

به نام خدا



دانشگاه تهران  
دانشکده فنی  
دانشکده مهندسی برق  
و کامپیوتر



## درس بازیابی هوشمند اطلاعات

پاسخ بخش تئوری تمرین ۴

نام و نام خانودگی: روژین پناو

شماره دانشجویی: ۲۲۰۷۰۱۰۴۶

تایید می‌کنم که از LLMها مطابق با دستورالعمل‌های بارگذاری شده در سامانه Elearn درس به طور مسئولانه استفاده کرده‌ام. تمام اجزای کار خود را درک می‌کنم و آماده بحث شفاهی درباره آنها هستم.

## آذر ماه ۱۴۰۴

### فهرست

- سؤال ۱ - روابط Word Association..... ۳
- سؤال ۲ - تحلیل Mutual Information..... ۴
- سؤال ۳ - ماهیت فضای Word Embedding..... ۵
- سؤال ۴ - مقایسه CBOW و Skip-gram..... ۶
- سؤال ۵ - رویکردهای Pointwise، Pairwise و Listwise در LTR..... ۶
- سؤال ۶ - سازگاری LambdaRank با معیارهای رتبه‌بندی..... ۷
- سؤال ۷ - شباهت واژگان با ویژگی‌های Interpretable..... ۹

## سؤال ۱ - روابط WORD ASSOCIATION

(الف)

بهترین راه برای درک این دو رابطه این است که ابتدا نقش هر واژه در جمله را بررسی کنیم. رابطه‌ی paradigmatic بیشتر روی "جانشینی" تمرکز دارد؛ یعنی واژه‌ها به خاطر شباهت در نقش دستوری یا موقعیتشان می‌توانند جای یکدیگر را بگیرند و جمله همچنان درست باشد. به عبارتی، این واژه‌ها در یک دسته هم‌رده هستند و انتخاب یکی از آن‌ها بستگی به معنای موردنظر دارد.

در مقابل، رابطه‌ی syntagmatic روی "کنار هم آمدن" واژه‌ها تمرکز دارد. اینجا مسئله جایگزینی نیست، بلکه این واژه‌ها به خاطر سازگاری معنایی یا نحوی، معمولاً با هم ظاهر می‌شوند و یک ترکیب طبیعی می‌سازند. به طور ساده، paradigmatic مربوط به گزینه‌های هم‌رده است و syntagmatic مربوط به هم‌نشینی در جمله.

(ب)

Paradigmatic نوعی شباهت نقشی را نشان می‌دهد؛ یعنی واژه‌ها از نظر جایگاه و نقش دستوری در جمله به هم شبیه‌اند و می‌توانند جای هم استفاده شوند. این شباهت معمولاً به دسته‌بندی‌های معنایی و واژگانی نزدیک است.

اما syntagmatic بیشتر بر "هم‌وقوعی" تمرکز دارد. واژه‌ها ممکن است از نظر دستوری متفاوت باشند، اما به دلیل ارتباط معنایی یا کارکردی، کنار هم دیده می‌شوند. این نوع شباهت نشان می‌دهد که دانستن یکی از واژه‌ها احتمال حضور دیگری را در همان بافت افزایش می‌دهد.

(پ)

برای syntagmatic مثال "کتاب" و "خواندن" مناسب است. این دو واژه معمولاً با هم دیده می‌شوند، چون خواندن کاری است که روی کتاب انجام می‌شود. اینجا جایگزینی مطرح نیست و واژه‌ها نقش‌های متفاوت دارند، اما کنار هم معنا و ارتباطشان تکمیل می‌شود.

برای paradigmatic می‌توان از رنگ‌ها مثال زد: در جمله‌ی "رنگ مداد آبی است"، واژه‌ی "آبی" می‌تواند با "قرمز" یا "سبز" جایگزین شود. جمله همچنان درست است و فقط اطلاعات رنگ تغییر می‌کند. این نشان می‌دهد که این واژه‌ها در یک مجموعه هم‌رده قرار دارند.

الف)

می‌توان استدلال کرد که Mutual Information (MI) به‌طور مستقیم میزان وابستگی آماری دو متغیر تصادفی را اندازه می‌گیرد؛ در متن‌های زبانی این متغیرها معمولاً حضور یا غیاب یک واژه در یک سند هستند. وقتی دو واژه در یک موضوع مشترک ظاهر می‌شوند، احتمال هم‌وقوعی آن‌ها از ضرب احتمال‌های فردی‌شان بیشتر می‌شود؛ MI همین اختلاف بین هم‌وقوعی واقعی و هم‌وقوعی مورد انتظار در صورت استقلال را می‌سنجد. بنابراین حتی اگر معنای لغوی دو واژه مشابه نباشد، چون در همان موقعیت‌های موضوعی تکراراً کنار هم ظاهر می‌شوند، مقدار MI آن‌ها بالا خواهد بود. یعنی نشان‌دهنده‌ی "ارتباط موضوعی" است، نه لزوماً "هم‌معنایی".

برای نمایش ریاضی این ایده، MI بین دو متغیر تصادفی (X) و (Y) به صورت زیر تعریف می‌شود:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \left( \frac{p(x,y)}{p(x) \cdot p(y)} \right)$$

که در آن  $p(x,y)$  احتمال هم‌وقوع رخداد (x) و (y) و  $p(x)$  و  $p(y)$  احتمال‌های حاشیه‌ای هر یک از رخدادها هستند. اگر واژه‌ها مستقل باشند، نسبت داخل لگاریتم برابر ۱ می‌شود و مقدار MI صفر خواهد بود؛ اما وقتی هم‌وقوعی بیش از انتظار وجود داشته باشد، آن نسبت بزرگتر از ۱ شده و MI مثبت و بزرگ می‌شود. از نظر مفهومی، همین خاصیت است که MI را برای کشف "هم‌نشینی موضوعی" مناسب می‌سازد.

- MI متقارن است) یعنی  $I(X;Y) = I(Y;X)$  و مقدارش نشان‌دهنده‌ی مقدار اطلاعاتی است که دانستن یکی از واژه‌ها درباره‌ی دیگری می‌دهد.

- MI نسبت به افزایش هم‌وقوعی‌های غیرمنتظره حساس است، بنابراین واژه‌هایی که در یک موضوع خاص همزمان ظاهر می‌شوند (حتی از دسته‌های معنایی متفاوت) را برجسته می‌کند.

- محدودیت مهم MI: برای رخداد‌های خیلی نادر می‌تواند مقدار بزرگ و گمراه‌کننده تولید کند؛ به همین دلیل در کاربردهای عملی معمولاً از آستانه‌ها، نرمال‌سازی یا حداقل فرکانس استفاده می‌شود.

ب)

یک مثال واقعی که در تحلیل متون خبری دیده می‌شود، جفت واژه‌های "واکسن" و "بورس" است. این دو واژه از نظر معنایی در سطح لغوی ارتباط مستقیمی ندارند. یکی به حوزه سلامت و دیگری به حوزه مالی مربوط است. اما در دوره‌هایی مانند پاندمی کرونا مقاله‌ها و گزارش‌های خبری زیادی وجود داشت که همزمان درباره‌ی پیشرفت واکسیناسیون و تأثیر آن بر بازار سهام مطلب می‌نوشتند. در چنین مجموعه داده‌هایی احتمال

هم‌وقوع "واکسن" و "بورس" از آنچه از فرکانس‌های فردی انتظار می‌رود، بیشتر است؛ در نتیجه مقدار MI بین آن‌ها بالا خواهد بود.

### سؤال ۳ - ماهیت فضای WORD EMBEDDING

فضای word embedding نوعی نمایش برداری برای واژه‌ها فراهم می‌کند که در آن هر واژه به جای برچسب متنی منفرد، به صورت یک بردار چندبعدی در نظر گرفته می‌شود. این بردارها نه تنها هویت واژه را مشخص می‌کنند، بلکه الگوهای معنایی و نحوی آن در زبان را نیز رمزگذاری می‌کنند. در عمل، مدل با پردازش مجموعه‌های عظیمی از متن، یاد می‌گیرد که کدام واژه‌ها معمولاً در زمینه‌های مشابه ظاهر می‌شوند و کدام‌ها با یکدیگر هم‌نشینی دارند. بنابراین، فاصله و جهت بردارها در این فضا منعکس‌کننده شباهت معنایی و نقش نحوی واژه‌ها است: واژه‌هایی که در بافت‌ها یا حوزه‌های مشابه به کار می‌روند، بردارهای نزدیکتری دارند و واژه‌هایی با نقش دستوری مشابه، در نواحی همجوار فضا قرار می‌گیرند.

علاوه بر این، روابط ضمنی واژه‌ها نیز در embedding ها بازتاب می‌یابند. حتی اگر دو واژه از نظر معنایی متفاوت باشند، اما در متن‌های رایج به طور مکرر با هم ظاهر شوند، بردارهای آن‌ها تمایل به نزدیکی دارند، که این ویژگی امکان تحلیل هم‌نشینی واژه‌ها را فراهم می‌کند.

مفهوم خطی بودن در فضای embedding

فضای embedding به شکلی سازمان‌دهی شده که برخی روابط میان واژه‌ها تقریباً خطی هستند. منظور از خطی بودن این است که تفاوت‌ها و ویژگی‌های زبانی می‌توانند با عملیات ساده جمع و تفریق بردارها مدل‌سازی شوند. به طور هندسی، اگر رابطه مشخصی بین دو واژه بارها تکرار شده باشد (مثلاً رابطه جنسیت، مفرد/جمع، زمان فعل، یا ارتباط پایتخت و کشور)، مدل تمایل دارد یک بردار جابه‌جایی پایدار برای آن رابطه ایجاد کند. به این ترتیب، اختلاف برداری بین واژه‌های مرتبط در نمونه‌های متعدد مشابه باقی می‌ماند.

یک نمونه از این ویژگی می‌تواند رابطه میان

$$\text{prince} - \text{boy} + \text{girl} \approx \text{princess}$$

باشد. در این مثال، بردار واژه prince از نظر معنایی به princess نزدیک است، با این تفاوت که اولی مذکر و دومی مونث است. بردار boy و girl تفاوت جنسیتی را نشان می‌دهند. هنگامی که بردار boy از prince کم می‌شود، مؤلفه‌های مرتبط با ویژگی مذکر کاهش می‌یابد، و با افزودن بردار girl، همان مفهوم کلی حفظ می‌شود ولی جنسیت به سمت مونث تغییر می‌کند. در نتیجه، بردار حاصل به نقطه‌ای در فضا نزدیک

می‌شود که هم ویژگی سلطنت را حفظ می‌کند و هم به جنسیت مونث تمایل دارد، که همان بردار princess است.

این روابط خطی یک تقریب تجربی هستند و به‌طور کامل ریاضی دقیق نیستند. موفقیت آن‌ها بستگی به کیفیت داده‌های آموزشی، میزان چندمعنایی واژه‌ها و تنوع بافت‌های زبانی دارد. همچنین embedding ها ممکن است سوگیری‌های موجود در متن‌ها را نیز در این جهت‌ها و فواصل منعکس کنند.

#### سؤال ۴ - مقایسه CBOW و SKIP-GRAM

در مدل Word2Vec دو روش CBOW و Skip-gram برای یادگیری بردارهای کلمات استفاده می‌شوند که تفاوت اصلی آن‌ها در جهت پیش‌بینی است. در روش CBOW، مدل از روی کلمات اطراف یک کلمه تلاش می‌کند خود کلمه مرکزی را پیش‌بینی کند. به عبارت دیگر، ورودی مدل چند کلمه‌ای است که در پیرامون کلمه هدف قرار دارند و خروجی همان کلمه مرکزی است. در این روش معمولاً ترتیب کلمات با اهمیت کمتری در نظر گرفته می‌شود و بردارها با میانگین‌گیری یا جمع ترکیب می‌شوند تا پیش‌بینی انجام شود.

در مقابل، روش Skip-gram برعکس عمل می‌کند. یعنی کلمه مرکزی به عنوان ورودی مدل داده می‌شود و مدل سعی می‌کند کلمات اطراف آن را پیش‌بینی کند. بنابراین از هر رخداد یک کلمه، چندین نمونه آموزشی ساخته می‌شود که هر کدام مربوط به یک کلمه بافت است.

در مورد کاربردها، CBOW برای کلمات پرتکرار بهتر عمل می‌کند، چون با میانگین‌گیری سیگنال‌های آموزشی، بردارهای پایدار و کم‌نویز برای این واژه‌ها ساخته می‌شود و یادگیری سریع‌تر انجام می‌گیرد. از سوی دیگر، Skip-gram برای کلمات کم‌تکرار مناسب‌تر است، زیرا از هر رخداد یک کلمه کم‌تکرار، چندین نمونه آموزشی تولید می‌شود و بنابراین حتی با داده محدود، بردار دقیق‌تری برای آن کلمه یاد گرفته می‌شود.

#### سؤال ۵ - رویکردهای POINTWISE، PAIRWISE و LISTWISE در LTR

Pointwise

در این روش هر آیت به تنهایی مدل می‌شود؛ برچسب همان نمره یا درجه مرتبط بودن هر سند با یک پرسوجو است. تابع هدف به صورت پیش‌بینی مستقیم این نمره تعریف می‌شود و مدل شبیه یک مسئله رگرسیون یا طبقه‌بندی ساده آموزش می‌بیند. مزیت اصلی سادگی و سرعت آموزش و

پیاده‌سازی است و برای مجموعه داده‌های بزرگ مقیاس‌پذیر است، اما عیبش این است که ترتیب نسبی بین آیتم‌ها را به‌طور مستقیم مدل نمی‌کند و بنابراین با معیارهای مرتب‌سازی رتبه‌ای مثل NDCG کمتر هم‌راستاست.

#### Pairwise

در این رویکرد مدل روی زوج‌های آیتم کار می‌کند؛ برچسب برای هر جفت مشخص می‌کند کدام یک باید بالاتر قرار گیرد. تابع هدف تلاش می‌کند احتمال یا ترتیب درست بین دو آیتم را یاد بگیرد، یعنی خطاها مربوط به جفت‌هایی است که ترتیبشان اشتباه پیش‌بینی شده. این روش نسبت به Pointwise بهتر ترتیب نسبی را پوشش می‌دهد و معمولاً با معیارهای ترتیبی مانند NDCG همخوانی بیشتری دارد. محدودیت اصلی افزایش نمایی تعداد جفت‌ها و در نتیجه هزینه محاسباتی و زمان آموزش است، به‌ویژه وقتی تعداد آیتم‌ها در هر پرس‌وجو زیاد شود.

#### Listwise

در این روش کل لیست اسناد مرتبط با یک پرس‌وجو به‌صورت یک واحد در نظر گرفته می‌شود و برچسب، ترتیب یا امتیاز کل لیست است. تابع هدف به‌طور مستقیم سعی می‌کند ترتیبی نزدیک‌تر به ترتیب مطلوب تولید کند و بنابراین از نظر نظری با معیارهایی مثل NDCG بهترین همخوانی را دارد. مزیتش توانایی مدل‌سازی وابستگی‌های میان تمام عناصر لیست و بهینه‌سازی مستقیم معیارهای رتبه‌ای است؛ اما معایب شامل پیچیدگی محاسباتی بالاتر، نیاز بیشتر به حافظه و داده و دشواری پیاده‌سازی در مقیاس بزرگ است.

کدام مناسب‌تر است

اگر هدف بهینه‌سازی معیارهایی مانند NDCG است، Listwise بهترین تطابق را دارد و پس از آن Pairwise قرار می‌گیرد، چون هر دو به ترتیب نسبی و اهمیت رتبه‌ای توجه نشان می‌دهند Pointwise. در عمل زمانی مناسب است که سادگی و سرعت اهمیت بیشتری داشته باشند یا برچسب‌های عددی مستقلی در دسترس باشند، اما برای کسب نتایج بهتر در معیارهای رتبه‌ای معمولاً کمتر ترجیح داده می‌شود.

### سؤال ۶ - سازگاری LAMBDARANK با معیارهای رتبه‌بندی

LambdaRank یک روش عملیاتی و مبتنی بر جفت (pairwise) برای یادگیری رتبه‌بندی است که به جای تعریف صریح یک تابع خطا و مشتق‌گیری از آن، از «سیگنال‌های گرادیان‌مانند» به نام  $\lambda$  (لامبدا) استفاده می‌کند. پاسخ‌های زیر به‌صورت ساده و دانشجویی توضیح می‌دهند که این  $\lambda$  ها چه هستند، چرا جابجایی‌ها در بالای فهرست اهمیت بیشتری دارند و چگونه این روش با معیارهایی مثل NDCG هم‌راستا می‌شود.

مفهوم  $\lambda$ -gradient چیست؟

$\lambda$ -gradient در واقع یک مقدار عددی است که برای هر جفت سند (که ترتیبشان باید نسبت به هم اصلاح شود) محاسبه می‌شود و نشان می‌دهد

چگونه باید امتیاز پیش‌بینی‌شده برای آن اسناد تغییر یابد تا ترتیب بهتر شود. به عبارت ساده، برای هر جفتی که ترتیب واقعی‌شان با ترتیب پیش‌بینی شده نادرست است، مقداری  $\lambda$  محاسبه می‌شود و آن مقدار به عنوان یک سیگنال به شبکه یا مدل داده می‌شود تا وزن‌ها طوری به‌روزرسانی شوند که ترتیب درست‌تر شود. برخلاف مشتق دقیق یک معیار ترتیبی که اغلب ناهموار یا غیرقابل‌تفاضل است،  $\lambda$  نقش یک «گرادیان مصنوعی» را بازی می‌کند که جهت و شدت اصلاح را مشخص می‌کند.

چرا تغییر رتبه سندهای بالای لیست نسبت به سندهای پایین‌تر  $\text{gradient}$  بزرگتری ایجاد می‌کند؟

معیارهای رتبه‌بندی مانند NDCG وزن بیشتری به جایگاه‌های بالای فهرست می‌دهند، یعنی اصلاح یک خطا در جایگاه اول تأثیر بسیار بزرگتری روی مقدار معیار دارد تا اصلاح همان خطا در جایگاه بیستم. در  $\text{LambdaRank}$  مقدار  $\lambda$  برای یک جفت معمولاً متناسب با تغییر در معیار (مثلاً NDCG) است که در صورت جابجایی آن دو سند به‌دست می‌آید. بنابراین اگر جابجایی دو سند موجب افزایش یا کاهش زیادی در NDCG شود (که غالباً وقتی یکی از آن‌ها در موقعیت‌های بالا باشد رخ می‌دهد)، مقدار  $\lambda$  بزرگتر خواهد بود تا مدل قوی‌تر به آن جابجایی واکنش نشان دهد. به زبان ساده: چون معیاری مثل NDCG به بالای لیست حساس‌تر است، هر تغییری در آن ناحیه پیامد قوی‌تری دارد و  $\text{LambdaRank}$  هم این را با  $\lambda$  های بزرگتر منعکس می‌کند.

چگونه این روش به‌طور مؤثر با معیارهایی مانند NDCG سازگار می‌شود؟  $\text{LambdaRank}$  به‌طور مستقیم از تغییرات معیار (مثلاً NDCG) برای وزن‌دهی به آپدیت‌های جفت‌ها استفاده می‌کند. این یعنی جفت‌هایی که با جابجایی‌شان بیشترین بهبود در معیار حاصل می‌شود، بزرگترین سیگنال‌های آموزشی را می‌گیرند. از این رو، به‌جای بهینه‌سازی یک تابع غلطی عمومی که ممکن است با NDCG بی‌ارتباط باشد،  $\text{LambdaRank}$  به شکل تقریبی اما هدفمند سعی می‌کند همان چیزی را که NDCG اندازه می‌گیرد بهبود دهد. این کار سه فایده دارد: اول اینکه بهینه‌سازی متمرکز روی چیزی انجام می‌گیرد که در نهایت برای ارزیابی مهم است؛ دوم اینکه از مشکل غیرقابل‌تفاضل بودن معیارها اجتناب می‌شود چون  $\Delta\text{NDCG}$  به‌عنوان وزن در گرادیان‌ها به کار می‌رود؛ و سوم اینکه مدل قادر می‌شود روی اصلاح اشتباهات مهم‌تر (آنهایی که NDCG را بیشتر تحت‌تأثیر قرار می‌دهند) تمرکز کند.



## سؤال ۷ - شباهت واژگان با ویژگی‌های INTERPRETABLE

(الف)

ترتیب ویژگی‌ها : [is\_animal, has\_tail, can\_fly, is\_fruit, is\_vehicle, is\_food]

بردارهای ویژگی:

$$\text{dog} = [1, 1, 0, 0, 0, 0]$$

$$\text{cat} = [1, 1, 0, 0, 0, 0]$$

$$\text{bird} = [1, 0, 1, 0, 0, 0]$$

$$\text{apple} = [0, 0, 0, 1, 0, 1]$$

$$\text{banana} = [0, 0, 0, 1, 0, 1]$$

$$\text{car} = [0, 0, 0, 0, 1, 0]$$

(ب)

: cosine similarity

$$\cos(x, y) = \frac{x \cdot y}{|x| |y|}$$

$$x \cdot y = \sum_{i=1}^6 x_i y_i,$$

$$|x| = \sqrt{\sum_{i=1}^6 x_i^2}, \quad |y| = \sqrt{\sum_{i=1}^6 y_i^2}$$

(۱)  $\cos(\text{cat}, \text{dog})$

$$\text{cat} = [1, 1, 0, 0, 0, 0], \quad \text{dog} = [1, 1, 0, 0, 0, 0]$$

$$\text{cat} \cdot \text{dog} = (1 \cdot 1) + (1 \cdot 1) + (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) = 1 + 1 = 2$$

$$|\text{cat}| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 0^2 + 0^2} = \sqrt{2}$$

$$|\text{dog}| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 0^2 + 0^2} = \sqrt{2}$$

$$\cos(\text{cat}, \text{dog}) = \frac{2}{\sqrt{2} \cdot \sqrt{2}} = \frac{2}{2} = 1$$

$$\cos(\text{bird,dog}) \quad (\Psi)$$

$$\text{bird} = [1,0,1,0,0,0], \text{ dog} = [1,1,0,0,0,0]$$

$$\text{bird} \cdot \text{dog} = (1 \cdot 1) + (0 \cdot 1) + (1 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) = 1 + 0 + 0 = 1$$

$$|\text{bird}| = \sqrt{1^2 + 0^2 + 1^2 + 0^2 + 0^2 + 0^2} = \sqrt{2}$$

$$|\text{dog}| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 0^2 + 0^2} = \sqrt{2}$$

$$\cos(\text{bird,dog}) = \frac{1}{\sqrt{2} \cdot \sqrt{2}} = \frac{1}{2} = 0.5$$

$$\cos(\text{banana,apple}) \quad (\Psi)$$

$$\text{banana} = [0,0,0,1,0,1], \text{ apple} = [0,0,0,1,0,1]$$

$$\text{banana} \cdot \text{apple} = (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) + (1 \cdot 1) + (0 \cdot 0) + (1 \cdot 1) = 1 + 1 = 2$$

$$|\text{banana}| = \sqrt{0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 1^2} = \sqrt{2}$$

$$|\text{apple}| = \sqrt{0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 1^2} = \sqrt{2}$$

$$\cos(\text{banana,apple}) = \frac{2}{\sqrt{2} \cdot \sqrt{2}} = \frac{2}{2} = 1$$

$$\cos(\text{apple,dog}) \quad (\Psi)$$

$$\text{apple} = [0,0,0,1,0,1], \text{ dog} = [1,1,0,0,0,0]$$

$$\text{apple} \cdot \text{dog} = (0 \cdot 1) + (0 \cdot 1) + (0 \cdot 0) + (1 \cdot 0) + (0 \cdot 0) + (1 \cdot 0) = 0$$

$$|\text{apple}| = \sqrt{0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 1^2} = \sqrt{2}$$

$$|\text{dog}| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 0^2 + 0^2} = \sqrt{2}$$

$$\cos(\text{apple,dog}) = \frac{0}{\sqrt{2} \cdot \sqrt{2}} = 0$$

$$\cos(\text{car,apple}) \quad (\Delta)$$

$$\text{car} = [0,0,0,0,1,0], \text{ apple} = [0,0,0,1,0,1]$$

$$\text{car} \cdot \text{apple} = (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 0) + (0 \cdot 1) + (1 \cdot 0) + (0 \cdot 1) = 0$$

$$|\text{car}| = \sqrt{0^2 + 0^2 + 0^2 + 0^2 + 1^2 + 0^2} = 1$$

$$|\text{apple}| = \sqrt{0^2 + 0^2 + 0^2 + 1^2 + 0^2 + 1^2} = \sqrt{2}$$

$$\cos(\text{car,apple}) = \frac{0}{1 \cdot \sqrt{2}} = 0$$

- بیشترین شباهتها بین جفت‌هایی دیده می‌شود که ویژگی‌های مشترک زیادی دارند. برای مثال، cat و dog شباهت کامل دارند زیرا هر دو حیوان هستند و دم دارند، بنابراین مقدار مشابهت کسینوس آنها برابر ۱ است. همین‌طور، apple و banana هم بیشترین شباهت را دارند چون هر دو میوه هستند و ویژگی‌های مشابه دیگری دارند. در مقابل، جفت‌هایی مانند apple و dog یا car و apple هیچ ویژگی مشترکی ندارند و مشابهتشان صفر است.
- در این فضای ویژگی، شباهت بر اساس **اشتراک ویژگی‌های قابل مشاهده** است. یعنی بردارها مشخص می‌کنند که کدام ویژگی‌ها بین دو شیء مشترک است، مثل حیوان بودن، داشتن دم، قابلیت پرواز، یا نوع خوراکی بودن. این شباهت کاملاً قابل تفسیر است و می‌توان به راحتی دلیل نزدیکی یا دوری دو شیء را تشخیص داد.
- در word embedding شباهت فقط بر اساس ویژگی‌های قابل مشاهده نیست؛ بلکه روابط معنایی و زمینه‌ای بین کلمات هم در نظر گرفته می‌شود. به عنوان مثال، کلمات «king» و «queen» در فضای embedding نزدیک‌اند حتی اگر ویژگی‌های مشخص آنها متفاوت باشد، زیرا در متن‌ها اغلب در زمینه‌های مشابه ظاهر می‌شوند. بنابراین بردارهای embedding علاوه بر شباهت‌های صریح، روابط ضمنی و نحوی را هم ضبط می‌کنند.
- نمایش مبتنی بر ویژگی صریح برای paradigmatic similarity مناسب‌تر است، چون می‌تواند شباهت بین اشیاء یا کلمات هم‌رده مانند cat و dog را به‌طور مستقیم مدل کند. در حالی که word embedding برای syntagmatic similarity بهتر است، زیرا روابط زمینه‌ای و ترتیبی بین کلمات را نشان می‌دهد و وابستگی‌های نحوی یا ترکیبی را مدل می‌کند.