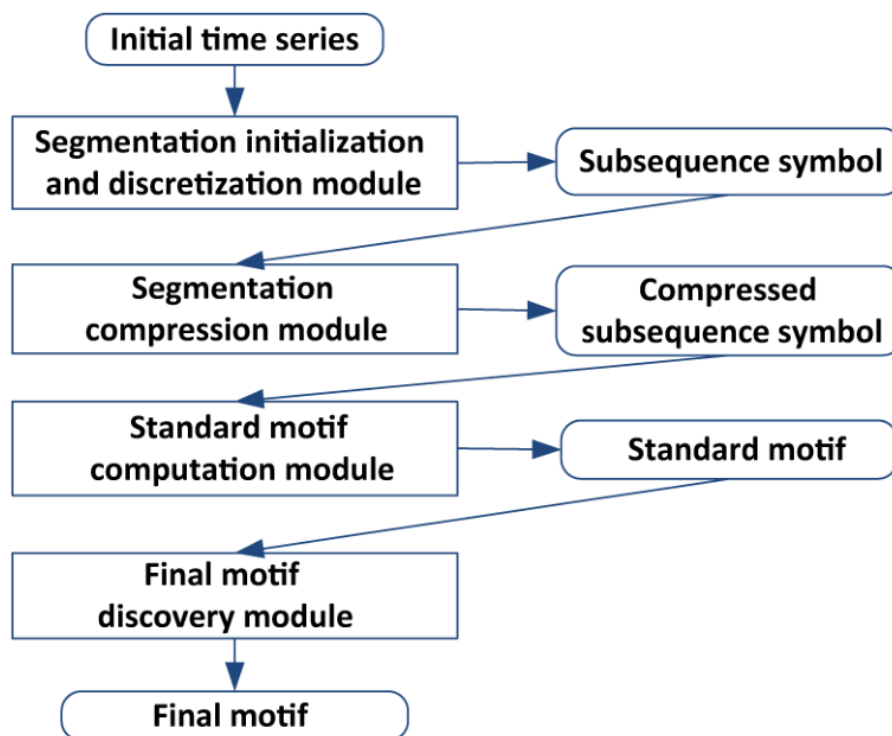
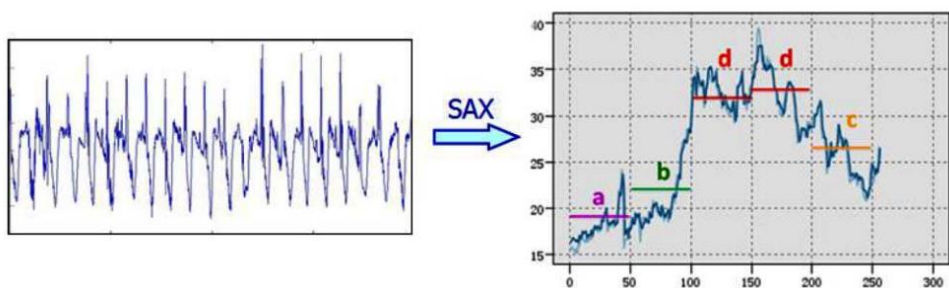


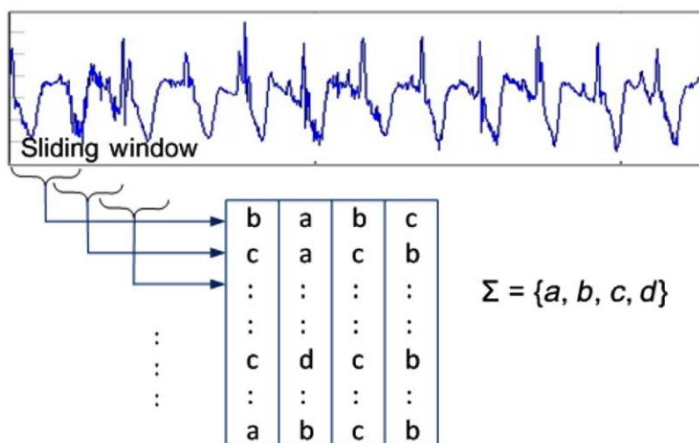
فلوچارت مراحل که طی می کنیم:



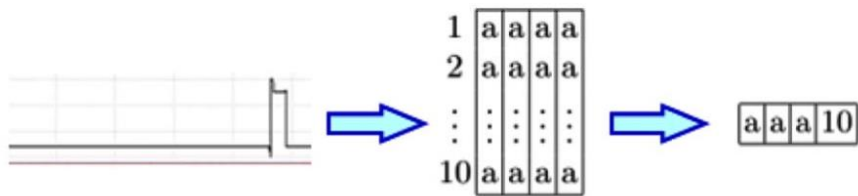
در این مرحله سری زمانی به عنوان ورودی به برنامه خورنده می‌شود. برنامه یک قسمت از آن را مثلاً از ۰ تا ۱۰۰۰ به عنوان مجموعه آموزشی جدا می‌کند. سپس مجموعه آموزشی را توسط الگوریتم های SAX یا PPA به تکه های کوچک تر می شکند. با این کار هر تکه نويز کمتری دارد. روی هر تکه یک برچسب می زند. با استناد به الگوریتم RP با این برچسب ها این سری آموزشی را می توان با یک کلمه نشان داد. مثلاً تعیین می کنیم فقط از مجموعه {a,b,c,d} 4 حرفی، به عنوان برچسب استفاده شود. SAX سری آموزشی شکل را به 5 قسمت شکسته است. بر روی هر قسمت یک برچسب می زند. مطابق با شکل الگوریتم تشخیص داده که قسمت های سوم و چهارم به هم شبیه هستند. به هر دو برچسب d می گذارد.



پنجره لغزان جلو می‌رود و برای هر پنجره یک لغت ساخته می‌شود.



در سری های طولانی از نوع اسپارس بعضی از قسمت های دنباله، صفر و یا نزدیک به صفر است، که به آن ها regular می گویند. گاهی اوقات پرش هایی fluctuant در بعضی زمان ها رویت میشود. اگر این سری ها را به ماتریس تبدیل کنیم به این نوع ماتریس ها اسپارس می گویند. می توان روی زیر دنباله ها یک اسکن سریع انجام داد. اگر یک سمبل پشت سر هم تکرار شده باشد، مثل ۴ تا a به آن بازه trivial می گویند. در این مرحله ما آنها را به یک سمبل فشرده می کنیم.

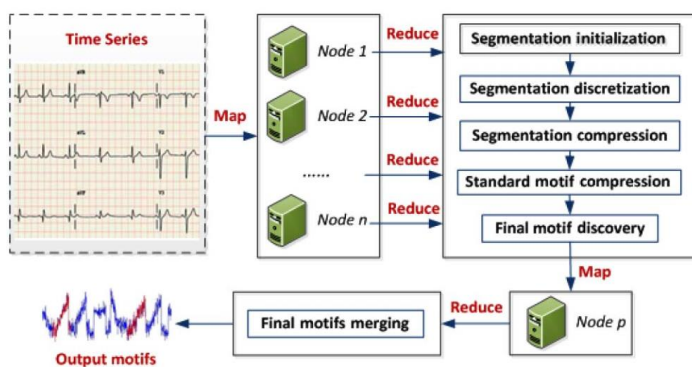


حال که برای هر زیر دنباله کلمه ساختیم، در این مرحله، برای یافتن شباهت بین این کلمات آنها را به ماتریس تداخل نگاشت می کنیم و تابع HASH را روی آن ها اعمال می کنیم.

| | | | | | | | | | | |
|----|---|---|---|---|----|----|----|----|----|---|
| S1 | b | a | b | c | S5 | 0 | 0 | 0 | 0 | 1 |
| S2 | c | a | c | b | S4 | 1 | 0 | 0 | 1 | 0 |
| S3 | c | d | c | b | S3 | 0 | 1 | 1 | 0 | 0 |
| S4 | b | a | b | c | S2 | 0 | 1 | 1 | 0 | 0 |
| S5 | a | b | c | b | S1 | 1 | 0 | 0 | 1 | 0 |
| | | | | | S1 | S2 | S3 | S4 | S5 | |

| | | | | | | | | | | |
|----|---|---|---|---|----|----|----|----|----|---|
| S1 | b | a | b | c | S5 | 0 | 0 | 0 | 0 | 1 |
| S2 | c | a | c | b | S4 | 2 | 0 | 0 | 1 | 0 |
| S3 | c | d | c | b | S3 | 0 | 1 | 1 | 0 | 0 |
| S4 | b | a | b | c | S2 | 0 | 1 | 1 | 0 | 0 |
| S5 | a | b | c | b | S1 | 1 | 0 | 0 | 2 | 0 |
| | | | | | S1 | S2 | S3 | S4 | S5 | |

می توان محاسبات مربوطه برای هر کدام از این زیر دنباله ها به یک گره (ماشین مجازی) بسپاریم. در این راستا از کاهش - نگاشت زیرساخت HADOOP استفاده می کنیم.



در نهایت ماتریس اسپارس به وجود می آید که اگر به صورت نمودار نشان دهیم، مقادیر صفر trace نامیده شده و مقادیر غیر صفر نقاط پراکنده هستند. در این مرحله آستانه را کم می کنیم. نقاط بیشتری روی نمودار ظاهر می شوند. سپس با محاسبه فاصله اقلیدسی، مشخص می کنیم که کدام یک از این زیر دنباله ها می تواند motif باشد.

