Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

CACKT: Sentiment-Conditioned Boundary Learning and Candidate Transfer for Targeted Sentiment Analysis

Jinghao Yan
Hubei University
Yutong Liu
Zhejiang University
junfeng shen
shenjunfengwindy@163.com
Hubei University

Research Article

Keywords: Targeted sentiment analysis, Aspect-based sentiment analysis, Span extraction, Contrastive learning, Candidate transfer

Posted Date: November 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11289899/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

CACKT: Sentiment-Conditioned Boundary Learning and Candidate Transfer for Targeted Sentiment Analysis

Jinghao Yan[1], Yutong Liu[2], Junfeng Shen[1*]

[1*]School of Artificial Intelligence, Hubei University, Wuhan, China.
[2]College of Materials Science and Engineering, Zhejiang University, Hangzhou, 310027, Zhejiang, P.R. China.

*Corresponding author(s). E-mail(s): shenjunfengwindy@163.com;

Abstract

This paper focuses on Targeted Sentiment Analysis (TSA), aiming to detect specific aspects from text and predict their corresponding sentiment polarities. Precise aspect boundary detection is pivotal to TSA, as erroneous candidate spans inevitably compromise downstream sentiment classification. Nevertheless, prevalent span-based approaches largely overlook sentiment signals during boundary generation, yielding boundary drift and aspect–sentiment misalignment. To alleviate this problem, we present CACKT, a candidate-optimized TSA framework. CACKT incorporates a Sentiment-Conditioned Boundary Extraction (SCBE) module that learns prototypical representations of sentiment polarities, allowing sentiment features to proactively steer the identification of span start and end positions. Building upon SCBE, a confidence-aware candidate transfer mechanism dynamically weights candidates by their reliability, suppressing noise from low-confidence spans. Furthermore, boundary-aware hard negative contrastive learning constructs informative negatives—including boundary shifts, competing aspects, and mismatched sentiments—to sharpen the model's discrimination against plausible errors. Extensive experiments show that CACKT refines candidate boundaries and delivers superior overall TSA performance.

Keywords: Targeted sentiment analysis, Aspect-based sentiment analysis, Span extraction, Contrastive learning, Candidate transfer

1 Introduction

Span-based joint models have improved targeted sentiment analysis (TSA), but extraction and sentiment prediction still use information differently. The sentiment classifier receives an aspect representation, whereas the boundary predictor often has no explicit polarity input. If the predicted span omits part of an aspect or includes unrelated tokens, assigning it the correct polarity does not repair the extraction error. CACKT brings polarity information into candidate generation so that it can influence boundaries before classification.

Sentiment expressions often help locate an opinion target. Opinion words, negation, and degree modifiers can indicate both the polarity of an opinion and the nearby span to which it applies. We use Sentiment-Conditioned Boundary Extraction (SCBE) to incorporate these cues into boundary prediction. SCBE learns one prototype per sentiment class and uses it to compute polarity-specific start and end scores. Candidate spans therefore carry boundary evidence and a source polarity before they reach the aspect-conditioned classifier.

Polarity-conditioned extraction introduces candidates of varying quality. Giving every predicted span the same training weight can reinforce a boundary error or an incorrect source polarity. Confidence-Aware Soft Candidate Transfer (CAST) instead ranks candidates within each sentence and uses their relative scores to weight supervision from the retained source labels.

The model must also distinguish a gold span from alternatives that differ by only a few tokens or have an incompatible polarity. Boundary-Aware Hard Negative Contrastive Learning (BA-HNCL) uses shifted endpoints, competing aspects, and mismatched sentiment prototypes as negatives. This objective updates the encoder states and prototypes that SCBE uses to generate candidates.

Our contributions are as follows:

1. SCBE uses trainable sentiment prototypes to compute polarity-specific start and end scores, allowing sentiment information to influence which spans enter the candidate pool.
2. CAST weights pseudo-label supervision using candidate scores and source polarities. BA-HNCL trains the model to distinguish gold spans from nearby boundary errors and incompatible polarities.
3. Boundary generation and contrastive training share the encoder states and sentiment prototypes. SCBE also supplies the candidate scores and source labels used by CAST, linking extraction and sentiment supervision through the same candidates.
4. CACKT achieves pair F1 scores of 0.7262, 0.8165, and 0.6474 on Laptop, Restaurant, and Tweets, respectively. Cumulative and component ablations examine sentiment-conditioned extraction, confidence-weighted transfer, prototype contrast, source priors, and pair verification.

The CACKT implementation is available at https://github.com/zhang-sas/cackt.

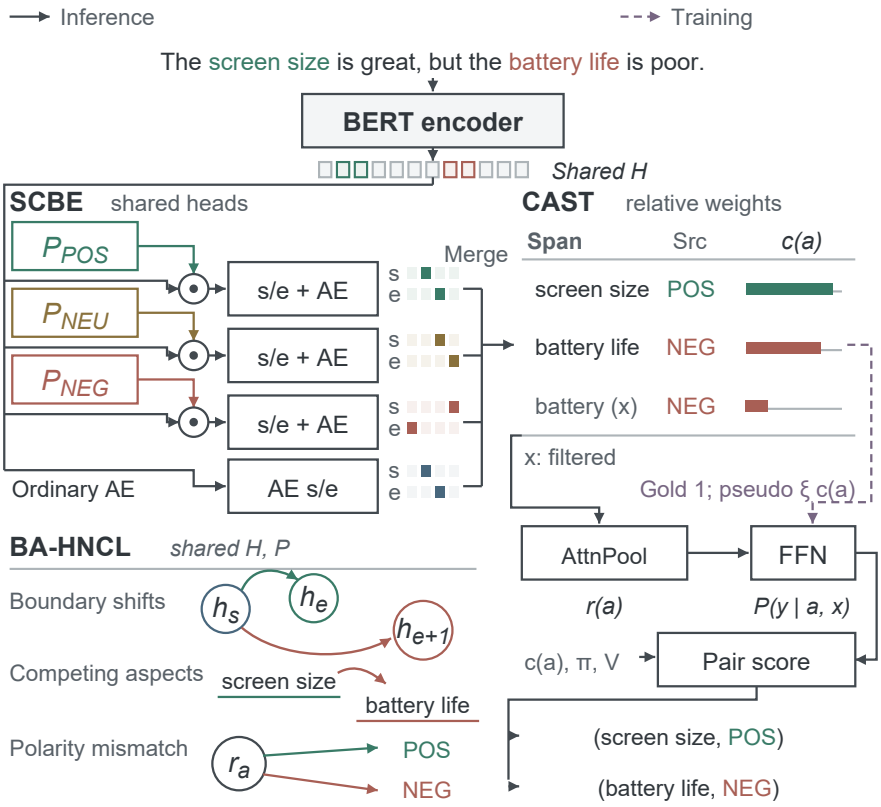


Fig. 1 CACKT architecture. SCBE combines prototype-gated boundary scores with ordinary AE logits. CAST uses sentence-relative candidate scores to weight sentiment supervision. Decoding uses the classifier distribution, candidate weight, source-polarity prior, and auxiliary verifier. The lower sketches illustrate BA-HNCL contrasts in the shared encoder–prototype space. Dashed purple paths denote training supervision. Spans, scores, and weight bars are illustrative.

2 Related Work

2.1 Targeted Sentiment Analysis and Polarity-Aware Extraction

Targeted sentiment analysis (TSA) identifies opinion targets and the sentiment expressed toward each target. Early pipeline systems extracted aspect terms first and then classified their polarity. This design is modular, but extraction errors pass to the classifier: polarity prediction cannot recover an omitted aspect or correct its boundaries. The SemEval shared tasks provided benchmark datasets and evaluation protocols for later end-to-end TSA research (Pontiki et al. 2014, 2015, 2016).

Pre-trained encoders allow aspect extraction (AE) and sentiment prediction (SP) to share contextual representations (Devlin et al. 2019). TSA models have used span-based extraction and classification, joint span prediction, hierarchical interaction, and

unified target–sentiment labeling (Hu et al. 2019; Zhou et al. 2019; Lv et al. 2021; Chen et al. 2022; Li et al. 2019). Related tasks, including aspect sentiment triplet extraction (ASTE) and aspect sentiment quad prediction (ASQP), predict structures that can also include opinion spans or aspect categories. Methods for these tasks include span interactions, dual-channel extraction, grid tagging, position-aware tagging, and diffusion-based grid denoising (Xu et al. 2021; Li et al. 2023; Wu et al. 2020; Sun et al. 2024; Xu et al. 2020; Zhu et al. 2024). Their output spaces differ from TSA, so their original results are not directly comparable with aspect–sentiment pair extraction. In many TSA models, sentiment prediction receives an aspect representation, but boundary prediction lacks an explicit sentiment hypothesis. Polarity information thus arrives too late to influence candidate generation.

Recent aspect-based sentiment analysis (ABSA) studies address both aspect-given classification and structured generation. When the target is given, methods improve its representation using syntactic plugins, recurrent or state-space architectures, graph alignment, hypergraphs, or geometric disentanglement (Tian et al. 2025; Lawan et al. 2025; Liao et al. 2025; Kashyap et al. 2026; Wang et al. 2026). Generative methods predict triplets or quadruples through element ordering, relation learning, draft correction, or multi-agent decomposition (Jun and Lee 2025; Lai et al. 2025; He et al. 2027; Hong et al. 2026). These studies focus on contextual representations or structured output generation. CACKT focuses on aspect-boundary recovery in discriminative TSA: SCBE uses polarity information when constructing the candidate pool.

2.2 Candidate Transfer and Span-Level Supervision

AE and SP can inform each other. Accurate boundaries identify the target to be classified, while opinion words, negation, and local discourse help identify plausible aspect spans. Coarse-grained transfer methods share this information through encoders, global attention, or task-level alignment (Chen et al. 2022, 2024; Lin and Yang 2020). Sentence-level or task-level signals, however, may not distinguish aspects with different polarities in the same sentence.

Fine-grained transfer links boundary evidence, span representations, and polarity decisions at the aspect or candidate level. FCKT combines fine-grained knowledge transfer with semantic contrastive learning and alternates between gold and predicted aspect inputs (Chen et al. 2025). Predicted candidates can therefore provide training inputs for sentiment classification. Their reliability remains a concern: a fixed threshold applies the same cutoff across sentences, and equal weights give uncertain candidates as much influence as reliable ones.

CAST uses the scores and source polarities of SCBE candidates. It ranks candidates within a sentence and converts their centered scores into continuous training weights. These weights express relative confidence within that sentence; they are not calibrated probabilities of a correct prediction.

2.3 Shared Prototypes and Boundary-Aware Contrastive Learning

Prototype learning uses class-level anchors to structure a representation space (Snell et al. 2017). In ABSA, sentiment-aware interactions and supervised contrastive objectives usually act on representations of given or extracted aspects (Chen et al. 2024; Liang et al. 2021; Xiong et al. 2022). Label-correction methods use sentiment feedback to refine predictions (Wang and Ren 2020), while span-based TSA methods explicitly predict aspect boundaries (Hu et al. 2019; Lv et al. 2021). SCBE combines polarity information with boundary prediction at the candidate-generation stage. Its trainable sentiment prototypes determine class-specific start and end scores and also serve as anchors for contrastive training. The same prototypes thus guide candidate selection and the representations used to distinguish candidate boundaries.

Supervised contrastive learning encourages class separation and has been applied to language-model fine-tuning and ABSA (Khosla et al. 2020; Gunel et al. 2021; Liang et al. 2021; Xiong et al. 2022). Hard-negative selection emphasizes examples that are difficult to distinguish from an anchor (Robinson et al. 2021). For TSA, FCKT contrasts aspect endpoints to align the start and end representations (Chen et al. 2025). Strict matching also requires distinguishing one-token boundary shifts, nearby competing aspects, and incorrect polarities. BA-HNCL targets these errors: endpoint contrast updates the encoder states used by the boundary heads, and span-to-prototype contrast updates the prototypes used in polarity-conditioned extraction.

3 Method

3.1 Overview

Given a WordPiece-tokenized review sentence $x = \{x_1, x_2, \ldots, x_n\}$, TSA predicts a set of aspect–sentiment pairs $Y = \{(a_m, y_m)\}_{m=1}^{M}$. Each aspect $a_m = (s_m, e_m)$ is defined by its start and end positions, and $y_m \in \mathcal{Y}_s = \{\text{neutral}, \text{positive}, \text{negative}\}$ is its polarity. A pre-trained BERT encoder maps x to contextual representations $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n\} \in \mathbb{R}^{n \times d}$.

Figure 1 shows the three main components:

(1) **Polarity-aware candidate generation (SCBE):** SCBE learns sentiment prototypes $\{\mathbf{P}_y\}_{y \in \mathcal{Y}_s}$ and computes polarity-specific start and end scores. After merging candidate sources, it returns each span, its boundary score, and its source polarity when available.

(2) **Candidate-weighted supervision (CAST):** CAST converts the sentence-centered candidate score into a weight $c(a) \in [0, 1]$. This weight controls the contribution of each SCBE pseudo-candidate to sentiment-classifier training.

(3) **Boundary-space regularization (BA-HNCL):** BA-HNCL contrasts gold spans with boundary shifts and polarity alternatives through three objectives: $s \to e$, $e \to s$, and span-to-prototype alignment. These objectives update the encoder states and prototypes $\mathbf{P}_y$ shared with SCBE.

Let Θ_{SCBE} denote the shared encoder, polarity prototypes, and sentiment-conditioned boundary parameters. SCBE returns

$$(\mathcal{A}_x,\, \mathbf{z}_x,\, \hat{\mathbf{y}}_b) = \mathrm{SCBE}(x; \Theta_{SCBE}), \tag{1}$$

where $\mathcal{A}_x$ is the merged candidate set, $\mathbf{z}_x$ contains the candidate scores, and $\hat{\mathbf{y}}_b$ contains source polarities where available. CAST uses these outputs, while BA-HNCL uses the shared encoder states $\mathbf{H}$ and prototypes $\mathbf{P} \subset \Theta_{SCBE}$:

$$\mathcal{L}_{CAST} = \mathcal{L}_{AC}^{soft}(\mathcal{A}_x, \mathbf{z}_x, \hat{\mathbf{y}}_b), \qquad \mathcal{L}_{BA} = \mathcal{L}_{HNCL}(\mathbf{H}, \mathbf{P}). \tag{2}$$

Equation 2 summarizes the training dependencies. CAST uses candidate scores and source polarities; BA-HNCL updates the encoder states and prototypes used by SCBE. An auxiliary verifier adds a pair-consistency score during decoding. Table 1 lists the inputs and role of each main component.

Table 1 Information flow and shared quantities in CACKT

Component	Inputs or shared quantities	Outputs	Function
SCBE	Encoder states and polarity prototypes $\mathbf{P}_y$	Candidate spans, scores, and source polarities	Predicts polarity-conditioned aspect boundaries
CAST	Candidate scores and source polarities	Sentence-relative transfer weights	Weights supervision from predicted candidates
BA-HNCL	Encoder states, gold spans, and polarity prototypes	Boundary- and prototype-level contrastive losses	Distinguishes gold spans from shifted boundaries and incorrect polarities

Note: CAST uses scores and source polarities from SCBE. BA-HNCL updates the encoder states and prototypes shared with SCBE. The auxiliary verifier adds a pair-consistency score during decoding.

3.2 Sentiment-Conditioned Boundary Extraction

A conventional AE head scores token i as an aspect start or end without conditioning on sentiment. SCBE instead computes a start and end score for each polarity $y \in \mathcal{Y}_s$. Polarity evidence can therefore affect candidate generation before aspect-conditioned classification. The resulting candidates support the objectives in Sections 3.3 and 3.4.

SCBE shares its boundary projections across polarity channels. A trainable sentiment prototype gates each contextual token representation, and a shared projection maps the gated features to start and end logits. These logits are added to the ordinary AE logits. The model retains the ordinary AE source for recall and adds three polarity-specific sources.

Each sentiment class has a trainable prototype vector:

$$\mathbf{P}_y \in \mathbb{R}^d, \quad y \in \{\text{neutral}, \text{positive}, \text{negative}\}. \tag{3}$$

The prototype is transformed into a bounded gate and applied element-wise to token i:

$$\mathbf{r}_y = \tanh(\mathbf{P}_y), \quad \tilde{\mathbf{h}}_i^y = \mathbf{h}_i \odot \mathbf{r}_y. \tag{4}$$

Here $\odot$ denotes element-wise multiplication. Although $\mathbf{h}_i$ is shared, the gated representation $\tilde{\mathbf{h}}_i^y$ depends on polarity. The conditioned projections are added to the ordinary AE logits $l_{s,i}$ and $l_{e,i}$:

$$l_{s,i}^y = \mathbf{W}_s^c \tilde{\mathbf{h}}_i^y + \mathbf{b}_s^c + l_{s,i}, \quad l_{e,i}^y = \mathbf{W}_e^c \tilde{\mathbf{h}}_i^y + \mathbf{b}_e^c + l_{e,i}. \tag{5}$$

The projection parameters $\mathbf{W}_s^c, \mathbf{W}_e^c \in \mathbb{R}^{1\times d}$ and biases $\mathbf{b}_s^c, \mathbf{b}_e^c \in \mathbb{R}$ are shared across polarity channels; $\mathbf{P}_y$ provides the class-specific information. Stacking the scores over classes and tokens gives $\mathbf{l}_s, \mathbf{l}_e \in \mathbb{R}^{C\times n}$, where $C = |\mathcal{Y}_s| = 3$.

A sentence can contain multiple aspects, so boundary prediction is a multi-label problem. Let $\mathbf{T}_s, \mathbf{T}_e \in \{0,1\}^{C\times n}$ be polarity-specific multi-hot targets, and let $M_i \in \{0,1\}$ indicate a valid, non-padding token. For logits $\mathbf{L}$ and targets $\mathbf{T}$, the masked binary cross-entropy is

$$\begin{aligned}\mathrm{MBCE}(\mathbf{L},\mathbf{T};\mathbf{M}) = -\frac{1}{C\sum_{i=1}^n M_i}\sum_{y=1}^{C}\sum_{i=1}^{n} M_i\big[T_{y,i}\log\sigma(L_{y,i}) \\ + (1-T_{y,i})\log\big(1-\sigma(L_{y,i})\big)\big].\end{aligned} \tag{6}$$

The SCBE loss is the average of the start and end losses:

$$\mathcal{L}_{SCBE} = \frac{1}{2}\big[\mathrm{MBCE}(\mathbf{l}_s,\mathbf{T}_s;\mathbf{M}) + \mathrm{MBCE}(\mathbf{l}_e,\mathbf{T}_e;\mathbf{M})\big]. \tag{7}$$

For a gold aspect (s, e, y), the start target at s and end target at e are positive in channel y. SCBE and the span-validity branch use only gold spans for training. Pseudo-candidates are used by the weighted sentiment-classification objective in Section 3.3. The boundary objective is

$$\mathcal{L}_{boundary} = \lambda_s\mathcal{L}_{start} + \lambda_e\mathcal{L}_{end} + \frac{\lambda_s+\lambda_e}{2}\mathcal{L}_{SCBE} + 0.1\mathcal{L}_{span}. \tag{8}$$

Here $\mathcal{L}_{start}$ and $\mathcal{L}_{end}$ train the ordinary AE heads, and $\mathcal{L}_{span}$ trains the span-validity branch. We set $\lambda_s = \lambda_e = 1$ and weight span validity by 0.1.

During inference, candidates from the ordinary AE source and the three polarity-specific sources are merged. If the highest-scoring source for candidate a is polarity-conditioned, its label $\hat{y}_b$ supplies a soft prior for the final pair score:

$$\pi(a, y) = \eta \cdot \mathbb{I}(y = \hat{y}_b) \cdot c_b(a). \tag{9}$$

Here $c_b(a) \in [0, 1]$ is the source-specific boundary confidence, and η controls the prior weight. The prior in Equation 9 supplements the aspect-conditioned prediction $P(y \mid a, x)$. Setting $\eta = 0$ removes the source-polarity prior.

3.3 Confidence-Aware Soft Candidate Transfer

CAST weights supervision from the candidate pool produced by SCBE. It uses the candidates, scores, and source polarities in Equation 1 to select candidates relative to others in the same sentence and assign continuous training weights.

Let $\mathcal{Q} = \{AE\} \cup \mathcal{Y}_s$ contain the ordinary AE source and the three polarity-specific SCBE sources. For a span $a = (s, e)$ generated by source $q \in \mathcal{Q}$, the source score is

$$z_q(a) = l_s^q(s) + l_e^q(e) + v(a) - \lambda_{len}|a|. \tag{10}$$

Here $v(a) \in [0, 1]$ is the span-validity score, $|a| = e - s + 1$ is the span length, and λ_{len} penalizes long spans. We set $v(a) = 0$ when the validity branch is unavailable and use $\lambda_{len} = 1$ when heuristic length ranking is enabled. When multiple sources produce the same span, we retain the maximum score:

$$z(a) = \max_{q \in \mathcal{Q}(a)} z_q(a), \tag{11}$$

The set $\mathcal{Q}(a)$ contains the sources that generate a. If the highest-scoring source is polarity-conditioned, its polarity is retained as $\hat{y}_b$ for the prior in Equation 9.

Candidate-score ranges can vary with sentence length and the number of aspects, making a fixed cutoff difficult to apply across sentences. CAST defines a reference score for each sentence. Given candidates $\mathcal{A}_x$ and scores $\mathcal{Z}_x = \{z(a) : a \in \mathcal{A}_x\}$, the reference is

$$\tau_x = \mu(\mathcal{Z}_x) + \rho \cdot \sigma(\mathcal{Z}_x). \tag{12}$$

Here $\mu(\mathcal{Z}_x)$ and $\sigma(\mathcal{Z}_x)$ are the within-sentence mean and standard deviation, and ρ controls the cutoff. We use $\rho = 0$, making τ_x the sentence mean. Positive values of ρ raise the threshold τ_x by a fraction of the standard deviation. Candidate retention also uses a minimum soft weight, top-K selection, and a minimum-retention rule to avoid an empty pool.

The centered score is converted to a continuous weight:

$$c(a) = \text{sigmoid}\big((z(a) - \tau_x)/T\big). \tag{13}$$

The temperature $T > 0$ controls the smoothness of this mapping. The weight $c(a)$ measures the score of candidate a relative to scores in the same sentence. It is not a globally calibrated probability that the candidate is correct.

Gold spans receive full weight and their gold sentiment labels. A predicted candidate that exactly matches a gold span also receives the gold label. An unmatched SCBE candidate uses its source polarity as a weak label, weighted by $c(a)$. An unmatched candidate from only the ordinary AE source has no source polarity and is excluded from the sentiment loss; it is not assigned the label *other*. To limit noisy supervision early in training, the pseudo-candidate branch starts after t_0 warm-up steps and is selected with probability $1 - p_g$.

For candidate $a = (s, e)$, attention pooling gives $\mathbf{r}(a) = \text{AttnPool}(\mathbf{H}_{s:e})$. A nonlinear projection and linear classifier produce logits $\mathbf{o}(a)$, with $P(y \mid a, x) = \text{softmax}(\mathbf{o}(a))_y$. Let y_a be the gold or weak sentiment label assigned under the rules above. The weighted classification loss is

$$\mathcal{L}_{AC}^{soft} = -\frac{\sum_{a \in \mathcal{A}_x} w(a) \log P(y_a \mid a, x)}{\sum_{a \in \mathcal{A}_x} w(a) + \varepsilon}. \tag{14}$$

$$w(a) = m(a) \cdot \begin{cases} 1, & a \text{ is gold;} \\ \xi \cdot c(a), & a \text{ is pseudo;} \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Here $m(a)$ masks invalid or unsupervised candidates, $\xi \in [0, 1]$ caps the contribution of pseudo-spans, and ε prevents division by zero. Normalization by the total active weight keeps the number of retained candidates from directly rescaling the loss.

Gold spans therefore retain full weight, while SCBE pseudo-spans contribute according to their relative confidence. Invalid, unretained, or unlabeled candidates contribute nothing. Section 4.5 tests the effect of removing source-polarity weak labels.

3.4 Boundary-Aware Hard Negative Contrastive Learning

BA-HNCL trains the representations used by SCBE to distinguish gold spans from nearby errors. Random negatives may be too dissimilar to a gold aspect to inform these local boundary decisions. We construct deterministic negatives from shifted start or end positions, boundaries of other aspects in the same sentence, and prototypes with the wrong polarity. The contextual state $\mathbf{h}_i$ is shared with the SCBE boundary heads, and $\mathbf{P}_y$ is the prototype in Equation 3.

These negatives support three contrastive objectives. The first two model compatibility between start and end positions in both directions. The third aligns a span with its sentiment prototype. Each objective uses cosine similarity:

$$\mathrm{sim}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\|_2 \|\mathbf{v}\|_2}, \tag{16}$$

The shared contrastive temperature is denoted by $\tau_{\mathrm{cl}} > 0$.

For a gold span $a = (s, e, y)$, the start-to-end objective uses $\mathbf{h}_s$ as the anchor and $\mathbf{h}_e$ as the positive. The negative set $\mathcal{N}_e(s)$ includes valid local shifts of e and end positions of other gold aspects in the sentence:

$$\mathcal{L}_{s \to e}(a) = -\log \frac{\exp(\mathrm{sim}(\mathbf{h}_s, \mathbf{h}_e)/\tau_{\mathrm{cl}})}{\exp(\mathrm{sim}(\mathbf{h}_s, \mathbf{h}_e)/\tau_{\mathrm{cl}}) + \sum_{j \in \mathcal{N}_e(s)} \exp(\mathrm{sim}(\mathbf{h}_s, \mathbf{h}_j)/\tau_{\mathrm{cl}})}. \tag{17}$$

The end-to-start objective reverses the endpoint roles. Its negative set $\mathcal{N}_s(e)$ contains valid local shifts of s and start positions of other aspects:

$$\mathcal{L}_{e \to s}(a) = -\log \frac{\exp(\mathrm{sim}(\mathbf{h}_e, \mathbf{h}_s)/\tau_{\mathrm{cl}})}{\exp(\mathrm{sim}(\mathbf{h}_e, \mathbf{h}_s)/\tau_{\mathrm{cl}}) + \sum_{j \in \mathcal{N}_s(e)} \exp(\mathrm{sim}(\mathbf{h}_e, \mathbf{h}_j)/\tau_{\mathrm{cl}})}. \tag{18}$$

Span-to-prototype contrast uses the mean of the endpoint representations, keeping the three objectives in the same boundary representation space:

$$\mathbf{r}_a = \frac{\mathbf{h}_s + \mathbf{h}_e}{2}. \tag{19}$$

The gold polarity prototype $\mathbf{P}_y$ is the positive; the other prototypes are negatives:

$$\mathcal{L}_{sp}(a) = -\log \frac{\exp(\mathrm{sim}(\mathbf{r}_a, \mathbf{P}_y)/\tau_{\mathrm{cl}})}{\sum_{k \in \mathcal{Y}_s} \exp(\mathrm{sim}(\mathbf{r}_a, \mathbf{P}_k)/\tau_{\mathrm{cl}})}. \tag{20}$$

Near sentence boundaries, some spans have no valid perturbation in one direction. Let $\mathcal{V}_x$ contain the contrastive terms that are defined for sentence x. The sentence-level loss is

$$\mathcal{L}_{HNCL}(x) = \begin{cases} \dfrac{1}{|\mathcal{V}_x|} \displaystyle\sum_{\ell \in \mathcal{V}_x} \ell, & |\mathcal{V}_x| > 0, \\ 0, & |\mathcal{V}_x| = 0. \end{cases} \tag{21}$$

Undefined directions are skipped, and the loss is averaged over the remaining terms. Gradients from Equations 17–21 update the shared encoder and $\mathbf{P}_y$, which determine the SCBE logits in Equations 4–5. The endpoint-only ablation in Section 4.5 retains both endpoint objectives and removes span-to-prototype contrast.

3.5 Training Objective

The training objective combines boundary prediction, weighted sentiment classification, hard-negative contrastive learning, and pair verification:

$$\mathcal{L} = \mathcal{L}_{boundary} + \lambda_{ac}\mathcal{L}_{AC}^{soft} + \lambda_{cl}\mathcal{L}_{HNCL} + \lambda_{pv}\mathcal{L}_{PV}. \tag{22}$$

The boundary loss $\mathcal{L}_{boundary}$ trains ordinary and polarity-specific extraction and span validity. The classification loss $\mathcal{L}_{AC}^{soft}$ uses the SCBE outputs $(\mathcal{A}_x, \mathbf{z}_x, \hat{\mathbf{y}}_b)$. The contrastive loss $\mathcal{L}_{HNCL}$ updates the shared encoder and prototypes, and $\mathcal{L}_{PV}$ trains the pair verifier. Their weights are λ_{ac}, λ_{cl}, and λ_{pv}, respectively. The verifier ablation sets $\lambda_{pv} = 0$. Hyperparameters are selected on validation data and kept fixed for test evaluation.

During inference, candidates from ordinary AE and polarity-specific sources are merged. For each candidate a and non-*other* sentiment y, the pair score is

$$\begin{aligned} S(a, y) = {} & \log\big(c(a) + \varepsilon\big) + \log \frac{P(y \mid a, x) + \varepsilon}{P(other \mid a, x) + \varepsilon} \\ & + \pi(a, y) + \gamma_{pv} \log\big(V(a, y) + \varepsilon\big), \end{aligned} \tag{23}$$

Here $V(a, y)$ is the auxiliary verifier probability, and γ_{pv} controls its decoding contribution. The verifier term is omitted when pair verification is disabled. Pairs below the non-*other* confidence bound are discarded. The remaining pairs are ranked by $S(a, y)$, deduplicated, filtered for overlap, and limited to top-K.

3.6 Auxiliary Pair Verification

The auxiliary verifier estimates whether a candidate span and polarity should be retained as a pair. It is trained on gold pairs and incorrect span–polarity combinations. Its score is then used in decoding.

Given a span representation $\mathbf{r}(a)$ and polarity prototype $\mathbf{P}_y$, a two-layer feed-forward network with hidden size $d/2$ and GELU activation computes

$$V(a, y) = \sigma\big(\mathrm{FFN}_{PV}([\mathbf{r}(a); \mathbf{P}_y])\big), \tag{24}$$

where $[\cdot;\cdot]$ denotes concatenation. Let $\mathcal{R}_x$ be the verifier training pairs for sentence x, with labels $t_{a,y} \in \{0, 1\}$. Gold aspect–sentiment pairs are positive examples; pairs containing a non-gold span or an incorrect polarity are negative examples. The verifier loss is

$$\mathcal{L}_{PV}(x) = -\frac{1}{|\mathcal{R}_x|} \sum_{(a,y)\in\mathcal{R}_x} \big[t_{a,y} \log V(a, y) + (1 - t_{a,y}) \log\big(1 - V(a, y)\big)\big]. \tag{25}$$

The verifier contributes to the training loss in Equation 22 and the decoding score in Equation 23. Its effect is evaluated separately in the component ablations.

4 Experiments

4.1 Datasets and Evaluation Metrics

We evaluate CACKT on three public TSA datasets: Laptop, Restaurant, and Tweets. Laptop comes from SemEval-2014; Restaurant combines restaurant reviews from SemEval-2014, SemEval-2015, and SemEval-2016; and Tweets comes from the Open Domain Targeted Sentiment corpus (Pontiki et al. 2014, 2015, 2016; Mitchell et al. 2013). Table 2 reports dataset statistics. Laptop and Restaurant use the established training/test splits. Tweets uses the ten-fold cross-validation protocol adopted by the compared TSA systems.

Table 2 Dataset statistics

Dataset	#Sentences	#Aspects	#Positive	#Negative	#Neutral
Laptop	1869	2936	1326	990	620
Restaurant	3900	6603	4134	1538	931
Tweets	2350	3243	703	274	2266

Note: #Sentences and #Aspects count sentences and aspect terms. #Positive, #Negative, and #Neutral count aspects in each sentiment class. Laptop and Restaurant use public data splits; Tweets uses 10-fold cross-validation.

We report precision, recall, and F1 for aspect–sentiment pair extraction (P_{all}, R_{all}, and $F1_{all}$) and aspect extraction (P_{ae}, R_{ae}, and $F1_{ae}$), along with sentiment prediction accuracy (Acc_{sp}). Pair evaluation uses strict matching: both aspect boundaries and polarity must be correct. CACKT and its controlled variants are summarized over five matched random seeds, with means and standard deviations reported in the ablation tables. The individual paired results needed for significance tests are unavailable. We therefore report descriptive comparisons without significance markers.

4.2 Implementation Details

We use BERT-large-uncased with a maximum input length of 85 tokens and a maximum candidate span length of 12. Training and prediction batch sizes are both 32. BERTAdam trains the model for 100 epochs with an initial learning rate of 2×10^{-5} and a warm-up proportion of 0.1. Each dataset is evaluated with five random seeds; Tweets uses ten-fold cross-validation for each seed.

Candidate selection uses the sentence-relative threshold with $\rho = 0$, temperature $T = 1$, and a minimum confidence of 0.55. At least one candidate is retained. The source-polarity prior weight is $\eta = 0.25$. CAST starts after 200 update steps, with gold-branch probability $p_g = 0.90$ and pseudo-candidate weight $\xi = 0.30$. The BA-HNCL temperature is $\tau_{cl} = 0.07$.

The objective weights are $\lambda_{ac} = 1.0$, $\lambda_{cl} = 0.10$, and $\lambda_{pv} = 0.20$, selected on validation data to balance classification, contrastive learning, and pair verification. Decoding uses a non-*other* sentiment confidence bound of 0.45 and an overlap-suppression threshold of 0.50.

4.3 Main Results

Table 3 compares CACKT with pipeline, span-based, hierarchical, dual-channel, grid-tagging, and cross-task transfer methods. CACKT achieves the highest pair F1 among the compared systems: 0.7262 on Laptop, 0.8165 on Restaurant, and 0.6474 on Tweets. These scores exceed the strongest listed baseline by 1.18, 0.12, and 2.49 percentage points, respectively. The gain is small on Restaurant and larger on Laptop and Tweets, suggesting that the benefit varies by domain.

Restaurant includes recurring aspect nouns such as *food*, *service*, and *price*, which existing span-based models can already recover well. Laptop contains more technical and multi-word aspect expressions, while Tweets includes informal and abbreviated language. These differences may help explain the larger gains on Laptop and Tweets, where exact boundary recovery can be more difficult. The aggregate results, however, do not establish the cause of the domain differences.

Candidate reliability also affects how useful predicted spans are for training. CAST assigns sentence-relative weights so that uncertain candidates contribute less to the sentiment loss. The ablations below examine the contribution of candidate transfer and its source-polarity labels.

Table 4 reports results for the two subtasks: aspect extraction, measured by $F1_{ae}$, and sentiment prediction, measured by accuracy Acc_{sp}.

Table 3 Aspect–sentiment pair extraction results

Model	Laptop			Restaurant			Tweets		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
TAG-Pipeline† (Hu et al. 2019)	0.6584	0.6719	0.6651	0.7166	0.7645	0.7398	0.5424	0.5437	0.5426
SPAN-Pipeline† (Hu et al. 2019)	0.6946	0.6672	0.6806	0.7614	0.7334	0.7492	0.6072	0.5502	0.5769
SPJM† (Zhou et al. 2019)	0.6140	0.5820	0.5976	0.7620	0.6820	0.7198	0.5484	0.4844	0.5144
SPAN-Joint† (Hu et al. 2019)	0.6741	0.6199	0.6459	0.7232	0.7261	0.7247	0.5703	0.5269	0.5455
S-AESC† (Lv et al. 2021)	0.6687	0.6492	0.6588	0.7826	0.7050	0.7418	0.5586	0.5374	0.5473
HI-ASA‡ (Chen et al. 2022)	0.6796	0.6625	0.6709	0.7915	0.7621	0.7765	0.5732	0.5622	0.5676
DCS‡ (Li et al. 2023)	0.6812	0.6640	0.6725	0.7835	0.7751	0.7793	0.5862	0.5835	0.5848
MiniConGTS‡ (Sun et al. 2024)	0.7206	0.6725	0.6957	0.7926	0.7952	0.7939	0.6164	0.5728	0.5938
PDGN‡ (Zhu et al. 2024)	0.7025	0.6812	0.6921	0.8036	0.7985	0.8010	0.6235	0.5924	0.6076
AIFI‡ (Chen et al. 2024)	0.7105	0.6915	0.7009	0.7925	0.8034	0.7979	0.6342	0.5911	0.6119
FCKT‡ (Chen et al. 2025)	**0.7599**	0.6740	0.7144	**0.8449**	0.7877	0.8153	0.6512	0.5962	0.6225
CACKT§	0.7221	**0.7303**	**0.7262**	0.8071	**0.8319**	**0.8165**	**0.6698**	**0.6265**	**0.6474**

Note: †, ‡, and § denote results from the original publications, reproduced results, and our results, respectively. Bold marks the best value in each column. DCS, MiniConGTS, and PDGN were developed for richer structured extraction tasks; the reproduced values shown here use the aspect–sentiment pair protocol. CACKT scores are means over five runs. Significance markers are omitted because paired per-seed results are not available for all baselines.

CACKT has the highest $F1_{ae}$ among the compared systems, with scores of 0.8856, 0.8802, and 0.7964 on Laptop, Restaurant, and Tweets, respectively. These results are consistent with the benefit of polarity-conditioned boundary prediction. They do not identify which types of boundary errors account for the gains.

For candidate-conditioned SP, CACKT achieves accuracies of 0.8480, 0.9298, and 0.8395. These are 1.32, 0.12, and 1.29 percentage points below FCKT on Laptop, Restaurant, and Tweets, respectively. Each model is evaluated on its own predicted candidates, so the scores depend on both classifier performance and the difficulty of those candidates. They cannot isolate the quality of the polarity classifiers.

CACKT nevertheless achieves higher pair F1 in Table 3. The joint results are consistent with stronger extraction outweighing the lower candidate-conditioned SP accuracy on Laptop and Tweets.

Table 4 Aspect extraction and candidate-conditioned sentiment prediction

Task	Method	Laptop	Restaurant	Tweets
AE	HI-ASA‡ (Chen et al. 2022)	0.8424	0.8511	0.7546
	DCS‡ (Li et al. 2023)	0.8455	0.8462	0.7543
	MiniConGTS‡ (Sun et al. 2024)	0.8374	0.8432	0.7514
	PDGN‡ (Zhu et al. 2024)	0.8485	0.8546	0.7593
	AIFI‡ (Chen et al. 2024)	0.8511	0.8631	0.7634
	FCKT‡ (Chen et al. 2025)	0.8534	0.8685	0.7754
	CACKT§	**0.8856**	**0.8802**	**0.7964**
SP	HI-ASA‡ (Chen et al. 2022)	0.8531	0.9257	0.8451
	DCS‡ (Li et al. 2023)	0.8467	0.9212	0.8429
	MiniConGTS‡ (Sun et al. 2024)	0.8541	0.9194	0.8349
	PDGN‡ (Zhu et al. 2024)	0.8519	0.9224	0.8496
	AIFI‡ (Chen et al. 2024)	0.8594	0.9305	0.8496
	FCKT‡ (Chen et al. 2025)	**0.8612**	**0.9310**	**0.8524**
	CACKT§	0.8480	0.9298	0.8395

Note: AE is measured by F1. SP accuracy is measured on each model's end-to-end candidate output and depends on both candidate difficulty and polarity prediction. ‡ and § denote reproduced results and our results, respectively. Bold marks the best value for each task and dataset.

4.4 Cumulative Ablation Study

Table 5 adds SCBE, CAST, BA-HNCL, and pair verification to Base in that order. SCBE raises F1 by 1.24, 1.17, and 1.70 percentage points on Laptop, Restaurant, and Tweets, respectively, averaging 1.37 points. CAST adds 0.60, 0.87, and 0.83 points, for an average of 0.77. BA-HNCL adds 0.40, 0.70, and 1.67 points, with the largest gain on Tweets. Pair verification adds 0.14, 0.23, and 0.39 points. These are differences between reported means; the available summary statistics do not establish statistical significance.

Table 5 Cumulative addition of SCBE, CAST, BA-HNCL, and pair verification (mean ± standard deviation)

Configuration	Laptop	Restaurant	Tweets	Mean step Δ (pp)
Base	0.7024 ± 0.0044	0.7868 ± 0.0030	0.6015 ± 0.0058	–
+ SCBE	0.7148 ± 0.0039	0.7985 ± 0.0026	0.6185 ± 0.0051	+1.37
+ CAST	0.7208 ± 0.0036	0.8072 ± 0.0024	0.6268 ± 0.0047	+0.77
+ BA-HNCL	0.7248 ± 0.0033	0.8142 ± 0.0022	0.6435 ± 0.0044	+0.92
Full (+ pair verifier)	**0.7262 ± 0.0031**	**0.8165 ± 0.0020**	**0.6474 ± 0.0042**	**+0.25**

Note: Each row retains the mechanisms added in earlier rows. Mean step Δ is the mean F1 change from the preceding configuration across datasets, expressed in percentage points. Significance markers are omitted because the required paired per-seed results are unavailable. The + BA-HNCL configuration is Full without pair verification.

The mean F1 across datasets increases from 0.6969 for Base to 0.7300 for Full. SCBE produces the largest average gain in this sequence. CAST adds a smaller gain on all three datasets, BA-HNCL is most helpful on Tweets, and pair verification adds the smallest final increment. Each step measures the effect of an addition conditional on the mechanisms already present.

4.5 Fine-Grained Component Analysis

Table 6 removes individual sources of information from Full. Removing source-polarity weak labels from CAST lowers mean F1 by 0.68 percentage points. Removing span-to-prototype contrast while retaining endpoint contrast lowers it by 0.73 points, suggesting a benefit beyond endpoint alignment. Disabling the source prior with $\eta = 0$ lowers mean F1 by 0.38 points. Removing pair verification yields the smallest decrease, 0.25 points.

Table 6 Fine-grained variants of candidate transfer, contrastive training, and decoding

Configuration	Laptop	Restaurant	Tweets	Mean Δ vs. Full (pp)
Full (CACKT)	**0.7262 ± 0.0031**	**0.8165 ± 0.0020**	**0.6474 ± 0.0042**	–
CAST w/o source-polarity labels	0.7220 ± 0.0037	0.8122 ± 0.0024	0.6356 ± 0.0048	-0.68
Endpoint-only HNCL	0.7230 ± 0.0038	0.8115 ± 0.0023	0.6338 ± 0.0047	-0.73
w/o source prior ($\eta = 0$)	0.7234 ± 0.0034	0.8142 ± 0.0021	0.6410 ± 0.0045	-0.38
w/o pair verifier ($\lambda_{pv} = 0$)	0.7248 ± 0.0033	0.8142 ± 0.0022	0.6435 ± 0.0044	-0.25

Note: Values are means ± standard deviations over five matched seeds. "CAST w/o source-polarity labels" retains candidate weighting and removes source-polarity weak labels. Endpoint-only HNCL removes span-to-prototype contrast. Mean Δ is the mean F1 difference from Full across datasets, expressed in percentage points.

4.6 Illustrative Error Analysis

Table 7 illustrates boundary truncation, boundary expansion, polarity mismatch across aspects, and reported or implicit stance. The examples describe how the proposed mechanisms are intended to address local boundary and pairing errors. They also highlight a limitation: CACKT does not explicitly model opinion holders or sentiment scope.

Table 7 Illustrative error patterns, relevant mechanisms, and limitations

Type	Example review	Possible error	Relevant mechanism	Intended effect or limitation
Boundary truncation	The screen size is satisfactory but the phone battery capacity is limited.	[phone battery], omitting *capacity*	SCBE allows the negative polarity cue *limited* to affect boundaries; BA-HNCL uses shifted boundaries as negatives	Favors the complete span [phone battery capacity]
Boundary expansion	You will obtain a gift if you buy the separate ram memory.	[separate ram memory], including the modifier *separate*	CAST weights supervision from the expanded candidate by its sentence-relative score	Reduces the influence of an expanded span if its relative score is low
Multi-aspect polarity mismatch	The food is good, but the service is not good.	Assigning positive polarity to *service*	SCBE conditions boundaries on polarity; span-to-prototype contrast separates incompatible polarities	Favors [food, positive] and [service, negative]
Reported or implicit stance	I am surprised by the bad reviews of the restaurant, though the menu's font is small.	Treating others' negative reviews as the speaker's view of the restaurant	CACKT models boundaries and local polarity without explicit opinion-holder or scope representations	May still misattribute sentiment despite recovering the aspect

Note: Brackets indicate illustrative aspect spans or aspect–polarity pairs. These examples explain possible errors and the intended role of each mechanism; they are not measured predictions or an independent performance comparison.

Figure 2 illustrates a multi-word aspect whose final token, *capacity*, is omitted by a possible sentiment-agnostic prediction. SCBE lets the negative cue *limited* influence the boundary scores, potentially favoring *phone battery capacity*. CAST then weights supervision from that candidate, and BA-HNCL contrasts the gold endpoints with shifted alternatives. The figure explains the intended behavior rather than reporting a measured prediction.

Correct extraction does not guarantee a correct interpretation of sentiment. In reported opinions, concessive clauses, or implicit sentiment, CACKT may recover the aspect but attribute the stance to the wrong opinion holder or scope. Addressing these errors requires representations of who expresses the opinion and which part of the sentence it applies to.

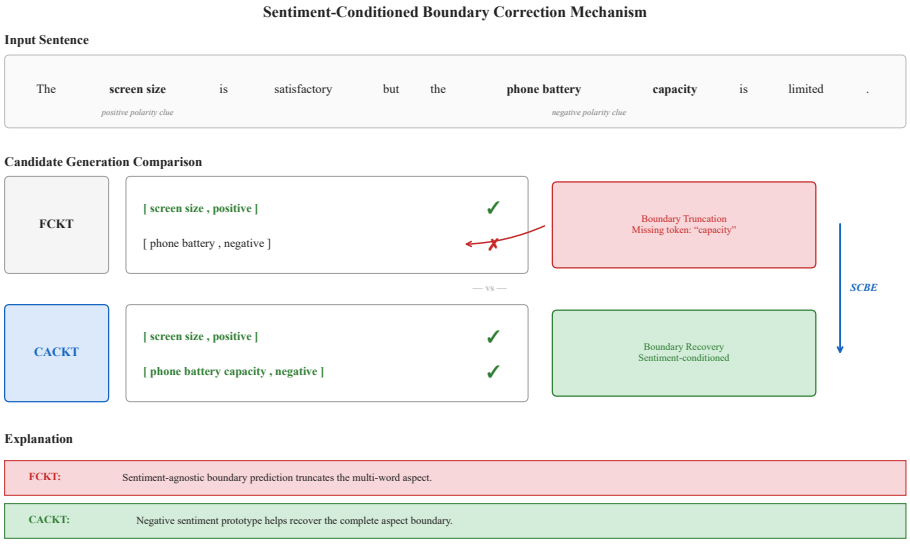


Fig. 2 Illustration of sentiment-conditioned boundary correction. The upper (FCKT) row shows a possible truncated span, while the lower (CACKT) row shows the complete aspect span that SCBE is intended to favor. The example is illustrative and does not report a measured prediction

External knowledge or large language models could help with implicit sentiment, pragmatic relations, and cross-sentence reasoning. Future work should assess whether these additions improve stance interpretation, alongside their computational cost and the reliability of the information they supply.

5 Conclusion

We presented CACKT for targeted sentiment analysis. SCBE uses sentiment prototypes to condition aspect-boundary prediction on polarity. CAST weights supervision from predicted candidates relative to other candidates in the sentence, and BA-HNCL distinguishes gold spans from shifted boundaries and incompatible polarity prototypes. An auxiliary verifier adds pair-consistency information during decoding.

CACKT achieves pair F1 scores of 0.7262, 0.8165, and 0.6474 on Laptop, Restaurant, and Tweets, respectively. In the cumulative ablation, SCBE, CAST, BA-HNCL, and pair verification add average gains of 1.37, 0.77, 0.92, and 0.25 percentage points. These comparisons are descriptive because paired significance tests cannot be verified from the available aggregate results. Removing source-polarity weak labels or span-to-prototype contrast has a larger effect than removing the source prior or pair verifier. CACKT remains limited in interpreting reported stance, concessive relations, and implicit opinion holders.

Statements and Declarations

Author Contributions

Jinghao Yan and Yutong Liu jointly contributed to method development, experiments, data analysis, and manuscript preparation. Junfeng Shen supervised the work.

Funding

No funding was received for the preparation of this manuscript.

Competing Interests

The authors declare no competing interests.

Data Availability

The Laptop and Restaurant datasets come from the SemEval-2014, SemEval-2015, and SemEval-2016 shared tasks (Pontiki et al. 2014, 2015, 2016). Tweets comes from the Open Domain Targeted Sentiment corpus (Mitchell et al. 2013). All datasets are publicly available from their original sources.

Code Availability

The CACKT source code is available at https://github.com/zhang-sas/cackt. The benchmark datasets are available from their original public repositories.

Ethical Approval

This study uses public benchmark datasets under their original distribution conditions. No additional personal data were collected, and ethical approval was not required.

Use of AI Tools

AI tools assisted with the layout of Figure 1. The framework diagram was drawn as an editable vector graphic using ReportLab and SVG. AI tools also assisted with English editing to improve clarity and style. This assistance did not generate experimental data or results. The diagrams are illustrative, and the authors retain responsibility for the scientific content.

References

Chen W, Du J, Zhang Z, et al (2022) A hierarchical interactive network for joint span-based aspect-sentiment analysis. In: Proceedings of the 29th International Conference on Computational Linguistics (COLING), https://doi.org/10.18653/v1/2022.coling-1.611

Chen W, Liu Y, Zhang Z, et al (2024) Modeling adaptive inter-task feature interactions via sentiment-aware contrastive learning for joint aspect-sentiment prediction. In:

Proceedings of the AAAI Conference on Artificial Intelligence, https://doi.org/10.1609/aaai.v38i16.29731

Chen W, Zhang Z, Yuan M, et al (2025) FCKT: Fine-grained cross-task knowledge transfer with semantic contrastive learning for targeted sentiment analysis. In: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI), pp 2731–2739, https://doi.org/10.24963/ijcai.2025/304

Devlin J, Chang MW, Lee K, et al (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), pp 4171–4186, https://doi.org/10.18653/v1/N19-1423

Gunel B, Du J, Conneau A, et al (2021) Supervised contrastive learning for pre-trained language model fine-tuning. In: Proceedings of the 9th International Conference on Learning Representations (ICLR), https://doi.org/10.48550/arXiv.2011.01403

He S, Wang H, Luo W (2027) Generate then correct: Single-shot global correction for aspect sentiment quad prediction. In: Huang DS, Zhang C, Chen W, et al (eds) Advanced Intelligent Computing Technology and Applications: ICIC 2026, Lecture Notes in Computer Science, vol 16663. Springer, Singapore, pp 203–213, https://doi.org/10.1007/978-981-92-3417-2_18, URL https://link.springer.com/chapter/10.1007/978-981-92-3417-2_18

Hong A, Wang L, Yue Z, et al (2026) MASTE: A multi-agent pipeline for zero-shot aspect sentiment triplet extraction. arXiv preprint https://doi.org/10.48550/arXiv.2607.08080, arXiv:2607.08080

Hu M, Peng Y, Huang Z, et al (2019) Open-domain targeted sentiment analysis via span-based extraction and classification. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), pp 537–546, https://doi.org/10.18653/v1/P19-1051

Jun Y, Lee H (2025) Dynamic order template prediction for generative aspect-based sentiment analysis. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp 614–626, https://doi.org/10.18653/v1/2025.acl-short.48, URL https://aclanthology.org/2025.acl-short.48/

Kashyap OM, Amit P, Kashyap M, et al (2026) From graphs to hypergraphs: Enhancing aspect-term sentiment analysis via multi-level relational modeling. In: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), pp 595–608, https://doi.org/10.18653/v1/2026.acl-srw.53, URL https://aclanthology.org/2026.acl-srw.53/

Khosla P, Teterwak P, Wang C, et al (2020) Supervised contrastive learning. In: Advances in Neural Information Processing Systems (NeurIPS), https://doi.org/10.5555/3495724.3497023

Lai W, Xie H, Xu G, et al (2025) STAR: Stepwise task augmentation with relation learning for aspect sentiment quad prediction. arXiv preprint https://doi.org/10.48550/arXiv.2501.16093, arXiv:2501.16093

Lawan A, Pu J, Yunusa H, et al (2025) AF-MAT: Aspect-aware flip-and-fuse xLSTM for aspect-based sentiment analysis. arXiv preprint https://doi.org/10.48550/arXiv.2507.01213, arXiv:2507.01213

Li P, Li P, Zhang K (2023) Dual-channel span for aspect sentiment triplet extraction. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), https://doi.org/10.18653/v1/2023.emnlp-main.17

Li X, Bing L, Li P, et al (2019) A unified model for opinion target extraction and target sentiment prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, https://doi.org/10.1609/aaai.v33i01.33016714

Liang B, Luo W, Li X, et al (2021) Enhancing aspect-based sentiment analysis with supervised contrastive learning. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM), https://doi.org/10.1145/3459637.3482096

Liao X, Chen X, Wang L, et al (2025) OTESGN: Optimal transport-enhanced syntactic-semantic graph networks for aspect-based sentiment analysis. In: 2025 IEEE International Conference on Data Mining (ICDM). IEEE, pp 447–456, https://doi.org/10.1109/ICDM65498.2025.00052

Lin P, Yang M (2020) A shared-private representation model with coarse-to-fine extraction for target sentiment analysis. In: Findings of the Association for Computational Linguistics: EMNLP 2020, https://doi.org/10.18653/v1/2020.findings-emnlp.382

Lv Y, Wei F, Zheng Y, et al (2021) A span-based model for aspect terms extraction and aspect sentiment classification. Neural Computing and Applications 33:3769–3779. https://doi.org/10.1007/s00521-020-05221-x

Mitchell M, Aguilar J, Wilson T, et al (2013) Open domain targeted sentiment. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp 1643–1654, URL https://aclanthology.org/D13-1171/

Pontiki M, Galanis D, Pavlopoulos J, et al (2014) SemEval-2014 task 4: Aspect based sentiment analysis. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval), pp 27–35, https://doi.org/10.3115/v1/S14-2004

Pontiki M, Galanis D, Papageorgiou H, et al (2015) SemEval-2015 task 12: Aspect based sentiment analysis. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval), pp 486–495, https://doi.org/10.18653/v1/S15-2082

Pontiki M, Galanis D, Papageorgiou H, et al (2016) SemEval-2016 task 5: Aspect based sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval), pp 19–30, https://doi.org/10.18653/v1/S16-1002

Robinson JD, Chuang CY, Sra S, et al (2021) Contrastive learning with hard negative samples. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=CR1XOQ0UTh-

Snell J, Swersky K, Zemel RS (2017) Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems (NeurIPS), https://doi.org/10.5555/3294771.3295149

Sun Q, Yang L, Ma M, et al (2024) MiniConGTS: A near ultimate minimalist contrastive grid tagging scheme for aspect sentiment triplet extraction. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP), https://doi.org/10.18653/v1/2024.emnlp-main.165

Tian Y, Li X, Wang W, et al (2025) Large language models enhanced by plug and play syntactic knowledge for aspect-based sentiment analysis. arXiv preprint https://doi.org/10.48550/arXiv.2506.12991, arXiv:2506.12991

Wang Q, Ren J (2020) Label correction model for aspect-based sentiment analysis. In: Proceedings of the 28th International Conference on Computational Linguistics, pp 822–832, https://doi.org/10.18653/v1/2020.coling-main.71, URL https://aclanthology.org/2020.coling-main.71/

Wang Y, Sun F, Wen H (2026) Beyond cosine similarity: Zero-initialized residual complex projection for aspect-based sentiment analysis. arXiv preprint https://doi.org/10.48550/arXiv.2603.28205, arXiv:2603.28205

Wu Z, Ying C, Zhao F, et al (2020) Grid tagging scheme for aspect-oriented fine-grained opinion extraction. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp 2576–2585, https://doi.org/10.18653/v1/2020.findings-emnlp.234

Xiong H, Yan Z, Zhao H, et al (2022) Triplet contrastive learning for aspect level sentiment classification. Mathematics https://doi.org/10.3390/math10214099

Xu L, Li H, Lu W, et al (2020) Position-aware tagging for aspect sentiment triplet extraction. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), https://doi.org/10.18653/v1/2020.emnlp-main.183

Xu L, Chia YK, Bing L (2021) Learning span-level interactions for aspect sentiment triplet extraction. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL), https://doi.org/10.18653/v1/2021.acl-long.367

Zhou Y, Huang L, Guo T, et al (2019) A span-based joint model for opinion target extraction and target sentiment classification. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), pp 5485–5491, https://doi.org/10.24963/ijcai.2019/762

Zhu L, Chen X, Guo X, et al (2024) Pinpointing diffusion grid noise to enhance aspect sentiment quad prediction. In: Findings of the Association for Computational Linguistics: ACL 2024, pp 3717–3726, https://doi.org/10.18653/v1/2024.findings-acl.222