

维度	1. 纯 STS (sts_fit.R)	2. BERTopic+STS (run_hybrid_pipeline.py)	3. 纯 BERTopic (run_bertopic_experiment.py)
核心模型	同时估计“主题-情感”双变量的结构主题模型 (Sentiment-Topic-STS)；用同一套潜变量解释词出现概率和情感极性。	两阶段：① BERTopic 无监督分主题→②用经典回归把主题概率回归到协变量（含评分），再把协变量回归到评分，做匹配。STS 仅作为“金标准”被比较。	纯无监督 BERTopic；用 sentence-BERT 编码→HDBSCAN 聚类→c-TF-IDF 生成词；完全不触碰 STS。
文本表示	词袋 (STM 内部 Count+TF 权重)，人工清洗停用词、数字、URL。	词袋：仅用于生成主题-文档概率；后续分析用该概率当数值变量。	sentence-BERT 语义向量 (384 维) 驱动聚类；词袋仅用于 c-TF-IDF 生成关键词。
主题发现机制	生成式：最大化文档-词共现似然+情感协变量；K 由 held-out 似然+coherence 网格 5-7 选最优。	判别式：先用 BERTopic 无监督生成分区，再把它当成“已观测”的暴露变量做回归；K 由 HDBSCAN 自适应决定 ($\approx 10-15$)。	同左，完全无监督； min_topic_size=10 控制粒度。
情感利用方式	情感是模型内生变量：每主题有 sentiment-discourse 参数 α_s ，可直接输出“高/低情感下的关键词”。	情感当外生变量：① 主题概率 $\rightarrow \text{logit} \rightarrow$ 协变量（含 reviewer_score）做“主题暴露”回归；② 协变量 \rightarrow reviewer_score 做“情感”回归；③ PSM 用业务/休闲做处理变量。	情感仅用于后分析：把 reviewer_score 当 sent_score，取 10%-90% 分位点，分别算 c-TF-IDF，输出“负/中/正”关键词，但模型训练阶段完全不用。
关键词提取	同一主题在不同 α_s 分位点下计算 softmax(β) \rightarrow 三组词 (baseline/low/high)。	不产关键词；只把 BERTopic 现成的 topic-word 矩阵拿来做映射对比。	每主题分三档 sent_score 子集 \rightarrow c-TF-IDF \rightarrow Top-15 词；可输出“负/平均/正”三栏。
输出指标	① held-out LL；② semantic coherence；③ 主题-关键词 JSON；④ prevalence & sentiment 回归系数 Γ ；⑤ 代表性文档。	① topic_exposure 回归系数 (logit 尺度)；② sentiment 回归 R ² ；③ PSM 的 ATT、平衡表；④ 与 STS 系数方向一致性表。	① coherence_cv；② exclusivity；③ 负/中/正关键词 CSV；④ 与 STS 关键词 Jaccard/Cosine 映射表。
因果/推断功能	无直接因果，但系数可解释“评分每升 1σ ，主题流行度/情感 discourse 变化多少”。	显性因果框架：把 Business-vs-Leisure 当处理变量，用 PSM 估计 ATT (对 Reviewer_Score 的因果效应)。	无因果，仅描述性。
与 STS 关系	自身即 STS	把 STS 当“金标准”：读取其 prevalence/sentiment 系数，比较符号一致性；主题间用 Jaccard 做映射。	同左，也读 STS 关键词，做对齐和重叠度计算，但不做系数比较。
数据需求	需原始文本+评分+协变量；需要跑 Gibbs/EM，迭代 200 轮左右，耗时。	需 BERTopic 结果+评分+协变量；后续全是 OLS/Logit，分钟级。	仅需文本+评分（评分仅后分析用）；训练分钟级。

维度	1. 纯 STS (<code>sts_fit.R</code>)	2. BERTopic+STS (<code>run_hybrid_pipeline.py</code>)	3. 纯 BERTopic (<code>run_bertopic_experiment.py</code>)
可重复性	设随机种子 2025; R 包 <code>stm+sts</code> 固定。	Python 层 seed 42; <code>sklearn/statsmodels</code> 固定; 但 <code>HDBSCAN</code> 内部有随机性。	同左。
优缺点	优点: 统一概率框架, 可解释“情感-主题”交互; 缺点: 慢, 对长文本大数据不友好, 超参 K 需网格。	优点: 快, 能直接套因果设计 (PSM), 符号一致性可验证 STS 稳健性; 缺点: 两阶段误差累积, 主题质量依赖 BERTopic, 无法修正情感内生。	优点: 完全数据驱动, 无需预设 K, 语义聚类准; 缺点: 无概率框架, 情感与主题割裂, 难以做统计推断。

总结一句话:

1. 纯 STS: 把“主题”和“情感”放在同一概率图里一次估计, 重解释、重不确定性, 但计算重。
2. BERT+STS: 先让 BERT 快速分区, 再用经典回归“模仿”STS 的 prevalence/sentiment 系数, 并做因果推断; 是一套“轻量验证+因果增强”的混合方案。
3. 纯 BERT: 完全数据驱动地找主题和关键词, 用情感只做后分析展示, 不做任何系数或因果, 适合快速探索和可视化。