش انتخاب شده، یعنی شبکهٔ CNN-LSTM بـه همـراه ویژگـی W2Vec، در این بخش، نتایج روش پیشنهادی برای دسته‌بندی سؤالات با استفاده از دنبالهٔ مکالمـات را که در فصل پیش ارائه شد، می‌آوریم. در این روش، اگر اختلاف امتیاز کلاس‌های موجود برای یـک نمونه از مقدار آستانهٔ تعیین شده کمتر باشد، به این معنی است که امکان دارد، رده‌بنـد در تشـخیص کلاس آن نمونه اشتباه کرده باشد. بنابراین در این حالت، از جمله قبلی سؤال مورد بررسی که احتمالاً به آن مرتبط است، کمک می‌گیریم. به این منظور برای هر کلاس، امتیاز جملهٔ قبل را با امتیـاز خـود نمونه میانگین و نتیجه را به عنوان نتیجهٔ نهایی در نظر می‌گیریم. این روش را روی سـاختار انتخـاب شده، یعنی شبکهٔ CNN-LSTM می‌آزماییم که نتیجهٔ آن در جدول ۴-۱۰ آورده شده است.

جدول (۵-۱۱) مقایسه دقت روش CNN-LSTM معمولی و تعاملی

| روش | دقت (بازه اطمینان) |
|---|---|
| CNN-LSTM | ۸۲,۲ (۸۲,۰ , ۸۲,۴) |
| CNN-LSTM + Conv. | ۸۵,۳ (۸۵,۱ , ۸۵,۵) |

همان‌طور که در جدول ۴-۱۰ مشاهده می‌شود، در صورت نامطمئن بودن بازهٔ تصمیم‌گیری، اسـتفاده از جملات قبلی موجود در مکالمه، میزان دقت سیستم را در دسـته‌بنـدی سـؤالات بـه صـورت قابـل توجهی بالا می‌برد.

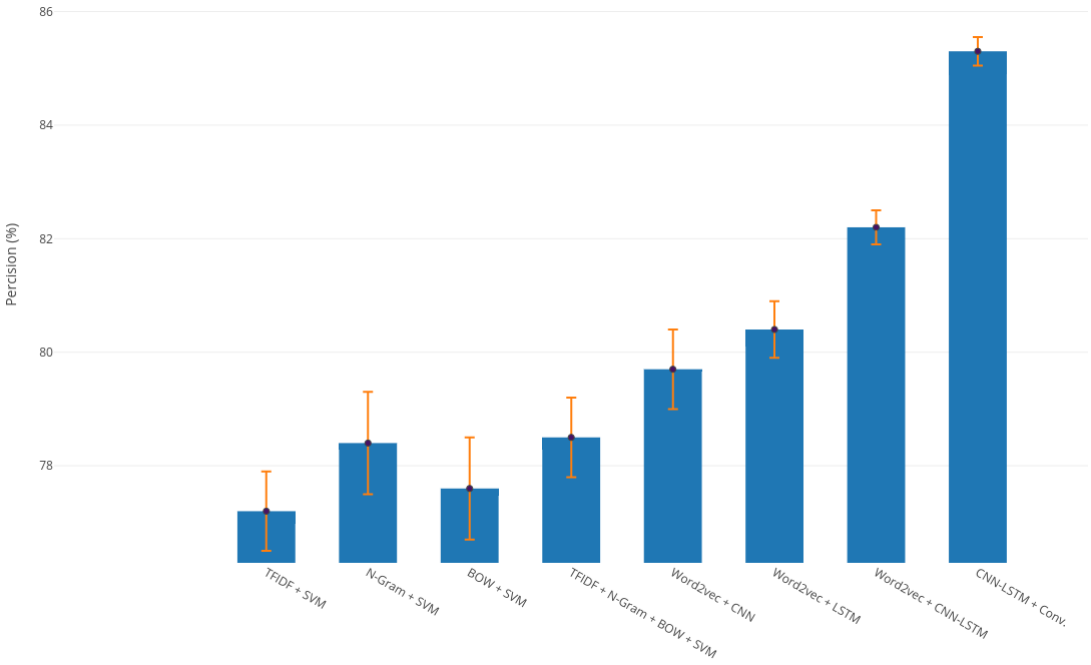
همچنین، در شکل ۴-۴، نمودار رشد دقت را به ازای تعداد قدم‌های تکرار، روی مجموعهٔ اعتبارسنجی

مشاهده می‌کنیم که در طول فرآیند یادگیری به ازای هر بستهٔ ورودی[1] به شبکه، به دست آمده است. همان‌طور که مشاهده می‌شود، دقت پس از شروع از یک مقدار پـایین، در پایـان آمـوزش، بـه مقـدار تقریباً ثابتی می‌رسد.

شکل (۵-۴) نمودار رشد دقت به ازای تعداد قدم‌های تکرار، روی مجموعه اعتبارسنجی

در پایان، تمامی روش‌های گفته شده را در قالب یک نمودار می‌آوریم. در شکل ۵-۴ دقت هر روش، به همراه بازهٔ اطمینان آن، مشاهده می‌شود.

[1] batch




شکل (۵-۵)  نمودار دقت و بازهٔ اطمینان روش‌های ارائه شده و سایر روش‌ها

## ۵-۷-۳- جمع‌بندی

در این فصل به بررسی نحوهٔ پیاده‌سازی و نتایج روش پیشنهادی پرداختیم. دیدیم که از لحاظ دقت، روش‌های یادگیری عمیق به طور معناداری نسبت به روش‌های دیگر برتری دارند. نتایج استفاده از روش دنبالهٔ مکالمات نیز برای داده‌هایی که سیستم در تشخیص کلاس آن‌ها دچار مشکل است، به طور امید بخشی دقت نتایج را بالا می‌برد. این روش قابلیت توسعهٔ زیادی دارد که می‌تواند در کارهای آینده مورد استفاده قرار گیرد.

# فصل ۶:

## نتیجه‌گیری

## ۶-۱- نتیجه‌گیری

تمرکز ما در این پژوهش، بر روی سیستم تشخیص نوع سؤال بود که در سیستم‌های خودکار پرسش و پاسخ استفاده می‌شود. در فصل دوم، به پیشینهٔ تحقیق پرداختیم. ابتدا روش‌هایی که جملات را مدل می‌کنند مانند N-گرم، کیسهٔ کلمات، مدل فضای برداری و غیره را بررسی کردیم. پس از بررسی کامل این بخش، به استفاده از این مدل‌ها برای طبقه‌بندی جملات و سوألات پرداختیم.

دیدیم که دو راهبرد کلی مبتنی بر قوانین و مبتنی بر یادگیری، برای دسته‌بندی جملات وجود دارد. در بخش مبتنی بر یادگیری، دیدیم که استفاده از شبکه‌های عمیق، به دلیل کارایی آن‌ها در همهٔ زمینه‌ها و به خصوص پردازش زبان‌های طبیعی گسترش زیادی داشته است.

ابتدا به طور اجمالی، یادگیری عمیق را بررسی نمودیم و مفاهیم مهم آن مانند یادگیری با ناظر در این شبکه‌ها، پس‌انتشار و غیره را شرح دادیم. پس از آن کارهای انجام شده با استفاده از یادگیری عمیق، در طبقه‌بندی متون و جملات را به طور خلاصه بررسی کردیم.

در فصل سوم راهبردی ارائه دادیم که در آن برای استخراج ویژگی‌ها و همچنین طبقه‌بندی سؤالات از این ویژگی‌ها، از شبکه عمیق استفاده می‌کند. برای استخراج مدل و ویژگی، از چارچوب word2vec استفاده کردیم؛ که هر کلمه را به برداری از ویژگی ها نگاشت می‌کند. سپس برای طبقه‌بندی جملات با استفاده از این ویژگی‌ها، از یک شبکه CNN-LSTM استفاده کردیم. در قسمت بعد، از توالی مکالمات برای دسته‌بندی استفاده کردیم؛ به طوری که جملات قبل از هر پرسش به تطبیق دقیق‌تر آن کمک کرده و برای تعیین کلاس پرسش موردنظر بررسی و استفاده می‌شود.

در فصل چهارم، به بررسی کارایی این راهبرد پرداختیم. مجموعه دادگان جمع‌آوری شده خود را معرفی نمودیم و پیاده‌سازی و استفاده عملی از سیستم خود را شرح دادیم. با مشاهدهٔ نتایج، دیدیم که استفاده از روش پیشنهادی، دقت طبقه‌بندی سؤالات را به طور معناداری افزایش می‌دهد. در ادامه به نتایجی که از این پژوهش قابل دستیابی است، اشاره می‌گردد.

ابتدا اهمیت توسعهٔ این سیستم‌ها می‌باشد. در دنیای فناوری امروز که هوش مصنوعی بیش از پیش به همهٔ کاربردهای زندگی نفوذ کرده است، استفاده از چت-بات‌ها و سیستم‌های پرسش و پاسخ گسترش فراوانی پیدا کرده است. در این زمینه، استفاده از سیستم‌های طبقه بندی پرسش و پاسخ، کارایی این ابزارها را به خوبی ارتقاء می‌دهد.

نکتهٔ بعد مربوط به همه‌گیری استفاده از شبکه‌های عمیق است. این شبکه‌ها توانسته‌اند تأثیر شگرفی در ارتقاء سیستم‌های کاربردی داشته باشند. در این پژوهش ما نیز سعی کرده‌ایم با استفاده از این چارچوب، به نتیجهٔ بهتری دست پیدا کنیم.

یکی دیگر از نکات این پژوهش، تولید یک مجموعه دادگان جدید، برای استفاده در زمینهٔ تحقیقات مشابه است. با بررسی‌هایی که انجام شد، مجموعه دادگان موجود فعلی تا حدود زیادی برای پروژه نامناسب بودند، از این رو مجموعه دادگان جدیدی ارائه دادیم که علاوه بر پژوهش جاری، می‌تواند در زمینه‌های مشابه، به کار گرفته شود. البته باید به این نکته اشاره کرد که در صورت دسترسی به مجموعه دادگان یک سیستم پرسش و پاسخ تعاملی که در آن تمرکز مکالمه روی یک موضوع مشخص باشد، به طور قطعی نتایج بدست آمده بهبود بیشتری خواهند داشت.

## ۶-۲- کارهای آینده

پژوهش‌های بیشتر در ادامه پروژه، می‌تواند برروی جنبه‌های مختلفی صورت گیرد که در زیر تعدادی از آن‌ها را پیشنهاد داده‌ایم:

- استفاده از روش‌های دیگر (غیر از چارچوب word2vec ) برای استخراج ویژگی. به طور مثال استفاده از چارچوب GLOVE برای استخراج بردار ویژگی.
- استفاده از شبکه‌های مختلف و پیچیده‌تر برای وظیفهٔ طبقه‌بندی. به طور مثال استفاده از شبکه‌های هیبرید عمیق.
- استفاده از روش‌های یادگیری ماشین، برای گام آخر مدل پیشنهاد شده. به عنوان مثال از یک

رده‌بند مرسوم، برای تطبیق با استفاده از امتیازات بدست آمده، استفاده کنیم.

- استفاده از شبکه‌های عمیق برای طبقه‌بندی، بر اساس ترکیبات سؤال و جمله‌های قبل. به این معنی که، به جای استفاده از ماژول بهبود دهندهٔ دقت بر اساس امتیاز، داده‌های خام را به طور مستقیم به خود شبکه بدهیم.

# مراجع

# مراجع


[1] Gupta, Poonam, and Vishal Gupta. "A survey of text question answering techniques." *International Journal of Computer Applications* 53, no. 4 (2012).

[2] Mishra, Amit, and Sanjay Kumar Jain. "A survey on question answering systems with classification." *Journal of King Saud University-Computer and Information Sciences* 28, no. 3 (2016): 345-361

[3] Ferret, Olivier, Brigitte Grau, Martine Hurault-Plantet, Gabriel Illouz, Laura Monceaux, Isabelle Robba, and Anne Vilnat. "Finding An Answer Based on the Recognition of the Question Focus." In *TREC*. 2001.

[4] Kolomiyets, Oleksandr, and Marie-Francine Moens. "A survey on question answering technology from an information retrieval perspective." *Information Sciences* 181, no. 24 (2011): 5412-5434.

[5] Dwivedi, Sanjay K., and Vaishali Singh."Research and reviews in question answering system." *Procedia Technology*10 (2013): 417-424.

[6] Katz, Boris, Sue Felshin, Deniz Yuret, Ali Ibrahim, Jimmy Lin, Gregory Marton, Alton Jerome McFarland, and Baris Temelkuran. "Omnibase: Uniform access to heterogeneous data for question answering." *Natural Language Processing and Information Systems* (2002): 230-234.

[7] Jones, Lyle V. "Beyond Babbage." *Psychometrika* 28, no. 4 (1963): 315-331.

[8] Woods, William A. "Progress in natural language understanding: an application to lunar geology." In *Proceedings of the June 4-8, 1973, national computer conference and exposition*, pp. 441-450. ACM, 1973.

[9] Voorhees, Ellen M., and L. Buckland. "Overview of the TREC 2003 Question Answering Track." In *TREC*, vol. 2003, pp. 54-68. 2003.

[10] Soricut, Radu, and Eric Brill. "Automatic question answering using the web: Beyond the factoid." *Information Retrieval* 9, no. 2 (2006): 191-206.

[11] Li, Xin, and Dan Roth. "Learning question classifiers." In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pp. 1-7.

Association for Computational Linguistics, 2002.

[12] Miller, George A. "WordNet: a lexical database for English." *Communications of the ACM* 38, no. 11 (1995): 39-41.

[13] Carbonell, Jaime, Donna Harman, Eduard Hovy, Steve Maiorano, John Prange, and Karen Sparck-Jones. "Vision statement to guide research in Question & Answering (Q&A) and Text Summarization." *Final version* 1 (2000).

[14] Unger, Christina, Lorenz Bühmann, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, Daniel Gerber, and Philipp Cimiano. "Template-based question answering over RDF data." In *Proceedings of the 21st international conference on World Wide Web*, pp. 639-648. ACM, 2012.

[15] Feng, M., Liu, W.Y., Xie, N.F., Chen, W.: Answer clustering and fusion in a user-interactive QA system.Proc. 2nd International Conference on Semantics, Knowledge and Grid (SKG2006), November 1–3, 2006, Guilin, China

[16] Google Answers.http://answers.google.com/

[17] Stroh, Eylon, and Priyank Mathur. "Question Answering Using Deep Learning".

[18] Weston, Jason, Antoine Bordes, Sumit Chopra, Alexander M. Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. "Towards ai-complete question answering: A set of prerequisite toy tasks." arXiv preprint arXiv:1502.05698(2015.(

[19] Loni, Babak. "A survey of state-of-the-art methods on question classification." (2011).

[20] Mishra M., Kumar V., Sharma H.R., (2013), "Question Classification using Semantic, Syntactic and Lexical features," International Journal of Web & Semantic Technology (IJWesT), 4(3), pp. 39-47.

[21] Laokulrat N., (2013), "A survey on question classification techniques for question
answering," KMITL Information Technology Journal, 2(1).

[22] Lashkari A.H., Mahdavi F., Ghomi V., (2009), "A Boolean Model in

Information
Retrieval for Search Engines," Proceedings International Conference on Information Management and Engineering (ICIME '09), pp. 385-389, Malaysia.

[23] Brill E., Dumais S., Banko M., (2002), "An Analysis of the AskMSR Question
Answering System," Proceedings conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 257-264, Philadelphia.

[24] Haris, Syahidah Sufi, and Nazlia Omar. "A rule-based approach in Bloom's Taxonomy question classification through natural language processing." In *Computing and Convergence Technology (ICCCT), 2012 7th International Conference on*, pp. 410-414. IEEE, 2012.

[25] Huang, Zhiheng, Marcus Thint, and Zengchang Qin. "Question classification using head words and their hypernyms." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 927-936. Association for Computational Linguistics, 2008.

[26] Silva, Joao, Luísa Coheur, Ana Cristina Mendes, and Andreas Wichert. "From symbolic to sub-symbolic information in question classification." *Artificial Intelligence Review* 35, no. 2 (2011): 137-154.

[27] Metzler, Donald, and W. Bruce Croft. "Analysis of statistical question classification for fact-based questions." *Information Retrieval* 8, no. 3 (2005): 481-504.

[28] Roth, Dan, and Kevin Small. "The role of semantic information in learning question classifiers." In *Proceedings of the Conference First International Joint Conference on Natural Language Processing*. 2004.

[29] Nguyen, Tri Thanh, Le Minh Nguyen, and Akira Shimazu. "Improving the accuracy of question classification with machine learning." In *Research, Innovation and Vision for the Future, 2007 IEEE International Conference on*, pp. 234-241. IEEE, 2007.

[30] Bullington, Jim, Ira Endres, and M. Rahman. "Open ended question classification using support vector machines." *MAICS 2007* (2007).

[31] Zhang, Dell, and Wee Sun Lee. "Question classification using support vector

machines." In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 26-32. ACM, 2003.

[32] Pan, Yan, Yong Tang, Luxin Lin, and Yemin Luo. "Question classification with semantic tree kernel." In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 837-838. ACM, 2008.

[33] Tomas, David, and Claudio Giuliano. "A semi-supervised approach to question classification." In *ESANN*. 2009.

[34] Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. "Indexing by latent semantic analysis." Journal of the American society for information science 41, no. 6 (1990): 391.

[35] Kocik, Krystle. "Question classification using maximum entropy models." *201 1-1 1-23]. http://www. it. usyd. edu. au/research/news/koeik—summary. pdf* (2004).

[36] Blunsom, Phil, Krystle Kocik, and James R. Curran. "Question classification with log-linear models." In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 615-616. ACM, 2006.

[37] Huang, Zhiheng, Marcus Thint, and Asli Celikyilmaz. "Investigation of question classifier in question answering." In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pp. 543-550. Association for Computational Linguistics, 2009.

[38] Le Nguyen, Minh, Nguyen Thanh Tri, and Akira Shimazu. "Subtree Mining for Question Classification Problem." In *IJCAI*, pp. 1695-1700. 2007.

[39] Li, Xin, and Dan Roth. "Learning question classifiers." In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pp. 1-7. Association for Computational Linguistics, 2002.

[40] Ying-wei, Li, Yu Zheng-tao, Meng Xiang-yan, Che Wen-gang, and Mao Cun-li. "Question classification based on incremental modified Bayes." In *Future Generation*

*Communication and Networking, 2008. FGCN'08. Second International Conference on*, vol. 2, pp. 149-152. IEEE, 2008.

[41] Radev, Dragomir, Weiguo Fan, Hong Qi, Harris Wu, and Amardeep Grewal. "Probabilistic question answering on the web." *Journal of the American Society for Information Science and Technology* 56, no. 6 (2005): 571-583.

[42] Pinto, David, Michael Branstein, Ryan Coleman, W. Bruce Croft, Matthew King, Wei Li, and Xing Wei. "QuASM: a system for question answering using semi-structured data." In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pp. 46-55. ACM, 2002.

[43] Li, Xin, and Dan Roth. "Learning question classifiers: the role of semantic information." *Natural Language Engineering* 12, no. 03 (2006): 229-249.

[44] Krishnan, Vijay, Sujatha Das, and Soumen Chakrabarti. "Enhanced answer type inference from questions using sequential models." In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 315-322. Association for Computational Linguistics, 2005.

[45] ] Li, Fangtao, Xian Zhang, Jinhui Yuan, and Xiaoyan Zhu. "Classifying what-type questions by head noun tagging." In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pp. 481-488. Association for Computational Linguistics, 2008.

[46] Li, Xin, Xuan-Jing Huang, and Li-de WU. "Question classification using multiple classifiers." In *Proceedings of the 5th Workshop on Asian Language Resources and First Symposium on Asian Language Resources Network*. 2005.

[47] Zhang, Dell, and Wee Sun Lee. "Question classification using support vector machines." In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 26-32. ACM, 2003.

[48] May, Richard, and Ari Steinberg. "AL, Building a Question Classifier for a TREC-Style Question Answering System." *AL: The Stanford Natural Language Processing Group, Final Projects* (2004).

[49] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature* 521, no. 7553 (2015): 436.

[50] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[51] Wang, J., Yu, L. C., Lai, K. R., & Zhang, X. (2016). Dimensional sentiment analysis using a regional CNN-LSTM model. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (Vol. 2, pp. 225-230).

## Abstract:

With the rapid growth of accessible stored data, the process of information retrieval and extraction has become more important than ever. Question answering systems are a form of information retrieval systems that can give users summarized and effective answers. Interactive question answering systems can reduce ambiguities in the question and increase accuracy of the answer by having a two-way interaction with the user.

After receiving the question, one of the steps in question analysis is question classification. Multiple machine learning methods for feature extraction and question classification have been used in interactive question answering systems. On the other hand deep learning has improved machine learning in many areas. In this research by using deep learning and the features that it extracts we provide a new method for extracting more relevant features and classifying questions.

In our method we use a pre-trained word2vec network for word representation. Additionally feature extraction and question classification is done by combining a convolutional neural network and an LSTM neural network. The designed network can use previous sentences of a dialog in case its accuracy is below a given threshold. Tests of our network show a 4.2 percent improvement compared to other machine learning methods for our collected dataset.

**Keywords: retrival system, interactive question answering system, question classification, deep learning**

**Shahrood University of Technology**

**Faculty of Computer Engineering**

**M.Sc. Thesis in Artificial Intelligence Engineering**

# Question classification in conversational system using artificial intelligence techniques

By:

Ghazaleh MoghadamNejad

Supervisor:

Dr. Morteza Zahedi

Jan 2018