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.

Task-Conditioned Multi-Task Learning for Actor-Centric Rhetorical Positioning in Political Discourse

Samantha Butcher

samantha.butcher@uea.ac.uk

University of East Anglia https://orcid.org/0009-0000-0041-6768

Beatriz De La Iglesia

University of East Anglia https://orcid.org/0000-0003-2675-5826

Research Article

Keywords: machine learning, discourse analysis, multi-task learning

Posted Date: March 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9022111/v2

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

Additional Declarations: The authors declare no competing interests.

Declarations

Funding

The authors received no external funding for this research.

Competing interests

The authors declare that they have no competing interests.

Ethics approval and consent to participate

This study uses publicly available textual data and does not involve human subjects research requiring institutional ethics approval.

Consent for publication

Not applicable.

Data availability

The dataset is derived from publicly available sources; processed data are available from the corresponding author upon reasonable request.

Code availability

The code used for model training and evaluation is available from the corresponding author upon reasonable request.

Author contributions

The authors conceived the study, developed the methodology, conducted experiments, and wrote the manuscript.

Task-Conditioned Multi-Task Learning for Actor-Centric Rhetorical Positioning in Political Discourse

Samantha Butcher[1*] and Beatriz De La Iglesia[2†]

[1*]Department of Computing Sciences, University of East Anglia, Earlham Road, Norwich, NR4 7TJ, Norfolk, England.
[2]Department of Computing Sciences, University of East Anglia, Earlham Road, Norwich, NR4 7TJ, Norfolk, England.

*Corresponding author(s). E-mail(s): samantha.butcher@uea.ac.uk;
Contributing authors: b.iglesia@uea.ac.uk;
†These authors contributed equally to this work.

Abstract

Automatically modelling how political actors are evaluated in discourse requires methods that move beyond general sentiment classification. Existing NLP techniques typically capture overall polarity but struggle to attribute evaluative language to the specific social entities referenced within a text.

This paper introduces actor-centric rhetorical positioning, a token-level framework that links references to social actors with Positive, Neutral, or Negative evaluative labels. To model the dependency between entity identification and evaluation, we propose a Task-Conditioned Multi-Task Learning (TC-MTL) architecture in which rhetorical positioning is conditioned on entity detection. We compare this approach against Single-Task Learning (STL) and parallel multi-task baselines using DistilBERT, RoBERTa, and DeBERTa on a corpus of U.S. political discourse spanning rally speeches, formal addresses, and online political commentary.

TC-MTL improves NER performance by approximately 2–4 F1 points over STL baselines. Explicit conditioning strengthens entity extraction while maintaining positioning performance, resulting in a more reliable framework for analysing actor-centred political discourse. Our approach enables the systematic detection of how actors are constructed and positioned in political speech, supporting large-scale analysis of rhetorical framing and identity-based polarisation.

Keywords: machine learning, discourse analysis, multi-task learning

1 Introduction

Understanding how actors interact within discourse is central to contemporary discourse analysis. Many forms of public communication rely on complex relational structures in which meaning emerges not from isolated words, but from uncovering relationships between social actors. Traditional qualitative approaches can unpack these dynamics in small-scale analyses, but they struggle to scale to large corpora or to model relational dependencies systematically. Advances in Machine Learning (ML) and Natural Language Processing (NLP) offer an opportunity to model structured actor interactions at scale.

Contemporary political discourse provides a particularly salient test case. It is marked by growing polarisation, with speakers routinely constructing sharp boundaries between ingroups and outgroups to mobilise their audiences[20, 27]. In these narratives, ingroups are typically presented as those who share a collective identity and grievances, while outgroups are cast as adversaries whose presence is a threat[4, 6, 28]. Identifying this in speech or writing is critical; however, traditional NLP approaches often struggle to map these relational nuances.

Within NLP, prior work has approached evaluating language through sentiment analysis, which assigns a single polarity to a text span. While useful for capturing overall affect, this approach is too coarse-grained for analysing divisive rhetoric because it obscures how evaluation is distributed across different actors. In such discourse, ingroups are often framed as victims of injustice: although the overall sentiment may be negative, this negativity functions to legitimise and morally elevate the ingroup. Conversely, outgroups may be depicted as powerful or successful, these are surface-level positive descriptions that nonetheless position them negatively as exploitative or undeserving. For example, in the statement "they are winning while we are left behind" the outgroup 'they' is attributed with positive words but they are being implicitly placed in a neutral or negative stance as, perhaps, the 'unfair victor'; the ingroup 'we' is positioned as a deserving victim, using negative words but as a means to achieve positive positioning. The NLP task require to indentify ingroups and outgroups in political discourse is not only to identify the entities that are being referred to but also the rhetorical position that is attached to them.

Capturing these dynamics requires moving beyond sentence-level sentiment to consider how specific actors are characterised within discourse, a process we refer to as positioning (or POS). This distinguishes our approach from traditional stance detection, which typically identifies whether a speaker is explicitly for or against a given proposition or target [25] In identity-driven political discourse, evaluation rarely takes the form of explicit propositional alignment (e.g., "I support X"). Instead, actors are characterised indirectly through attribution, implication, and repeated framing. For example, statements such as "they flooded our communities and took advantage of the system" assigns a negative evaluation to an actor (they) without expressing an explicit stance toward a policy or claim.

This form of actor characterisation has been widely documented in studies of political and populist discourse, where groups are routinely constructed as threats, victims, or moral antagonists through descriptive framing rather than overt advocacy [26]. Using Positive, Neutral, and Negative positioning labels therefore more accurately reflects how political othering and ingroup can be spotted in practice.

To capture the inherent dependency between identifying an actor and evaluating their stance, we utilise a Multi-Task Learning (MTL) framework. MTL is a modelling paradigm in which a single model is trained to perform multiple related tasks simultaneously. By sharing underlying representations (such as a transformer-based encoder) across tasks, the model can exploit commonalities in the data, leading to improved generalisation and more robust feature extraction than if each task were learned in isolation.

Building on this, we propose a modified version of this architecture - Task-Conditioned Multi-Task Learning (TC-MTL) pipeline. While standard MTL models usually treat tasks as parallel, our architecture explicitly conditions rhetorical positioning on entity detection. This reflects the linguistic reality of political speech, where the identification of a specific actor - whether a pronoun, a collective identity, or a symbolic group - is a prerequisite for judging their rhetorical positioning.

Given the specialised nature of positioning within identity-driven discourse, this model is not intended as a general-purpose sentiment or stance classifier. Rather, it is designed as a theory-aligned computational tool for analysing actor-centred evaluation in political rhetoric. Because conventional NLP benchmarks prioritise broad cross-domain performance, we instead assess the architecture based on its ability to capture structured patterns of ingroup and outgroup positioning within the target

discourse. Its contribution therefore lies not in universal applicability, but in providing a scalable and systematic framework for modelling complex social actor relationships in specialised political corpora.

Our contributions, therefore, are threefold:

- We introduce actor-centric positioning as a theoretical and empirical refinement of stance detection and sentiment analysis.
- We present a specialist annotated corpus of political discourse linking abstract entities (e.g., "the elite", "our people") to their rhetorical positioning.
- We evaluate a dependency-aware MTL architecture and show that modelling the interdependence of reference and evaluation yields consistent performance advantages over single-task and flat multi-task baselines.

1.1 Our Approach

The rationale for our TC-MTL approach is grounded in the structural dependency inherent in discourse analysis. In adversarial rhetoric, meaning emerges from the interaction between who is referenced and how that reference is framed. Entities may be collective identities and symbolic groups, often outside conventional Named Entity Recognition (NER) schemas. They may involve pronominal references (e.g., pronouns such as "they" or "we") or abstract collectives (e.g., "the forgotten") and hence challenging to identify. By treating entity detection and positioning as a staged, interdependent pipeline, we allow the model to learn the "actor-role" relationship explicitly.

This same dependency structure is relevant beyond political discourse. For example, in modelling social or interpersonal relationships, evaluative judgements often depend on first identifying who is being referenced. In a sentence such as "She's always supportive, but her manager is impossible," evaluation is only meaningful once the relational targets (she, her manager) have been correctly identified. A staged entity–positioning pipeline ensures that the evaluative component operates over detected relational actors rather than inferring both target and stance simultaneously. We, therefore, suggest that this enhancement to NLP may have more general applicability than political discourse analysis.

Our experimental design explicitly evaluates the Task-Conditioned Multi-Task Learning (TC-MTL) architecture as a joint framework for both entity detection and rhetorical positioning. We begin by establishing baselines using Single-Task Learning (STL) and standard multi-task learning (MTL) with shared encoders, allowing the two tasks to be compared when trained independently or in parallel. We then introduce explicit task conditioning, in which the positioning component is trained on predicted entity outputs rather than gold-standard annotations, reflecting the operational dependency between the tasks. This staged design enables us to assess how conditioning affects performance in both entity detection and positioning, while ensuring that evaluation reflects realistic deployment settings in which oracle entity spans are unavailable.

This paper begins by situating the proposed approach within prior work on multi-task learning, sentiment analysis, and stance detection, with particular attention to how existing methods handle target-dependent evaluation. We then describe the methodological framework, including data preprocessing, model selection, and the design of the proposed TC-MTL architecture, focusing on how conditioning strategies are used to link entity identification with rhetorical positioning. The subsequent sections report the results across architectures and training regimes, followed by an examination of the results and potential future directions of this model.

2 Related Work

2.1 Multi-Task Learning and Dependency Modelling

First looking at how MTL is being used in similar settings, Wang et al. [18] introduce a conditional multi-task model, BERT-SAN (Bidirectional Encoder Representations from Transformers with Self-Attention Network), for aspect-based sentiment analysis. Aspect-based sentiment analysis is a process which identifies sentiment toward specific targets in a text, which parallels our treatment of rhetorical positioning as entity-centric evaluation. For example, in the sentence "The camera is excellent, but the battery life is disappointing," the target "camera" is associated with positive sentiment, while the target "battery life" is associated with negative sentiment. In their approach, sentiment classification is only triggered when an aspect is detected, effectively masking the sentiment loss when no aspect is present. This dependency-aware learning strategy improves the model's focus and results in a +1.81% accuracy gain over a single-task sentiment baseline.

Pei et al. [11] propose BERT-MHSA-Skip (BERT with Multi-Head Self-Attention and Skip Connections), which jointly models aspect term and aspect category sentiment classification using shared representations enhanced via multi-head self-attention and skip connections. Although the tasks are parallel rather than sequential, the architecture still captures useful inter-task signals, yielding +0.83–1.23% gains over separate STL baselines.

Zou et al. [12] adopt a more explicitly hierarchical approach in their Triple-MRC (Triple Machine Reading Comprehension) model for aspect sentiment triplet extraction. Their architecture cascades predictions through a sequence of subtasks — aspect identification, opinion extraction, and sentiment classification — each formulated as a machine reading comprehension problem. This structured dependency between sub-components yields a reported improvement of +2.37% F1 over previous benchmark systems.

Moving beyond sentiment, Chen et al. [16] propose Causal-MTL for semantic parsing, which involves translating text into structured representations of meaning. Their method introduces a mechanism for learning task dependencies by allowing discrete decisions to be learned during training. Their method achieves +1.0% F1/accuracy over both STL and fully shared MTL baselines.

Finally, Jiang et al. [24] present HMLC (Hierarchical Multi-Label Classification), a hierarchy-consistent MTL framework for multi-label classification across taxonomy levels. By combining adaptive loss weighting with constraints that preserve label consistency between parent and child classes, they achieve +1.7% F1 over a flat MTL baseline that treats labels independently.

2.2 Stance and Positioning as Structured Tasks

Stance detection is often framed as a classification task distinct from sentiment analysis. However, as shown by Alsaif and Markert [21], stance is frequently constructed in relation to a specific target, and as such, requires structural modelling to capture the relationship between entities and evaluative content. This is particularly evident in political speech, where stance is rarely explicit and often emerges through subtle framing of identity groups, policies, or abstract ideals.

Recent work increasingly acknowledges this complexity. Alturayeif et al. [17] and Peng et al. [19] show that transformer-based models outperform earlier architectures in stance detection tasks, but only when targets are clearly defined. Fu et al. [23] introduce an auxiliary "opinion-towards" task to resolve ambiguity in sentiment direction, and Li et al. [7] use hierarchical graphs to align stance and rumour veracity. Meanwhile, Abercrombie and Batista-Navarro [22] demonstrate that shared encoders struggle with long, multi-entity parliamentary speeches unless task boundaries are enforced explicitly.

These findings support the move toward structured architectures in stance modelling, which includes adopting tactics like employing auxiliary tasks, hierarchical supervision, and dependency

constraints. positioning, as a specific form of stance detection, benefits from the same framing: it should not be treated as a flat classification task, but rather as one conditioned on the identification of its targets.

Across these studies, gains range from modest to substantial, but all rely on introducing explicit dependencies between related tasks, reflecting a broader move toward dependency-aware MTL. We extend this work to adversarial ingroup–outgroup discourse, where rhetorical positioning depends on resolving who is being referenced, including abstract and collective entities. Unlike prior dependency-aware models, which primarily use task dependencies to improve performance on well-defined subtasks such as sentiment or parsing, our approach leverages dependency to support higher-level interpretive analysis, adopting a staged pipeline that moves from soft multi-task learning to explicit conditioning of positioning on upstream entity predictions, with the explicit goal of constructing an integrated pipeline that outperforms comparable STL baselines.

A full breakdown of the State-Of-The-Art (SOTA) models and tasks identified here is available in Table 1, with their relative gains versus baselines identified.

Paper	Model	Task Structure	Mechanism Type	Absolute Gain
Wang et al. [18]	BERT-SAN	Hierarchical: Aspect → Sentiment	Conditional MTL (loss gating)	+1.81% Accuracy
Pei et al. [11]	BERT-MHSA-Skip	Parallel: ATSA + ACSA	Shared MTL with MHSA	+0.83–1.23% Accuracy
Fan & Zhang [10]	BLAB	Joint: AWE + Sentiment	MTL with contextual refinement	+2.31% F1
Zou et al. [12]	Triple-MRC	Cascaded: AE → AOE → ASC	MTL with QA-style decoder	+2.37% F1
Fang et al. [15]	SenBERT	Primary + Auxiliary (MSA + Unimodal)	MTL with auxiliary regularisation	+1.9% F1
Gul et al. [14]	SP-MTL-BERT	Joint: AE + SC	MTL + Data Augmentation	+10.8% F1
Hakim et al. [9]	MTL-BERT	Joint: Aspect presence + Sentiment	Standard MTL	+1.1% Accuracy
Jia et al. [5]	ALL-IN-ONE MTL	Multi-label Peer Feedback	Standard MTL	+6% F1 (vs older baselines)
Shrestha et al. [13]	MTL-PLM	MIS + clinical symptom tasks	Auxiliary MTL	+1.5% F1
Xue et al. [8]	BERT-MLTKE	KE + Boundary ID	MTL + Feature Fusion	+1.5% F1
Chen et al. [16]	Causal-MTL	Hierarchical NLP tasks	Causal-Inference MTL	+1.0% F1/Accuracy
Jiang et al. [24]	HMLC	Hierarchical multi-label classification	Consistency Loss + Adaptive Weights	+1.7% F1

Table 1: Comparison of Recent Dependency-Aware MTL Architectures

3 Methodology

The core of the corpus comprises 674 speeches delivered between 2015–2024, drawn from publicly available collections of U.S. political speeches and campaign rallies [1, 2]. To increase structural diversity beyond rally-style monologic discourse, we supplemented this material with politically relevant texts drawn from a large-scale misinformation dataset [3]. From this source, we randomly sampled a subset of political news articles and associated reader commentary and annotated them using the same entity–positioning framework.

For preprocessing, timestamps and non-verbal metadata were stripped, and each speech was segmented into context windows averaging 130–160 characters. This segmentation balances semantic coherence with model efficiency: smaller context windows further help isolate rhetorical structures, particularly where multiple entities appear in close proximity. This design choice also responds to the issue noted by Abercrombie and Batista-Navarro [22], who found that transformer encoders underperform on long parliamentary speeches due to input length limits.

Annotation proceeded in two stages. First, entities were identified and tagged across the corpus, creating, in some instances, multi-word entity spans. An entity span is the smallest continuous piece of text that names or refers to an actor or idea being discussed or evaluated in the discourse. This includes not only named entities, but also pronominal references (e.g., *we*, *they*), collective or institutional groups (e.g., *American workers*, *the Senate*), and abstract or symbolic referents (e.g., *the American Dream*, *freedom*, *10,000 soldiers*) when these function as targets of praise, blame, or moral judgment. This represents a divergence away from conventional NER schema, which typically restrict entities to named persons, locations, or organisations. Our schema was extended to include collective and abstract referents, reflecting the fact that political speech frequently targets groups, identities, and symbolic concepts rather than only concrete individuals or institutions. Abstract references such as *freedom* or *the American Dream* often operate as proxies for social groups or belief systems, allowing speakers to legitimise allies or delegitimise opponents without explicitly naming them. Treating such references as entities therefore enables the model to capture how evaluation is distributed across both concrete and symbolic actors, reflecting the communicative strategies through which political meaning is constructed. We distinguish between pronominal references, labelled pronouns in the dataset (e.g., *we, they*), and a category we call identity markers (e.g., *American workers, the elite*) because they behave differently in discourse. Pronouns are context-dependent references, whereas identity markers explicitly encode group membership. Separating these categories allows the model to learn distinct boundary and referential patterns associated with these differently named actors.

Second, each entity was assigned a positioning label (`POSITIVE`, `NEUTRAL`, `NEGATIVE`) according to its surrounding context. Labelling was performed at the entity level rather than the sentence level, since a single sentence can frame different entities in opposing ways. For example:

"They are destroying our country while we defend freedom."

Here, the entities *they*, *our country*, *we*, and *freedom* each receive separate entity labels, reflecting their different rhetorical roles they play within the same span of text.

An initial annotation pass produced a large pool of entity–positioning examples, exceeding the number ultimately used for model training. From this pool, a balanced subset was curated by reducing over-represented entities and removing repeated or structurally similar spans. This selection process was primarily qualitative rather than strictly statistical, with the aim of preventing the model from overfitting to highly frequent actors or formulaic constructions. The resulting dataset prioritised structural and referential variety over raw frequency, supporting more robust generalisation.

The final dataset contained 10,104 annotated entity instances, which corresponded to 5,030 unique sentences. The difference reflects the annotation format: for practical labelling purposes, each entity within a sentence was recorded on a separate row. As a result, sentences containing multiple actors appeared multiple times in the raw annotation file. For example:

They are corrupt and our people deserve better.
They PRONOUN NEGATIVE

They are corrupt and our people deserve better.
our people IDENTITY_MARKER POSITIVE

During preprocessing for model training, these duplicate sentence rows were merged so that each unique sentence appeared once, with all associated entity spans converted into BIO tags within a single token sequence. An example of this collapsed representation is provided in the following section.

3.0.1 BIO-tagging

All models are trained end-to-end using dual BIO-tagged supervision. BIO (Begin–Inside–Outside) tagging is a standard sequence-labelling scheme in NLP that represents span-based annotations by

Dataset Name	Date Published	Last Updated	Date of Access	Dataset Count	Domain
US Presidential Rally Speeches	2020	2020	July 2025	35	Political Speeches
USA Political Speeches	2022	2024	July 2025	639	Political Speeches
Misinformation & Fake News Text Dataset	2022	2022	August 2025	78,617	Political News and Online Commentary

Table 2: Overview of datasets used in corpus creation.

marking the beginning of an entity (B for Begin), its continuation across tokens (I for Inside), and tokens that fall outside any annotated span (O for Outside). This formulation allows models to learn both token-level labels and multi-token span boundaries in a single prediction task, making it well suited to entity-based annotation problems.

Unlike conventional BIO schemes, which typically encode only span boundaries for a single task, our formulation jointly encodes both span structure and entity type. For entity recognition, we assign `B-` tags (e.g., `B-PRONOUN`, `B-IDENTITY_MARKER`) to mark the start of a span and the corresponding `I-` tags to indicate continuation across multiple tokens. An equivalent labelling scheme is applied for rhetorical positioning, with tags such as `B-POSITIVE` or `I-NEGATIVE`. This dual labelling produces two aligned BIO sequences for every input: one for entity detection and one for positioning.

Using BIO tagging for both tasks allows the model to learn consistent span boundaries while assigning different interpretive labels to those spans. This supports fine-grained modelling of entity-conditioned evaluation, enabling entity recognition and rhetorical positioning to be learned as distinct but interdependent tasks within a shared token-level framework.

Token	NER BIO	Positioning BIO
They	`B-PRONOUN`	`B-NEGATIVE`
are	`O`	`O`
targeting	`O`	`O`
our	`B-IDENTITY_MARKER`	`B-POSITIVE`
veteran	`I-IDENTITY_MARKER`	`I-POSITIVE`
##s	`I-IDENTITY_MARKER`	`I-POSITIVE`
.	`O`	`O`

Table 3: Illustration of dual BIO sequences for entity recognition and rhetorical positioning.

To formalise this dual sequence labelling task, we define the label spaces as follows. Here, $\mathcal{Y}_{NER}$ and $\mathcal{Y}_{POS}$ denote the sets of possible output labels for the NER and positioning tasks, respectively.

Formal Task Definition:

Let an input token sequence be $T = (t_1, \ldots, t_n)$. For each token t_i, the model predicts two aligned label sequences:

$$y_i^{NER} \in \mathcal{Y}_{NER}, \qquad y_i^{POS} \in \mathcal{Y}_{POS} \tag{1}$$

where

$$\mathcal{Y}_{NER} = \{O, B\text{-}PRONOUN, B\text{-}IDENTITY_MARKER, I\text{-}IDENTITY_MARKER\} \tag{2}$$

$$\mathcal{Y}_{POS} = \{O, B\text{-}POSITIVE, I\text{-}POSITIVE, B\text{-}NEGATIVE, I\text{-}NEGATIVE, B\text{-}NEUTRAL, I\text{-}NEUTRAL\} \tag{3}$$

Thus, the task is formulated as dual token-level sequence labelling with aligned BIO structures.

3.1 Architecture

The study evaluates three transformer-based encoder models — **RoBERTa**, **DistilBERT**, and **DeBERTa**. They were chosen to represent a spectrum of architectural complexity and computational cost, as well as linguistic sensitivity. DistilBERT serves as a smaller, faster baseline with fewer parameters, making it efficient to train and evaluate. RoBERTa is a strong and widely used transformer model that benefits from large-scale pre-training and generally provides reliable contextual representations. DeBERTa extends this architecture by separating content and positional information within its attention mechanism, which often makes it particularly effective for detailed token-level tasks such as entity detection. Together, these models allow us to assess whether the proposed training regimes benefit predominantly from increased representational capacity or from architectural features that better capture subtle discourse structures.

The core experimental setup uses each backbone under three training regimes:

- **Single-Task Learning (STL)**: independent models trained separately for NER and for rhetorical positioning.
- **Multi-Task Learning (MTL)**: a shared encoder with two parallel heads, one for NER and one for positioning, trained jointly but without explicit conditioning.
- **Task-Conditioned Multi-Task Learning (TC-MTL)**: a shared encoder with an NER head feeding into a positioning head, such that rhetorical positioning is conditioned on entity detection. This is our particular contribution. Two different versions of this pipeline were tested, which will be covered in a later section.

The TC-MTL training pipeline is illustrated in Figure 1.

3.1.1 Shared Encoder Configuration:

In the TC-MTL configuration only, the positioning head is conditioned on the NER probability distribution. So given token sequence T, a transformer encoder produces contextual representations in this manner:

$$H = (h_1, \ldots, h_n), \qquad h_i \in \mathbb{R}^d \tag{4}$$

NER Head:

Token-level NER logits are computed as:

$$z_i^{NER} = W_{NER} h_i + b_{NER} \tag{5}$$

with probability distribution:

$$p_i^{NER} = \text{softmax}(z_i^{NER}) \tag{6}$$

Positioning:

In the TC-MTL configuration, positioning is conditioned on the NER distribution. The NER probabilities are projected and concatenated with encoder features:

$$\tilde{h}_i = \left[h_i \parallel W_C p_i^{NER}\right] \tag{7}$$

Positioning logits are then computed as:

$$z_i^{POS} = W_{POS}\tilde{h}_i + b_{POS} \tag{8}$$

$$p_i^{POS} = \text{softmax}(z_i^{POS}) \tag{9}$$

3.2 Initial Baselines and Hyperparameters

We began with a set of hyperparameters chosen to provide a stable and conservative baseline across all architectures (Table 4). The learning rate was fixed at $5e^{-5}$ with a constant schedule and AdamW

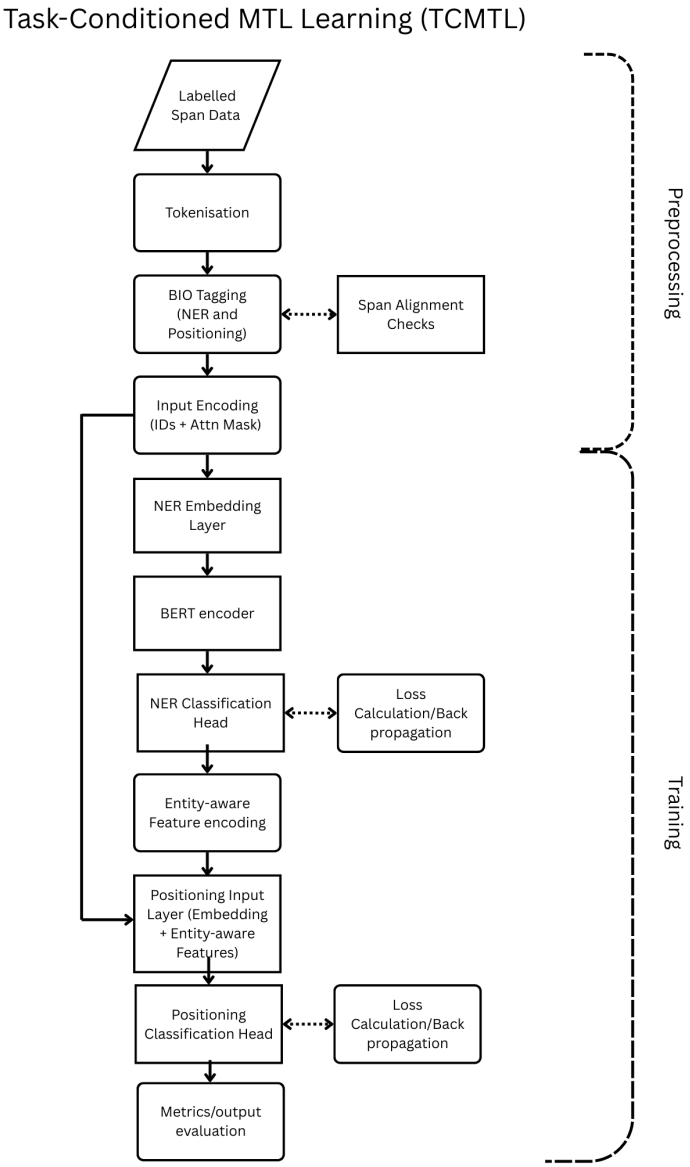


Fig. 1: TC-MTL Pipeline

optimiser, settings widely used in transformer fine-tuning. Batch size (16), weight decay (0.05), and dropout (0.1) were kept moderate to reduce overfitting risk while maintaining efficiency. Class weights were defined manually to counteract the dominance of the `O` class and to emphasise span boundaries, under the assumption that boundaries carry more informational weight than interiors. The specific weight maps used for NER and positioning are shown in Table 5.

For TC-MTL, a two-epoch warm-up phase was introduced for NER only, allowing the entity head to stabilise before conditioning the positioning head. Losses for both tasks were computed using weighted cross-entropy.

We set $\lambda_{NER} : \lambda_{POS} = 0.3 : 0.7$ after pilot runs showed that equal weighting produced unstable joint optimisation due to loss-scale imbalance and the greater semantic diffuseness of the positioning objective.

Hyperparameter	**STL**	**MTL**	**TC-MTL**
Base encoder	Finetuned BERT-family (RoBERTa, DistilBERT, DeBERTa)		
Learning rate	5e–5		
LR scheduler	Constant (no warmup, no decay)		
Optimizer	AdamW		
Weight decay	0.05		
Task warmup	None		NER-only (2 epochs)
Batch size	16		
Epochs	10		
NER loss	Weighted CE		
POS loss	Weighted CE		
Loss setup (NER:POS)	Single-task	Sum (1.0:1.0)	Weighted (0.3:0.7)
Conditioning	None		Softmax NER → POS
Seq. length	512		
Dropout	0.1		
Ignore index	-100		
Seeds	21,22,42,84,128		

Table 4: Initial hyperparameters and training setups across STL, MTL, and baseline TC-MTL models.

3.2.1 Joint Optimisation

For multi-task configurations (MTL and TC-MTL), training is formulated as joint minimisation of NER and positioning losses.

Standard Positioning Loss

In STL, MTL, and soft TC-MTL, positioning is trained over all tokens using weighted cross-entropy:

$$\mathcal{L}_{POS} = -\frac{1}{n}\sum_{i=1}^{n} w_{y_i^{POS}}^{POS} \log p_i^{POS}(y_i^{POS}) \tag{10}$$

Entity-Masked Positioning Loss (Advanced TC-MTL)

In the advanced conditioning regime, positioning is supervised only on entity tokens:

$$m_i = \mathbb{1}[y_i^{NER} \neq O] \tag{11}$$

The masked positioning loss is:

$$\mathcal{L}_{POS}^{masked} = -\frac{1}{\sum_{i=1}^{n} m_i}\sum_{i=1}^{n} m_i \, w_{y_i^{POS}}^{POS} \log p_i^{POS}(y_i^{POS}) \tag{12}$$

Total Training Objective

For standard MTL:

$$\mathcal{L}_{total} = \lambda_{NER}\mathcal{L}_{NER} + \lambda_{POS}\mathcal{L}_{POS} \tag{13}$$

For advanced TC-MTL:

$$\mathcal{L}_{total}^{adv} = \lambda_{NER}\mathcal{L}_{NER} + \lambda_{POS}\mathcal{L}_{POS}^{masked} \tag{14}$$

Task	Fixed weight map
NER	{`O`: 0.05, `B-PRONOUN`: 2.0, `B-IDENTITY_MARKER`: 2.0, `I-IDENTITY_MARKER`: 1.0}
Positioning	{`O`: 0.05, `B-POSITIVE`: 2.5, `I-POSITIVE`: 1.2, `B-NEGATIVE`: 2.5, `I-NEGATIVE`: 1.2, `B-NEUTRAL`: 1.5, `I-NEUTRAL`: 0.8}

Table 5: Initial fixed class weight mappings for NER and positioning.

3.3 From Baselines to Focused Investigation

Although our overarching aim was always to evaluate a task-conditioned multi-task learning pipeline, we first established STL and MTL baselines to provide comparative reference points. These runs ensured that performance across different training regimes could be contextualised, allowing us to assess the added complexity of conditioning against simpler alternatives.

Once baseline comparisons were established, we concentrated all further refinement efforts on the TC-MTL setup. This decision reflected the central role of TC-MTL in our research question and enabled a focused programme of conditioning strategy development.

3.4 Conditioning Strategies

The TC-MTL architecture enabled exploration of different strategies for conditioning the positioning head on NER outputs. In the baseline setup, the positioning head was trained directly on the softmax distributions of the NER head at each token. We then extended this to discrete conditioning strategies, masking non-entity tokens during loss computation so that the positioning head was supervised only on entity spans. We called this our "advanced" TC-MTL pipeline.

3.4.1 Conditioning During Training

In the advanced TC-MTL configuration, rhetorical positioning is supervised only on tokens identified as entities. Conditioning is introduced progressively to stabilise optimisation and to reduce the training–inference discrepancy.

Gold Conditioning Phase

Following the two-epoch NER warm-up, the positioning head is trained using gold-standard entity labels to construct the entity mask:

$$m_i^{gold} = \mathbb{1}[y_i^{NER} \neq O] \tag{15}$$

During this phase, the positioning head receives perfect boundary information, allowing us to assess its intrinsic classification capacity independent of upstream prediction noise.

Mixed Conditioning Phase

To bridge the gap between training and deployment, we introduce a mixed regime in which each mini-batch conditions on gold labels with probability 0.5, and on predicted NER labels otherwise:

$$m_i^{pred} = \mathbb{1}[\hat{y}_i^{NER} \neq O] \tag{16}$$

This progressively exposes the positioning head to noisy upstream predictions while retaining stabilising access to correct spans.

Predicted Conditioning Phase

In the final training phase, conditioning relies exclusively on predicted NER outputs. This mirrors inference-time behaviour, where gold entity spans are unavailable.

Given a mask m, the masked positioning loss is defined as:

$$\mathcal{L}_{POS}^{masked}(m) = -\frac{1}{\sum_{i=1}^{n} m_i} \sum_{i=1}^{n} m_i \, w_{y_i^{POS}}^{POS} \log p_i^{POS}(y_i^{POS}) \tag{17}$$

At inference time, only predicted conditioning is available.

3.5 Metrics

Given the nature of the two distinct tasks encapsulated in this study, metric evaluation must capture both independent task performance and the interactions between them. We therefore report several complementary metric sets. Each targets a distinct aspect of system behaviour and, taken together, provides a comprehensive account of both span identification and stance assignment.

- **Overall Metrics**: Calculated across all tokens (including 'O'). These provide a high-level indication of label reliability but can be inflated by the large number of non-entity tokens (which are easier to identify due to the quantity of them in the corpora). They function mainly as a general stability check.
- **Entity Metrics**: Computed only over entity tokens, excluding 'O'. These metrics directly assess the model's ability to detect and correctly BIO-label meaningful spans, offering the clearest view of NER and positioning performance on relevant units.
- **Gold Metrics**: Evaluates positioning strictly on gold-standard entity spans. By removing the influence of NER errors, they estimate the upper bound of the model's positioning-classification ability.
- **NER-Correct Metrics**: Assesses positioning only where the model's predicted NER labels match the gold labels. This reflects how well the model performs when it has correctly identified the entities on which it conditions its positioning interpretation.
- **Chained Metrics**: Evaluates positioning on the model's predicted entity spans, regardless of correctness. This forces the positioning head to operate with NER as a hard dependency and reveals the system's end-to-end behaviour, a particularly important element in the advanced TC-MTL architecture, where task dependencies are enforced.

4 Results

This section evaluates how increasingly structured forms of multi-task learning affected both entity detection and rhetorical positioning within a dependency-aware framework. We began with the STL baseline, to establish how well each task can be learned independently, without shared representations or cross-task influence.

We then examined standard MTL, in which the two tasks share a common encoder. This allowed us to assess whether shared representations produce mutual gains, or whether performance is affected by task interference.

Metric Type	Question Addressed
Overall Metrics	How reliably does the system label tokens across the entire sequence, including non-entity positions?
Entity Metrics	How well does the model identify and correctly label entity spans, excluding non-entity tokens?
Gold Metrics	How well does the POS classifier perform when supplied with perfect (gold) entity boundaries?
NER-Correct Metrics	How well does the POS classifier perform when the model's own NER predictions match the gold spans?
Chained Metrics	How effectively does the full system operate end-to-end, with POS depending entirely on the entity spans predicted by NER?

Table 6: Summary of metric types and the diagnostic questions they address.

Building on this, we then tested TC-MTL, in which the structural dependency between the tasks was modelled explicitly. We first evaluated a soft conditioning regime, where positioning was informed by entity representations without being fully constrained by them (namely via softmax probabilistic conditioning, allowing positioning to incorporate NER uncertainty signals without being strictly constrained by predicted entity boundaries - a "soft" signal). Finally, we examined explicit conditioning, progressively moving toward a fully chained configuration in which positioning only operated over predicted entity spans. This staged evaluation allowed us to examine how modelling task dependency influences both entity boundary learning and evaluative classification under increasingly realistic conditions.

4.1 STL

Our baseline is to first study each task independently using Single-Task Learning (STL). This allows us to assess the intrinsic difficulty of each task in isolation and to measure any gains introduced by multi-task learning in subsequent sections.

4.1.1 NER

Performance in the NER STL configuration was highly consistent across all evaluated architectures. Results are reported using both entity-level metrics, which assess performance on entity spans only, and overall token-level metrics, which reflect accuracy across the full sequence including non-entity tokens (see Table 7).

At the entity level, Entity F1 scores clustered within a narrow range. The highest Entity F1 score was 0.710 (achieved by both RoBERTa and DeBERTa), marginally exceeding DistilBERT's 0.707. Differences were observed in stability: RoBERTa and DeBERTa demonstrated the lowest variability (SD = 0.026), indicating tighter clustering of results across runs, while DistilBERT exhibited the highest standard deviation (SD = 0.033).

Despite similar Entity F1 scores, models showed different Precision–Recall trade-offs. RoBERTa achieved the highest entity-level Recall (0.788), whereas DeBERTa and DistilBERT achieved higher entity-level Precision (0.680 and 0.670, respectively).

At the token level, Overall F1 scores ranged narrowly from 0.906 to 0.912, with DeBERTa achieving the highest value. Standard deviations for these overall metrics were uniformly small (between 0.003 and 0.005), reflecting stable performance across runs.

Taken together, these results indicate that under an STL setup, NER performance is both high and stable across all models. Differences between architectures are minor and primarily reflect Precision–Recall trade-offs rather than substantive differences in overall entity detection capability.

	Entity			E. Loss	Overall		
Model	**F1**	**Pr**	**Re**		**F1**	**Pr**	**Re**
DistilBERT	0.707 (0.033)	0.670 (0.042)	0.755 (0.035)	0.918 (0.048)	0.908 (0.005)	0.911 (0.005)	0.906 (0.005)
RoBERTa	**0.710** (**0.026**)	0.647 (**0.026**)	**0.788** (**0.028**)	**0.772** (0.071)	0.906 (**0.004**)	0.913 (0.004)	0.903 (0.005)
DeBERTa	**0.710** (**0.026**)	**0.680** (0.029)	0.751 (0.035)	0.964 (**0.034**)	**0.912** (**0.004**)	**0.915** (**0.003**)	**0.910** (**0.004**)

Table 7: NER performance under STL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

4.1.2 POS

The positioning task clearly appears to be the harder task, which also shows greater architectural discrimination between models, although the absolute standard deviations (SDs) remain low (full results in Table 8). For the entity-filtered scores, RoBERTa offers both the highest mean score (0.607) and the most stable performance on this metric, achieving an exceptionally low SD of 0.002. The Gold metrics, which evaluate positioning only on gold-standard entity spans (the classification head's upper bound), show the same ordering: Gold F1 is highest for RoBERTa (0.742, SD = 0.005). SDs are very small for all three models, indicating that the classification head itself is generally reliable.

On positioning, RoBERTa provides the strongest and most stable performance, DeBERTa is competitive but slightly less consistent, and DistilBERT lags behind on both entity-filtered and Gold scores.

	Entity			E. Loss	Overall			Gold		
Model	**F1**	**Pr**	**Re**		**F1**	**Pr**	**Re**	**F1**	**Pr**	**Re**
DistilBERT	0.561 (0.008)	0.495 (0.016)	0.657 (0.012)	**0.537** (**0.009**)	**0.854** (0.006)	**0.876** (0.016)	0.841 (0.007)	0.690 (**0.005**)	0.734 (0.006)	0.657 (0.012)
RoBERTa	**0.607** (**0.002**)	0.539 (**0.005**)	**0.698** (**0.007**)	1.612 (0.049)	0.847 (**0.002**)	0.847 (**0.002**)	**0.858** (**0.001**)	**0.742** (**0.005**)	0.793 (**0.003**)	**0.698** (**0.007**)
DeBERTa	0.606 (0.009)	**0.576** (0.019)	0.652 (0.013)	1.624 (0.569)	0.813 (0.123)	0.834 (0.092)	0.816 (0.104)	0.723 (0.007)	**0.800** (0.006)	0.660 (0.009)

Table 8: POS performance under STL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

4.2 MTL

4.2.1 NER

For full results, see Table 9. The NER task under MTL shows a modest trade-off between mean accuracy and run-to-run stability. DeBERTa and DistilBERT had better mean F1 scores. RoBERTa

produced the lowest mean Entity F1 (0.667) but was the most stable model, with a standard deviation of only 0.010. RoBERTa performs slightly worse on average, but it converges more consistently across runs. As with the STL configuration, the three models displayed distinct precision–recall tendencies.

Relative to the STL baseline, NER performance under standard MTL decreases slightly across all three architectures, with lower mean Entity F1 scores and, in some cases, increased variability. This pattern suggests that naïve parameter sharing introduces a degree of task interference, likely reflecting representational noise from the positioning task being injected into the shared encoder. While RoBERTa benefits from improved stability under MTL, this comes at the cost of reduced mean performance, whereas DeBERTa and DistilBERT retain higher average scores but exhibit greater variance. These results align with prior findings that soft parameter sharing can be suboptimal for hierarchically related tasks, particularly when one task (here, rhetorical positioning) is more complex and semantically diffuse than the other. Importantly, this degradation motivates the move toward explicit task conditioning, rather than further shared learning, to see whether this trend continues.

	Entity			E. Loss	Overall		
Model	**F1**	**Pr**	**Re**		**F1**	**Pr**	**Re**
DistilBERT	0.682 (0.029)	0.619 (0.029)	0.682 (0.043)	0.256 (**0.029**)	0.901 (**0.004**)	**0.909** (**0.005**)	0.897 (**0.004**)
RoBERTa	0.667 (**0.010**)	0.575 (**0.013**)	**0.806** (**0.026**)	**0.220** (0.046)	0.890 (0.005)	0.905 (0.007)	0.884 (0.005)
DeBERTa	**0.687** (0.032)	**0.626** (0.041)	0.770 (0.053)	0.265 (**0.029**)	**0.902** (**0.004**)	**0.909** (**0.005**)	**0.898** (**0.004**)

Table 9: NER performance under MTL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

4.2.2 POS

Table 10 shows the full results for the positioning task. The Entity F1 scores clustered tightly around 0.60 - DistilBERT performed lower overall (0.577) but was the most stable model for this metric (SD = 0.005).

The Gold metrics show more separation between models. RoBERTa achieved the highest Gold F1 (0.728, SD = 0.020), followed by DeBERTa (0.710, SD = 0.015) and DistilBERT (0.681, SD = 0.008). Although RoBERTa exhibited slightly higher variance here, it still demonstrated the strongest upper-bound classification capacity.

Compared to the STL baseline, standard MTL does not yield consistent gains, with entity-level F1 scores remaining clustered around 0.60. One possible explanation is that shared encoder representations alone do not resolve the structural dependency between entity detection and positioning: the positioning head must still infer entity boundaries without explicit conditioning. Although Gold scores indicate strong upper-bound classification capacity, this potential is not fully realised under naïve joint training. These findings motivate the move to Task-Conditioned MTL, in which entity predictions are supplied directly to the positioning head to reflect their hierarchical relationship.

Model	Entity			E. Loss	Overall			Gold		
	F1	Pr	Re		F1	Pr	Re	F1	Pr	Re
DistilBERT	0.577 (**0.005**)	0.534 (0.008)	0.633 (**0.008**)	0.630 (**0.038**)	0.865 (**0.004**)	0.876 (**0.005**)	0.858 (**0.004**)	0.681 (**0.008**)	0.741 (0.015)	0.633 (**0.008**)
RoBERTa	0.600 (0.013)	0.535 (0.012)	**0.690** (0.026)	**0.522** (0.050)	0.859 (0.006)	0.877 (0.008)	0.849 (0.005)	**0.728** (0.020)	**0.777** (0.015)	**0.690** (0.026)
DeBERTa	**0.604** (0.009)	**0.559** (**0.007**)	0.661 (0.022)	0.618 (0.046)	**0.869** (0.005)	**0.879** (0.006)	**0.863** (0.005)	0.710 (0.015)	0.772 (**0.009**)	0.661 (0.022)

Table 10: POS performance under MTL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

4.3 Softmax TC-MTL

4.3.1 NER

Full results can be found in Table 11. In the TC-MTL configuration, NER performance sits below the STL baselines but remains broadly comparable to the MTL results. DeBERTa achieved the highest mean Entity F1 (0.665, SD = 0.017), slightly below its MTL score (0.687). RoBERTa performed almost identical to its MTL implementation, while DistilBERT showed the largest drop relative to MTL.

RoBERTa was the most stable model (SD = 0.016), with DeBERTa close behind, and showed a similar imbalance between Precision–recall.

Relative to the STL baseline, NER performance under softmax TC-MTL is consistently lower, indicating that the introduction of task conditioning does not benefit entity detection itself. Compared to standard MTL, results are broadly comparable: mean Entity F1 scores remain in the same range, and differences across architectures are small. This suggests that soft conditioning neither substantially alleviates nor exacerbates the task interference observed under shared-encoder MTL, with stability rather than raw performance being the primary distinguishing factor between models.

Model	Entity			E. Loss	Overall		
	F1	Pr	Re		F1	Pr	Re
DistilBERT	0.605 (0.018)	0.511 (0.019)	0.748 (0.030)	0.178 (0.013)	0.880 (0.004)	0.903 (**0.001**)	0.871 (0.006)
RoBERTa	0.647 (**0.016**)	0.544 (**0.011**)	0.802 (**0.023**)	**0.156** (**0.007**)	0.872 (**0.003**)	0.901 (**0.001**)	0.863 (**0.004**)
DeBERTa	**0.665** (0.017)	**0.562** (0.022)	**0.823** (0.029)	0.170 (0.037)	**0.885** (0.005)	**0.907** (0.005)	**0.877** (0.006)

Table 11: NER performance under TC-MTL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

4.3.2 POS

Please see Table 12 for full metrics. For the positioning task, the mean Entity F1 scores were tightly clustered around 0.57. DeBERTa achieved the highest score (0.579, SD = 0.013).

Although the mean differences were small, RoBERTa was the most stable model, showing substantially lower variance than DeBERTa and DistilBERT.

The Gold metrics showed a clearer separation between models. RoBERTa achieved the highest Gold F1 (0.736, SD = 0.007), followed by DeBERTa (0.723, SD = 0.013) and DistilBERT (0.703, SD = 0.017). RoBERTa again combined the strongest mean score with the lowest variability.

For the positioning task, softmax TC-MTL produces results similar to those observed under standard MTL. Entity-level F1 scores remain tightly clustered and do not exceed STL performance, indicating that soft conditioning alone is insufficient to unlock the upper-bound capacity suggested by the Gold metrics. However, Gold scores remain high and stable, particularly for RoBERTa, confirming that the positioning head retains strong classification potential. Taken together, these results suggest that while soft conditioning preserves model stability, it does not meaningfully strengthen the dependency between entity detection and positioning.

	Entity			E. Loss	Overall			Gold	
Model	**F1**	**Pr**	**Re**		**F1**	**Pr**	**Re**	**F1**	**Pr**	**Re**
DistilBERT	0.557 (0.013)	0.484 (0.017)	0.666 (0.021)	0.550 (0.031)	0.849 (0.005)	0.875 (0.002)	0.835 (0.007)	0.703 (0.017)	0.751 (0.010)	0.666 (0.021)
RoBERTa	0.577 (**0.006**)	0.490 (**0.007**)	**0.710** (**0.006**)	**0.473** (**0.017**)	0.842 (**0.002**)	0.875 (**0.001**)	0.826 (**0.004**)	**0.736** (**0.007**)	**0.771** (**0.006**)	**0.710** (**0.006**)
DeBERTa	**0.579** (0.013)	**0.502** (0.025)	0.694 (0.022)	0.506 (0.056)	**0.853** (0.005)	**0.879** (0.006)	**0.839** (0.007)	0.723 (0.013)	0.760 (0.012)	0.694 (0.022)

Table 12: POS performance under TC-MTL. Entity metrics evaluate performance on entity spans only, while Overall metrics reflect token-level performance across the full sequence. Values show mean and standard deviation (shown in brackets) across runs.

Taken together, the standard MTL and softmax TC-MTL results show that neither shared encoders nor soft conditioning yield clear performance gains over learning the tasks independently. Relative to STL, both MTL and softmax TC-MTL tend to reduce mean Entity F1 scores, particularly for NER, indicating that additional task interaction introduces a degree of interference rather than synergy. Where differences do emerge, they are primarily reflected in stability rather than absolute performance: RoBERTa, in particular, benefits from reduced run-to-run variance under multi-task configurations, especially for positioning. However, these stability gains do not translate into consistent improvements in entity-conditioned positioning accuracy. Overall, these findings suggest that simply encouraging interaction between tasks is insufficient; stronger, more explicit dependency modelling is may be required to realise the potential advantages of multi-task learning in this setting.

4.4 Advanced Conditioning TC-MTL

4.4.1 NER

Table 13 reports full NER performance under the advanced TC-MTL conditioning regime. DeBERTa achieves the highest Entity F1 (0.749, SD = 0.014), alongside improvements in overall accuracy and overall F1, indicating both stronger mean performance and improved stability. RoBERTa attains a lower mean Entity F1 (0.729) and exhibits substantially higher variance, despite maintaining a recall-dominant profile.

Figure 2 situates these results relative to the earlier STL, MTL, and soft TC-MTL configurations. The comparison shows that advanced TC-MTL yields the largest absolute improvement for DeBERTa, while RoBERTa's gains in Entity F1 are accompanied by increased run-to-run variability. Together, these results indicate that advanced conditioning benefits DeBERTa more consistently, whereas RoBERTa remains sensitive to this new conditioning complexity despite strong recall.

Model	Entity F1	Entity Pr	Entity Re	E. Loss	Overall F1	Overall Pr	Overall Re
RoBERTa	0.729 (0.095)	0.606 (**0.014**)	**0.916** (**0.011**)	**0.407** (**0.042**)	0.880 (**0.007**)	0.907 (**0.005**)	0.871 (**0.008**)
DeBERTa	**0.749** (**0.014**)	**0.644** (0.029)	0.897 (0.021)	0.492 (0.067)	**0.891** (0.010)	**0.909** (0.006)	**0.884** (0.012)

Table 13: Advanced TC-MTL NER performance (mean with SD shown beneath in parentheses).

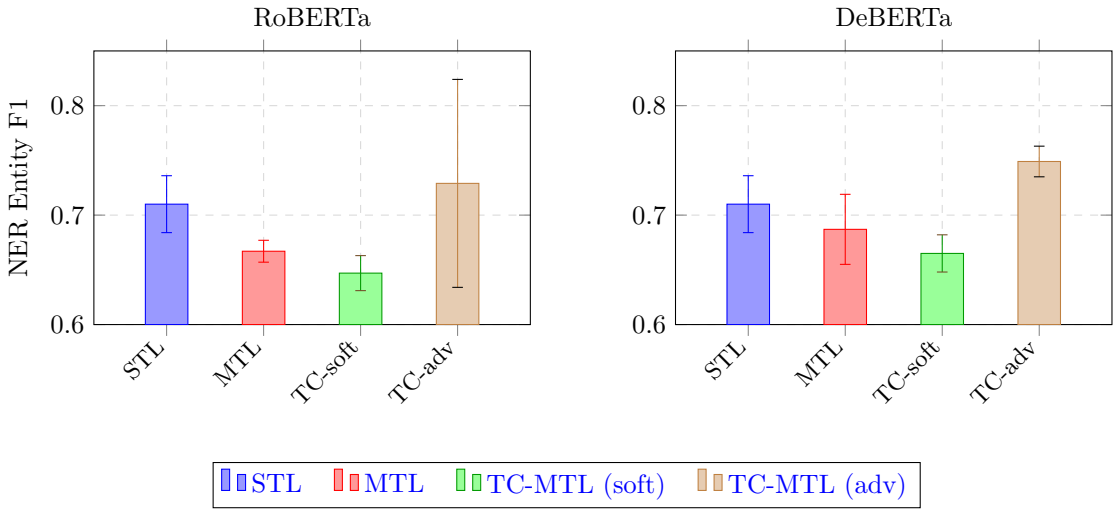


Fig. 2: NER Entity F1 scores (mean ± SD) for RoBERTa and DeBERTa across STL, MTL, and TC-MTL configurations.

4.4.2 POS

While absolute differences in positioning accuracy remain small, the advanced TC-MTL configuration reveals a distinct shift in how the models handle task dependency (see Table 14). At the entity level, DeBERTa achieves a marginally higher mean F1 (0.585) than RoBERTa (0.577), though RoBERTa continues to demonstrate superior run-to-run stability.

One of the key takeaways is that the entity-level results (Figure 3) show that advanced TC-MTL also leads to a performance dip compared to the STL baseline, following the pattern of MTL structures. RoBERTa falls from 0.607 to 0.577, and DeBERTa declines from 0.606 to 0.585. Although positioning performance declines once again relative to STL, the underlying cause most likely differs under this new architecture. In the STL setting, the positioning head was likely performing a degree of implicit entity detection alongside its evaluative task. By effectively running two parallel boundary processes, one specialised (NER) and one auxiliary (POS), the STL regime could achieve higher boundary precision. Under explicit conditioning, this redundancy is removed. The hand-off between tasks becomes a bottleneck: any misalignment in NER boundaries now directly constrains the positioning output, and this is the reduction (albeit small) we are seeing in Entity F1 scores. The drop might appear to be similar, but the causes are different.

The gold-standard evaluation (Figure 4) confirms that this change is structural rather than a degradation of classification capacity. When provided with perfect entity spans, performance (0.736 for RoBERTa; 0.724 for DeBERTa) remains comparable to STL. This indicates that the positioning head's evaluative logic remains intact; what has changed is the removal of the auxiliary boundary inference in favour of a stricter and more transparent dependency.

The chained positioning scores illustrate how this system functions in real-world settings where both tasks rely on model predictions. In this "noisy" environment, DeBERTa achieves a higher chained F1 (0.612) than RoBERTa (0.595). This suggests that while RoBERTa has the edge in "oracle" settings (perfect data), DeBERTa is more resilient when handling imperfect entity information.

Taken together, the results indicate that explicit task conditioning enhances entity boundary learning while largely preserving positioning competence. The advanced TC-MTL configuration therefore provides a stronger and more structurally coherent modelling approach than naïve multi-task alternatives.

	Entity			E. Loss	Overall			Gold		
Model	**F1**	**Pr**	**Re**		**F1**	**Pr**	**Re**	**F1**	**Pr**	**Re**
RoBERTa	0.577	0.489	**0.712**	**1.882**	0.848	**0.881**	0.831	**0.736**	**0.766**	**0.712**
	(**0.006**)	(**0.006**)	(**0.014**)	(**0.095**)	(**0.009**)	(**0.005**)	(**0.010**)	(**0.012**)	(**0.011**)	(**0.014**)
DeBERTa	**0.585**	**0.512**	0.691	2.271	**0.857**	0.880	**0.844**	0.724	0.765	0.691
	(0.014)	(0.024)	(0.031)	(0.213)	(0.012)	(0.009)	(0.014)	(0.021)	(0.011)	(0.031)

Table 14: Advanced TC-MTL positioning performance (mean with SD shown beneath in parentheses).

	NER-correct			Chained		
Model	**F1**	**Pr**	**Re**	**F1**	**Pr**	**Re**
RoBERTa	**0.779**	**0.785**	**0.778**	0.595	0.494	**0.758**
	(**0.012**)	(**0.011**)	(**0.011**)	(**0.005**)	(**0.005**)	(**0.011**)
DeBERTa	0.776	0.783	0.771	**0.612**	**0.518**	0.753
	(**0.012**)	(0.012)	(0.014)	(0.014)	(0.022)	(0.013)

Table 15: Advanced TC-MTL positioning performance for NER-correct and chained metrics (mean with SD shown beneath in parentheses).

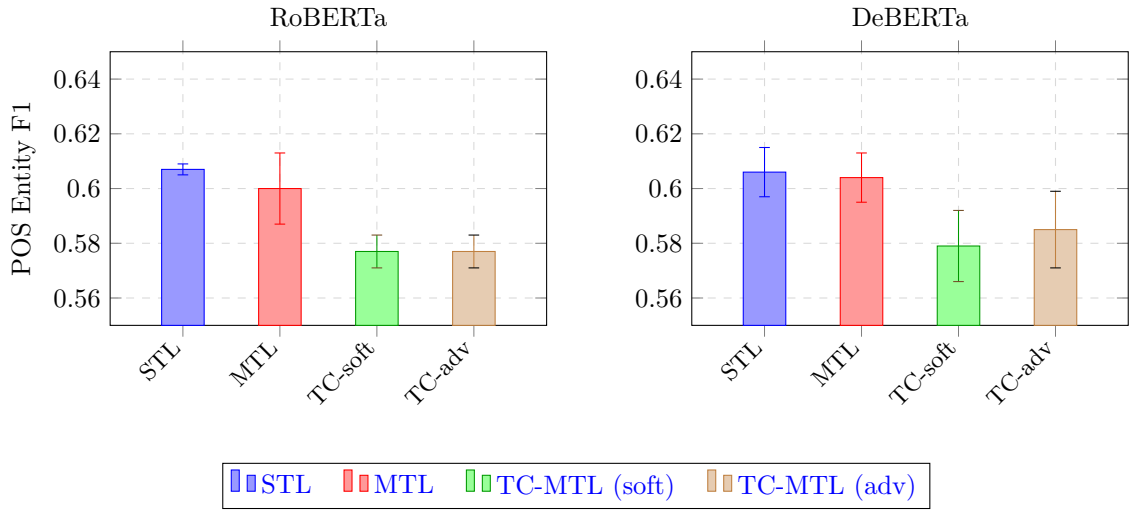


Fig. 3: Positioning Entity F1 scores (mean ± SD) for RoBERTa and DeBERTa across STL, MTL, and TC-MTL configurations.

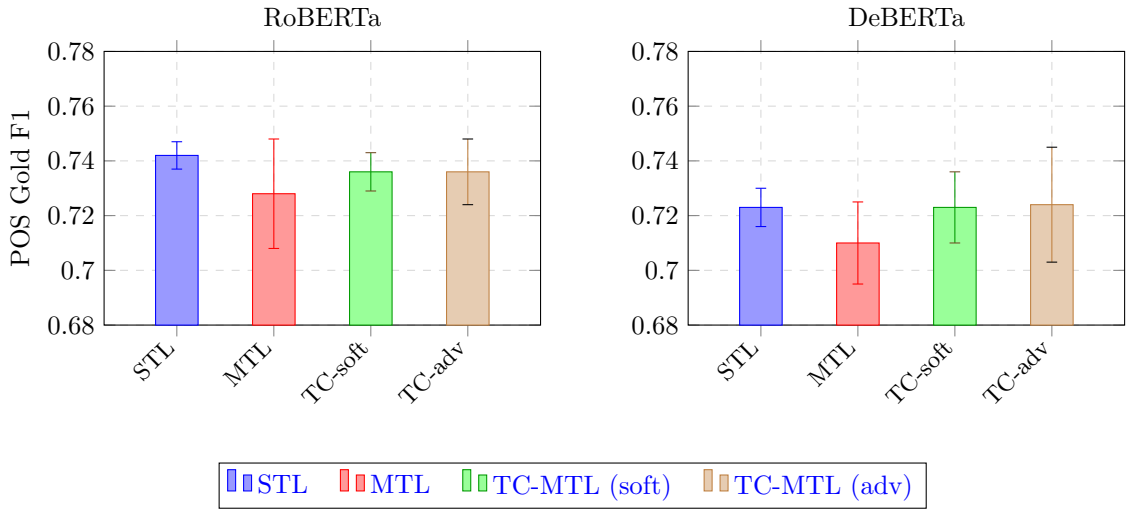


Fig. 4: Positioning Gold F1 scores (mean ± SD) for RoBERTa and DeBERTa across STL, MTL, and TC-MTL configurations.

4.5 SOTA model comparison

Prior work on dependency-aware MTL consistently shows that introducing explicit task structure, through techniques like hierarchical supervision, conditional loss gating, or staged prediction, yields modest improvements over flat STL baselines, typically in the range of 1–3% absolute gain (Table 1). The advanced TC-MTL architecture adopts the same dependency-driven design principle, extending it to a setting in which entity detection and rhetorical positioning are structurally linked, especially when discourse actors are indirect or abstract.

Overall, advanced TC-MTL improves one task substantially (entity extraction) while maintaining stable downstream classification, reshaping the trade-off between accuracy and robustness rather than uniformly improving all metrics. The following examples provide qualitative insight into the system's behaviour, illustrating the model's specific strengths and weaknesses:

Example 1: Implicit blame without explicit sentiment

"They go away free, what's going on with that?"

Predicted: they (NEG)
Gold: they (NEG)

A potential challenging case, marked correctly by the model. "They" is identified as negative, even though the blame is implicit. Sentiment can be misled by "free," and stance doesn't map well because the speaker isn't taking a clear "for/against" position as they're highlighting an outcome and assigning blame.

Example 2: Multi-actor contrast with span truncation

"When they say Biden won, the smart people know they didn't."

Predicted: they (NEG); Biden (NEG); smart people (POS)
Gold: they (NEG); Biden (NEG); the smart people (POS)

The model gets the contrast correct, "they" are negative and "the smart people" are positive, but it drops the word "the." The meaning is correct, but the span isn't perfectly captured.

Example 3: Missed evaluative span and lexical bias

"He is also trying to impose a vicious increase to the estate tax or the death tax."

Predicted: He (NEG); estate tax (NEU); the death tax (NEG)

Gold: He (NEG); a vicious increase (NEG); the estate tax (NEU); the death tax (NEU)

The model finds the main actors but misses the phrase "a vicious increase," which carries the strongest negative judgement. It also labels "the death tax" as negative, likely because of the proximity of the word "death". This shows how missing key phrases or reacting to single words can affect the output.

Study	Model	Primary Task	Mechanism	Gain vs STL
Wang et al. (2021)	BERT-SAN	Aspect Sentiment	Loss gating	+1.81% Acc.
Pei et al. (2022)	BERT-MHSA	Aspect Sentiment	Shared + MHSA	+1.23% Acc.
Zou et al. (2024)	Triple-MRC	Sentiment Triplet	Cascaded	+2.37% F1
Chen et al. (2020)	Causal-MTL	Semantic Parsing	Learned deps.	+1.00% F1
This work	DeBERTa	NER (Entity F1)	Explicit conditioning of positioning on predicted entity spans	**+3.9% F1**
	RoBERTa	NER (Entity F1)		**+1.9% F1**
	DeBERTa	Pos. (Entity F1)		**−1.9% F1**
	RoBERTa	Pos. (Entity F1)		**−2.3% F1**
	DeBERTa	Pos. (Gold F1)		≈ 0.0%
	RoBERTa	Pos. (Gold F1)		≈ 0.0%

Table 16: Comparison of dependency-aware MTL gains relative to STL baselines.

5 Discussion

At its core, this study shows that a model built on just two linked components—entity detection and evaluative positioning—can systematically capture rhetorically complex actors in adversarial political discourse. By tying positioning directly to detected entities at the token level, the framework moves beyond conventional NER, which often misses the abstract, collective, and multi-token references that carry political meaning. Seen through this lens, political discourse becomes easier to read in terms of who is being praised, who is being blamed, and how those evaluations are distributed. These actor–position pairs have the potential to form stable anchors around which network analysis can pivot, allowing clusters of similarly positioned actors to emerge. When combined with techniques such as dependency parsing, the model also makes it possible to trace how actors relate to one another within evaluative structures. This is particularly powerful in adversarial speech: it allows us to see, for example, how "they" is positioned negatively in contrast to "us," and to identify the specific words that construct that contrast.

Given this actor-centred foundation, we then asked whether explicitly modelling the dependency between entity detection and positioning would improve performance in settings where evaluation depends on resolving entity reference first. Rather than assuming that shared representations are sufficient, we tested whether progressively enforcing task order produces more reliable behaviour for entity-conditioned positioning.

Across all experimental settings, the results highlight the importance of modelling the relationship between entity identification and rhetorical evaluation. Positioning performance consistently improves when entity boundaries are fixed and declines when boundaries must be inferred implicitly

(as reflected in the difference between Gold and Entity-level F1). The two tasks also respond differently to explicit conditioning: Advanced TC-MTL produces its strongest and most consistent gains in the upstream NER task, while its impact on positioning is more structural than numerical.

For entity detection, advanced conditioning yields a clear improvement over STL and MTL. NER Entity F1 increases notably for both architectures, with DeBERTa achieving the highest score observed in the study (0.749). This improvement is accompanied by stable convergence across runs, indicating that explicit dependency modelling strengthens the system's primary bottleneck. Given that all downstream components depend on accurate entity boundaries, this represents a meaningful and important gain. It would appear that, because the positioning task shares the encoder with NER, positioning errors back-propagate through the shared representation. When weak entity boundaries make positioning harder, this feedback adjusts the encoder in ways that also improve entity detection. This benefit does not emerge under softmax TC-MTL because probabilistic conditioning allows the positioning task to compensate for imprecise entity boundaries, reducing the back-propagated pressure on the encoder to refine NER representations.

Moving onto the positioning task, it is clear that it responds to task conditioning quite differently than entity detection. Across all regimes, STL, MTL, and softmax TC-MTL, a persistent gap remains between Entity-level positioning F1 (0.56–0.60) and Gold F1 (0.69–0.74). This discrepancy reveals that while the positioning head possesses a strong intrinsic capacity to classify rhetoric, its performance is strictly capped by boundary uncertainty. Essentially, the model is much better at interpreting a stance once an actor is fixed than it is at identifying exactly where that actor's span begins and ends in a sentence. This reveals the complex nature of our proposed NLP task.

Crucially, Advanced TC-MTL preserves these Gold-level scores even as it removes the "oracle" access to perfect entity spans (0.72–0.74). The fact that the positioning logic does not degrade when forced into a stricter hierarchy suggests that the"brain" of the model (its interpretive capacity) is stable. The lower Entity-level scores do not reflect a failure of logic, but rather the compounded difficulty of a system that is no longer allowed to "cheat". In a standalone STL setup, the positioning head is "free-roaming" and likely performs its own implicit entity detection on the side to help locate labels. By enforcing Advanced TC-MTL, we remove this redundancy, forcing the positioning head to rely solely on the spans provided by the NER gatekeeper.

In political discourse analysis, actor identification is the prerequisite for all evaluation. A correct stance assigned to an incorrectly identified actor will not enable the correct analysis; our system prioritises identifying "who" which is in line with our final identification target. As a result, the marginal dip in F1 for positioning may be an acceptable trade-off. The chained positioning scores show the final, realistic measure of this behaviour. While DeBERTa's chained F1 (0.612) and RoBERTa's (0.595) are not as high as their oracle-backed baselines, they offer a more transparent account of end-to-end reliability. DeBERTa, in particular, demonstrates a stronger ability to "cope" with imperfect or noisy entity boundaries.

6 Conclusion and Future Work

This study presents an ML pipeline for identifying both referenced entities and their rhetorical positioning within political discourse. It introduces a conditioned multi-task framework that jointly models entity identification and evaluative framing, demonstrating that the relationship between reference resolution and interpretation is best captured when learned in a dependency-aware architecture rather than in isolation. Across STL, MTL, and TC-MTL configurations, the results show that explicitly structuring this dependency improves entity extraction while preserving the positioning head's classification capacity. The advanced TC-MTL setup further enables chained evaluation, providing a direct view of end-to-end system behaviour under realistic constraints.

Future work will extend this framework beyond the current corpus to examine a broader range of political speeches and contexts. Applying the model to additional speakers, time periods, and different settings will allow for the mapping of larger-scale patterns in how actors are positioned across discourse. This opens the door to tracing shifts in evaluative framing over time, comparing rhetorical strategies across political domains, and integrating the model with complementary NLP techniques to analyse relational and structural dynamics at scale.

6.1 Limitations

Several limitations should be acknowledged: first, although the annotation scheme supports detailed modelling of entities and positioning, the dataset remains relatively small for transformer-based architectures, with approximately 10,000 annotated instances. This constrains the representational breadth the models can learn and increases susceptibility to variance across training runs. Additionally, model performance, particularly in the TC-MTL setting, shows sensitivity to random seeds and training instability, a common challenge when fine-tuning large models on limited data.

Second, the study adopts a single-annotator approach. This is because the task requires specialist, theory-driven evaluations that cannot reliably be distributed to crowd workers or non-experts. The annotation task is complex, and, in this context, Inter-Annotator Agreement (IAA) would not be meaningfully calculated. Instead, the methodology prioritised internal consistency and expert-driven coherence, which are more appropriate validity criteria for highly interpretive annotation tasks of this kind.

References

[1] Beridze, G. (n.d.). USA Political Speeches *Kaggle Dataset* Available at: https://www.kaggle.com/datasets/beridzeg45/usa-political-speeches

[2] Lillelund, C. (n.d.). Donald Trump's Rally Speeches *Kaggle Dataset* Available at: https://www.kaggle.com/datasets/christianlillelund/donald-trumps-rallies

[3] Peutz, S. (n.d.). Misinformation / Fake News Text Dataset (79K) *Kaggle Dataset* Available at: https://www.kaggle.com/datasets/stevenpeutz/misinformation-fake-news-text-dataset-79k

[4] Jörg Friedrichs, Niklas Stoehr, and Giuliano Formisano. Fear-anger contests: Governmental and populist politics of emotion. *Online Social Networks and Media*, 32:100240, December 2022. Elsevier.

[5] S. Jia and H. Yan. Critical Discourse Analysis of Algorithmic Bias Awareness of Social Media Users. *Journal of Modern Information*, 43(6):14–23, 2023.

[6] Alexandra Homolar and Ronny Scholz. The power of Trump-speak: Populist crisis narratives and ontological security. *International Relations*, 33(3):344–364, September 2019. SAGE Publications.

[7] Zuchao Li, Hai Zhao, Junru Zhou, Kevin Parnow, and Shexia He. Dependency and Span, Cross-Style Semantic Role Labeling on PropBank and NomBank. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 21(6):1–16, November 2022.

[8] Dingming Xue, Yang Ding, Kaira Sekiguchi, and Yukio Ohsawa. BERT-MLTKE: A Multi-task Deep Learning Framework for Keyphrase Extraction in Social Media. In *2024 IEEE International Conference on Big Data (BigData)*, pages 7252–7261, Washington, DC, USA, December 2024. IEEE.

[9] Muhammad Naufal Hakim, Syukron Abu Ishaq Alfarozi, and Paulus Insap Santosa. Multi-Task Learning Aspect Based Sentiment Analysis with BERT. In *2024 16th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pages 264–269, Bali, Indonesia, October 2024. IEEE.

[10] Xin Fan and Zhonglin Zhang. A fine-grained sentiment analysis model based on multi-task learning. In *2024 4th International Symposium on Computer Technology and Information Science (ISCTIS)*, pages 157–161, Xi'an, China, July 2024. IEEE.

[11] Yu Pei, Yuqi Wang, Wei Wang, and Jun Qi. Aspect-Based Sentiment Analysis with Multi-Task Learning. In *2022 5th International Conference on Computing and Big Data (ICCBD)*, pages 171–176, Shanghai, China, December 2022. IEEE.

[12] Wang Zou, Wubo Zhang, Wenhuan Wu, and Zhuoyan Tian. A Multi-task Shared Cascade Learning for Aspect Sentiment Triplet Extraction Using BERT-MRC. *Cognitive Computation*, 16(4):1554–1571, July 2024.

[13] Avantika Shrestha, Ml Tlachac, Ricardo Flores, Kevin Hickey, and Elke A. Rundensteiner. Multi-task Learning with Pre-trained Language Models for Mental Illness Screening. In *2024 IEEE International Conference on Big Data (BigData)*, pages 6142–6150, Washington, DC, USA, December 2024. IEEE.

[14] Shahla Gul, Muhammad Asif, Fazal-E Amin, Kashif Saleem, and Muhammad Imran. Advancing Aspect-Based Sentiment Analysis in Course Evaluation: A Multi-Task Learning Framework With Selective Paraphrasing. *IEEE Access*, 13:7764–7779, 2025.

[15] Lingyong Fang, Gongshen Liu, and Ru Zhang. Sense-aware BERT and Multi-task Fine-tuning for Multimodal Sentiment Analysis. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, Padua, Italy, July 2022. IEEE.

[16] Shijie Chen, Yu Zhang, and Qiang Yang. Multi-Task Learning in Natural Language Processing: An Overview. *ACM Computing Surveys*, 56(12):1–32, December 2024.

[17] Nora Alturayeif, Hamzah Luqman, and Moataz Ahmed. Enhancing stance detection through sequential weighted multi-task learning. *Social Network Analysis and Mining*, 14(1):7, December 2023.

[18] Lan Wang, Junjie Peng, Cangzhi Zheng, Tong Zhao, and Li'an Zhu. A cross modal hierarchical fusion multimodal sentiment analysis method based on multi-task learning. *Information Processing & Management*, 61(3):103675, May 2024.

[19] Junjie Peng, Ting Wu, Wenqiang Zhang, Feng Cheng, Shuhua Tan, Fen Yi, and Yansong Huang. A fine-grained modal label-based multi-stage network for multimodal sentiment analysis. *Expert Systems with Applications*, 221:119721, July 2023.

[20] Pau Muñoz, Alejandro Bellogín, Raúl Barba-Rojas, and Fernando Díez. Quantifying polarization in online political discourse: A novel method based on emotional style. *EPJ Data Science* (Preprint), 2025. arXiv:2512.05231.

[21] Hissa F. Alsaif and Hmood D. Aldossari. Review of stance detection for rumor verification in social media. *Engineering Applications of Artificial Intelligence*, 119:105801, March 2023.

[22] Gavin Abercrombie and Riza Batista-Navarro. Policy-focused Stance Detection in Parliamentary Debate Speeches. *Northern European Journal of Language Technology*, 8(1), July 2022.

[23] Yujie Fu, Xiaoli Li, Yang Li, Suge Wang, Deyu Li, Jian Liao, and Jianxing Zheng. Incorporate opinion-towards for stance detection. *Knowledge-Based Systems*, 246:108657, June 2022.

[24] Ruobing Jiang, Mengzhe Liu, Haobing Liu, and Yanwei Yu. Hierarchy-Consistent Learning and Adaptive Loss Balancing for Hierarchical Multi-Label Classification. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 1190–1199, November 2025. arXiv:2508.13452 [cs].

[25] Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and Sentiment in Tweets, May 2016. arXiv:1605.01655 [cs].

[26] Ruth Wodak. *The Politics of Fear: What Right-Wing Populist Discourses Mean.* SAGE Publications Ltd, 55 City Road, London, February 2026.

[27] Michalinos Zembylas. Interrogating the Affective Politics of White Victimhood and Resentment in Times of Demagoguery: The Risks for Civics Education. *Studies in Philosophy and Education*, 40(4):437–455, July 2021. Springer.

[28] Teun A. van Dijk. Discourse and Ideology. In Teun A. van Dijk, editor, *Discourse Studies: A Multidisciplinary Introduction*, pages 379–407. SAGE Publications Ltd, 2nd edition, 2011.