توکنایز کردن (Tokenization)

توکنایز کردن فرآیندی در پردازش زبان طبیعی (NLP) است که متن را به واحدهای کوچک‌تر به نام "توکن" تقسیم می‌کند. این توکن‌ها معمولاً کلمات، عبارات، جمله‌ها یا حتی کاراکترها هستند. هدف از توکنایز کردن، ساده‌تر کردن تجزیه و تحلیل متن است.

مثال:

متن: "Hello, world! This is a test."

- توکن‌ها: ["Hello", "world", "This", "is", "a", "test"]

ریشه‌یابی (Lemmatization) و استمینگ (Stemming)

هر دو فرآیند ریشه‌یابی و استمینگ برای کاهش کلمات به ریشه‌های خود استفاده می‌شوند تا کلمات با اشکال مختلف به یک ریشه یا پایه مشترک تبدیل شوند. این کمک می‌کند تا تحلیل‌های متنی بهتر انجام شود.

Lemmatization

Lemmatization کلمات را به شکل پایه‌ای یا قاموسی خود برمی‌گرداند که به آن لِما (lemma) گفته می‌شود. این فرآیند معمولاً از دیکشنری‌ها و تحلیل‌های صرفی برای تعیین شکل پایه کلمه استفاده می‌کند.

مثال:

- کلمات: ["running", "ran", "runs"]

- لِما: ["run"]

Stemming

Stemming فرآیندی است که کلمات را به ریشه یا استم (stem) خود برمی‌گرداند. این فرآیند اغلب با حذف پسوندها و پیشوندها انجام می‌شود و نسبت به Lemmatization ساده‌تر و سریع‌تر است، اما دقت کمتری دارد.

مثال:

- کلمات: ["running", "ran", "runs"]

- استم: ["run"]

حذف کلمات توقف (Stopwords Removal)

حذف کلمات توقف فرآیندی است که کلمات بی‌اهمیت و پر تکراری که در اکثر جملات و متون وجود دارند و اطلاعات معنایی زیادی ندارند، از متن حذف می‌شوند. این کلمات معمولاً شامل حروف اضافه، ضمایر، حروف ربط و سایر کلمات رایج هستند.

مثال:

- جمله: "This is a sample sentence"

- پس از حذف کلمات توقف: "sample sentence"

حذف علائم نشانه‌گذاری (Punctuation Removal)

حذف علائم نشانه‌گذاری به معنی حذف کاراکترهایی مانند نقطه، کاما، نقل‌قول‌ها، علامت سوال و سایر علائم از متن است. این علائم معمولاً در تجزیه و تحلیل معنایی متن نقشی ندارند و حذف آنها می‌تواند به ساده‌تر شدن پردازش کمک کند.

مثال:

- جمله: "Hello, world! This is a test."

- پس از حذف علائم نشانه‌گذاری: "Hello world This is a test"

این تکنیک‌ها برای بهبود کیفیت تجزیه و تحلیل متون در کاربردهای مختلف NLP بسیار مهم هستند. آنها به ما کمک می‌کنند تا متون را بهتر پردازش و تحلیل کنیم، مدل‌های یادگیری ماشین را بهبود بخشیم و در نهایت نتایج دقیق‌تری را بدست آوریم.

Elbow Method:

ابتدا با استفاده از روش Elbow تعداد بهینه خوشه‌ها را تعیین می‌کنیم. این کار با رسم نمودار تحریف (distortion)و اینرسی (inertia) برای تعداد مختلف خوشه‌ها انجام می‌شود.

تعداد خوشه بهینه معمولاً در نقطه‌ای است که نمودار Elbow تغییر زیادی را نشان می‌دهد.

K-Means Clustering

پس از تعیین تعداد بهینه خوشه‌ها، مدل K-Means را آموزش می‌دهیم و خوشه‌ها را پیش‌بینی می‌کنیم.

DBSCAN Clustering

با استفاده از DBSCAN که یک روش خوشه‌بندی مبتنی بر چگالی است، خوشه‌ها را پیدا می‌کنیم.

پارامترهای eps و min_samples باید با توجه به داده‌ها تنظیم شوند.

Hierarchical Clustering

با استفاده از خوشه‌بندی سلسله‌مراتبی و تعداد بهینه خوشه‌ها، خوشه‌ها را پیدا می‌کنیم.

برای مقایسه نتایج خوشه‌بندی‌ها از معیار Silhouette Score استفاده می‌کنیم. این معیار به ما کمک می‌کند تا کیفیت خوشه‌بندی را ارزیابی کنیم.

Silhouette Score یکی از معیارهای رایج برای ارزیابی کیفیت خوشه‌بندی است. دلایل استفاده از این معیار عبارتند از:

اندازه گیری جدایی و فشردگی

- Silhouette Score به طور همزمان میزان جدایی (separation) خوشه‌ها از یکدیگر و میزان فشردگی (cohesion) نقاط درون هر خوشه را ارزیابی می‌کند. این معیار به ما می‌گوید که چقدر نقاط داده در یک خوشه به مرکز خوشه نزدیک هستند و چقدر خوشه‌ها از هم دور هستند.

بازه نمره‌دهی واضح

- Silhouette Score عددی بین -۱ تا ۱ است. نمرات نزدیک به ۱ نشان‌دهنده خوشه‌بندی خوب، نمرات نزدیک به ۰ نشان‌دهنده خوشه‌بندی نامشخص، و نمرات منفی نشان‌دهنده خوشه‌بندی ضعیف و اشتباه است. این بازه نمره‌دهی واضح، تفسیر نتایج را آسان می‌کند.

عدم نیاز به برچسب‌های واقعی

- برخلاف برخی از معیارهای دیگر ارزیابی (مثل دقت و F1-Score)، Silhouette Score نیاز به برچسب‌های واقعی ندارد و می‌تواند بدون دانش قبلی از ساختار داده‌ها استفاده شود.

روش tf-idf یکی از رایج‌ترین روش‌های محاسبه وزن کلمات در یک متن است که برای استخراج مهم‌ترین کلمات یا اصطلاحات از متن‌ها استفاده می‌شود. این روش بر پایه دو مفهوم اصلی فراوانی تکرار کلمات و برخورد معکوس سند استوار است.

- فراوانی تکرار کلمات (Term Frequency - TF): این مفهوم نشان‌دهنده تعداد ظاهر شدن یک کلمه در یک سند است. در واقع، tf می‌سنجد که یک کلمه چقدر در یک سند تکرار شده است.

- برخورد معکوس سند (Inverse Document Frequency - IDF): این مفهوم نشان‌دهنده میزان "مهمیت" یک کلمه در میان اسناد مختلف است. کلماتی که در اسناد مختلف زیاد تکرار شده‌اند، معمولاً ارزش کمتری دارند چون توانایی تمایز بین اسناد را ندارند.

الگوریتم tf-idf برای محاسبه وزن هر کلمه در یک متن، از ضرب tf و idf استفاده می‌کند. وزن tf-idf یک کلمه در یک سند معمولاً به صورت زیر محاسبه می‌شود:

$$tf\text{-}idf(t,d,D) = tf(t,d) * idf(t,D)$$

که در آن:

- t نشان‌دهنده کلمه مورد نظر است.
- d نشان‌دهنده سند مورد نظر است.
- D نشان‌دهنده کل مجموعه اسناد است.

مزیت اصلی این روش این است که به کلماتی که در اسناد مختلف کمتر تکرار شده‌اند و در عین حال در یک سند خاص زیاد تکرار شده‌اند، وزن بالاتری نسبت می‌دهد. این باعث می‌شود که کلمات کلیدی یا اصطلاحات مهم‌تری که ممکن است در متن‌ها کمتر تکرار شوند، به عنوان کلمات مهم تشخیص داده شوند.