Syntactic Patterns in Political Speech: Dependency-Based Ad Hominem Detection in Bulgarian Social Media

Nevena Grigorova
Gate Institute
Sofia, Bulgaria
nevena.grigorova@gate-ai.eu

Stanislav Penkov
Gate Institute
Sofia, Bulgaria
stanislav.penkov@gate-ai.eu

Keith Kiely
Gate Institute
Sofia, Bulgaria
keith.kiely@gate-ai.eu

Dobromir Tsolyov
Gate Institute
Sofia, Bulgaria
dobromir.tsolyov@gate-ai.eu

Iglika Ivanova
Gate Institute, Sofia University
Sofia, Bulgaria
iglika.ivanova@gate-ai.eu

Abstract

This paper explores the automatic detection of ad hominem attacks in Bulgarian political social media discourse by leveraging dependency-based structural features. Based on a linguistic analysis of ad hominem constructions, we derive structural patterns using Universal Dependencies relations. Grounded in a corpus of Bulgarian political discourse annotated for logical fallacies, we compare three classification settings: structural features alone, lexical features based on TF–IDF representations, and a combined model integrating both. Experiments with logistic regression, decision trees, and support vector machines show that dependency-based structural features achieve competitive detection performance and provide complementary signal to lexical TF–IDF baselines, with performance estimates sensitive to evaluation protocol in this small-data setting. Overall, ad hominem discourse exhibits identifiable syntactic configurations that can complement lexical cues in its automatic detection. The study demonstrates the value of linguistically informed structural features for analyzing political rhetoric and contributes computational methods for studying argumentative strategies in a low-resource language such as Bulgarian.

Keywords: ad hominem, political speech, social media, NLP methods, dependency parsing

1 Introduction

Research on offensive and manipulative language in social media has become a key focus across multiple disciplines, including media studies and linguistics, highlighting the need for a multidisciplinary approach that integrates methods from computer science, particularly Natural Language Processing (NLP).

Advances in NLP have significantly improved the ability to analyze and generate human language, enabling more sophisticated approaches to detecting problematic discourse. One area of growing interest is the identification of logical fallacies as a tool for fostering critical thinking and limiting the spread of misinformation. In the context of "errors in reasoning that can undermine the validity of an argument" (Tindale, 2007), an important subtype is ad hominem, that is, arguments that target an individual rather than their position. Such attacks are especially prevalent in politically sensitive contexts and often contribute to misleading or manipulative communication.

This study models the syntactic structure of ad hominem attacks in a systematic way using tools from computational linguistics. A defining feature of such arguments is the presence of negative personal targeting, which can be identified through recurring syntactic patterns.

The investigation relies on a linguistic analysis of selected examples, for which UD-style annotations were inspected using UDPipe (Straka, 2018) for parse visualization, and on dataset-wide UD parses produced with Stanza for feature extraction and modelling. This methodology enables the extraction and generalization of structural patterns characteristic of ad hominem discourse.

The experimental evaluation assesses whether these syntactic features provide meaningful signals for detecting such arguments. Improved performance of models incorporating structural information, compared to those relying solely on lexical features, would support the hypothesis

that dependency-based representations encode distinctive and informative patterns for identifying ad hominem attacks.

2 Related Work

The detection of offensive and manipulative language in online platforms has become a major topic in NLP, with surveys highlighting both the societal urgency and the technical complexity of modelling abusive, hateful, and toxic discourse (Schmidt and Wiegand, 2017; Fortuna and Nunes, 2018). Early work in this area largely relied on surface-level lexical features such as dictionaries, bag-of-words, and n-grams, which proved effective for explicit slurs and profanity but struggled with more implicit or context-dependent forms of abuse (Fortuna and Nunes, 2018: pp. 2–4); (Wenjie and Zubiaga, 2021). This limitation has motivated a shift towards richer representations that incorporate syntax, discourse structure, and pragmatic cues in order to better capture the underlying strategies of offensive and manipulative communication. Within this broader line of research, logical fallacies and related manipulations have recently been proposed as a useful lens for understanding how arguments mislead or manipulate audiences, and several projects now aim to operationalise fallacy theory in computational models (Lawrence and Reed, 2019; Cruz et al., 2023).

3 Ad Hominem in Political Speech

Ad hominem attacks occupy a central place in the argumentation-theory literature on manipulation techniques because they directly target an interlocutor's character or credibility rather than addressing the merits of their claims, thus functioning as a powerful mechanism of delegitimisation and polarisation in political debate (Walton, 1998; Tindale, 2007). From an argumentation perspective, ad hominem is typically treated as a violation of the "freedom rule" in pragma-dialectics or as a derailment of the dialogical norms that should govern critical discussion (Walton, 1998: pp. 218–225).

Whereas many formal fallacies are inherently invalid, in political contexts, an ad hominem move is not always fallacious, creating unique hurdles for automated NLP detection. In political and legal contexts, questioning a speaker's credentials, past misconduct, or conflicts of interest can be a "reasonable move" if it bears on their reliability as a source of testimony. An attack only becomes a fallacy when the personal attribute (e.g., a candidate's appearance or a past speeding ticket) is logically irrelevant to the truth of the policy claim they are defending. This distinction between an attack assessing reliability and dismissing a claim entirely is central to Tindale (2007). These properties of ad hominem in political discourse make it significantly harder for computational models to detect than other fallacies.

The conceptual distinction between a fallacy and an insult as a rhetorical strategy is prominently developed by Kišiček (2013) in her analysis of parliamentary discourse. Her work emphasizes that many personal attacks in politics are not errors in reasoning (fallacies) but are pure slanging or rhetorical devices intended to rattle an opponent.

In their study "Breaking Down the Invisible Wall of Informal Fallacies in Online Discussions," Sahai et al. (2021) demonstrated that identifying fallacies like ad hominem is highly context-dependent, and neural models achieve higher accuracy when they can "see" the preceding dialogue rather than analyzing a single comment in isolation. Similarly, Habernal et al. (2018) investigated how much debate context is required for both humans and machine learning systems to accurately recognize an ad hominem.

In political communication, empirical studies show that such attacks are pervasive and can blur the line between legitimate scrutiny of a politician's ethos and purely abusive, diversionary discourse (Kišiček, 2018). These observations have motivated specific computational work on ad hominem detection, treating it as a fine-grained subtype of abusive language that requires modelling both content and argumentative role.

In NLP, ad hominem has been addressed both as part of broader datasets on argumentative quality and as a dedicated classification task. Habernal et al. (2018) introduce resources from online debates that annotate various fallacies and argumentative weaknesses, including ad hominem, enabling supervised learning approaches that go beyond coarse "toxic vs. non-toxic" labels. These studies collectively suggest that identifying ad hominem requires attention to how negative personal targeting is realised in context, not only which words are used.

Parallel developments in propaganda and argumentation mining further support the move towards structurally informed models. Shared

tasks on fine-grained propaganda detection, such as the SemEval-2020 Task 11, treat techniques like name-calling and "loaded language" as structured rhetorical devices and show that systems combining lexical information with contextual and syntactic features achieve strong performance (Da San Martino et al., 2020: pp. 1377–1380). Work on argumentation structure parsing in persuasive essays demonstrates that dependency-based and discourse-level representations can effectively capture relations such as support and attack between argument components (Stab and Gurevych, 2017: pp. 619–623). In a similar spirit, recent fallacy-detection frameworks argue that robust models must account for semantics, discourse structure, pragmatics, and world knowledge, and they often rely on manually annotated corpora where specific fallacy types are marked at the span or argument level (Lawrence and Reed, 2019; Cruz et al., 2023).

Despite these advances, most existing resources and systems focus on English or a small set of high-resource languages, and they are trained on relatively large, domain-specific corpora. Multilingual surveys underline that offensive language detection in low-resource and morphologically rich languages remains challenging, and that leveraging linguistic knowledge and structural features is especially valuable when data is scarce (Fortuna and Nunes, 2018; Wenjie and Zubiaga, 2021). This creates a clear gap for work that combines dependency-based syntactic patterns with traditional lexical features to detect ad hominem attacks in low-resource political settings, such as Bulgarian social media, thereby testing whether the recurring configurations identified in prior studies generalise beyond English and high-resource domains.

4 Data

We use a Bulgarian political social media corpus collected via CrowdTangle and annotated for logical fallacies by six annotators. To ensure label reliability, we restrict the present study to high-agreement items, i.e., texts for which annotators show at least 85% agreement on the ad hominem label (majority-consistent subset), rather than reporting a separate inter-annotator agreement statistic.

For this study, we focus on ad hominem and formulate a binary sentence classification task (1 = `ad_hominem`, 0 = `control`). Positive instances were collected from the annotator sheets by selecting items labeled `ad_hominem` under the agreement threshold and segmenting the corresponding texts into sentences. Negative instances were drawn from the same domain and curated as non-`ad_hominem` controls. The resulting dataset contains 177 sentences in total (53 positives, 124 negatives; positive rate: 29.9%).

All sentences were parsed with Stanza (Qi et al., 2020) using its Bulgarian UD pipeline, aligned with the UD Bulgarian-BTB treebank (Universal Dependencies; Universal Dependencies, GitHub; Simov et al., 2002) for tokenization, POS tagging, lemmatization, and dependency parsing. For each sentence, we retain token-level UD annotations and derive sentence-level structural features from these parses. Evaluation uses the stratified protocols described in Section 6.

5 Linguistic Analysis and Patterns

5.1 Dependency-Based Representation

The present study adopts the Universal Dependencies (UD) framework as its formal basis for syntactic representation, specifically relying on the UD Bulgarian-BTB treebank. All dependency patterns are defined using standard UD relations such as `nsubj`, `amod`, `conj`, `ccomp`, `xcomp`, `obl`, `nmod`, `cop`, and `aux`, consistent with the Universal Dependencies framework and its annotation conventions (Nivre et al., 2020; Kübler et al., 2009).

Because the analysis of political statements containing ad hominem attacks requires modelling relations between words within the sentence, Part-of-Speech (POS) tags alone are insufficient for our purposes. However, POS and related morphological features can support the automatic detection of other markers associated with disinformation and manipulative language, such as the Bulgarian evidential renarrative (Temnikova et al., 2025). In the present study, we include UPOS as additional features while focusing primarily on dependency relations.

Unlike constituency-based approaches, which represent syntax through phrase-level structure, dependency parsing captures direct relations between individual words within a sentence. This makes it particularly suitable for identifying relational patterns associated with ad hominem attacks.

Based on the analyzed data, five recurring

dependency-based configurations were identified. They were selected because they represented the most frequent syntactic realizations of ad hominem attacks in the dataset.

- `amod` → `nsubj` → `root` (qualified subject)
- `nsubj + cop + NEG_PRED` (predicative insult)
- `nsubj` → `root` → `ccomp/xcomp` (attack in complement clause)
- `amod + conj` (coordinated discrediting)
- `nsubj:pass + NEG in obl/nmod` (passive discrediting)

Building on these observations, the identified configurations can be abstracted into generalized dependency templates. Moving from individual annotated instances to such templates enables the systematic formalization of recurring structures and supports their application in automatic detection. Accordingly, these configurations provide a foundation for pattern-based modelling of ad hominem.

5.2 Structural Patterns of Ad Hominem Attacks

The five patterns proposed below are not standard rhetorical categories. Rather, they are dependency-based syntactic realizations of ad hominem, derived from the analyzed data and informed by classical argumentation theory (Walton, 1998) as well as research on abusive language detection (Wenjie and Zubiaga, 2021).

The classification is intended as an operational taxonomy for feature extraction rather than a linguistic typology of argumentation. Consequently, some closely related constructions are represented as separate patterns when they correspond to distinct dependency structures that can be reliably identified and encoded as computational features.

Complement clause constructions often instantiate the same underlying attributive or predicative configurations as matrix clauses. However, within the proposed taxonomy they are represented as separate dependency-based patterns because clausal embedding (e.g., via ccomp or xcomp) introduces distinct dependency relations that are captured by different computational features during dependency parsing and feature extraction.

In the present study, the notion of ad hominem is operationalized to include attacks directed not only at individual persons but also at collective political entities, such as political parties, institutions, and social groups. In political discourse, such entities frequently function as proxies for their members, representatives, or supporters and therefore constitute legitimate targets of personal discrediting strategies. Across the data, the most recurrent configuration involves a negative lexical item targeting a person or group in a syntactic position associated with either the subject or the predicate.

To capture this phenomenon, the analysis introduces a set of higher-level semantic abstractions, including labels such as `NEG_PRED`, `NOUN_person/group`, and `NEG`. These labels are not part of the UD annotation scheme but serve as analytical categories that encode evaluative and referential properties relevant to the identification of ad hominem attacks. In this way, the proposed patterns operate at the interface between syntactic structure and semantic interpretation: UD relations provide the formal structure, while the additional labels capture the evaluative dimension of the utterance. Importantly, the proposed dependency patterns are not assumed to be unique to ad hominem discourse. Each of the identified syntactic configurations may also occur in semantically neutral or non-argumentative contexts. Their relevance lies not in the syntactic structure alone, but in the interaction between structural organization and negative evaluative lexical content targeting a person or social group. Consequently, the patterns should be interpreted as recurring structural realizations of ad hominem attacks rather than as sufficient conditions for their identification.

The analysis does not include automatic coreference resolution. When examples contain pronouns such as *he* or *they*, these are treated only as locally interpretable elements of the selected sentence or as part of the human linguistic interpretation of the example. The computational features extracted for classification do not resolve antecedents across preceding turns or wider discourse context. Consequently, attacks whose target can only be recovered from previous discourse remain outside the scope of the present sentence-level setup.

For the qualitative linguistic analysis, we inspected UD parses for a small set of representative

examples using UDPipe (Straka, 2018) as an annotation and visualization aid. For the quantitative experiments reported below, all dataset sentences were parsed with Stanza, and all structural features were extracted from the Stanza-generated UD annotations.

Attributive Insult (Qualified Subject). In this pattern, a negative adjective or insulting noun modifies a noun denoting a person or group, and the modified noun serves as the clausal subject.

Structure:

```
[ADJ_neg | NOUN_insult] → amod
→ [NOUN_person/group]
[NOUN_person/group] → nsubj →
[VERB_root]
```

Schema:

```
(amod* → nsubj) → root
```

Example:

„Днес обаче **престъпният режим на Радев** надмина своите мутренски предшественици.“ (bg)
*“Today, however, **Radev’s criminal regime** has surpassed its thug-like predecessors.”* (eng)

Predicative Insult. Here, the target person or group is realized as the subject, while the negative evaluation is expressed in the predicate via a copular construction.

Structure:

```
[NOUN_person/group] →
nsubj → [cop] → [ADJ_neg |
NOUN_insult]
```

Schema:

```
nsubj + cop + NEG_PRED
```

Example:

„500 делегати казаха, че не искат **Витанов** да е член на БСП, защото **е хулиган**. . . “ (bg)
*“500 delegates said that they do not want **Vitanov** to be a member of the BSP because **he is a hooligan**. . . ”* (eng)

Attack within Complement Clauses. In this configuration, the negative characterization is embedded in a complement clause rather than expressed directly in the matrix clause.

Structure:

```
[NOUN_person/group] → nsubj →
[VERB_root]
[VERB_root] → ccomp/xcomp →
[PRED]
```

Within `PRED`**:**

```
(amod → NOUN_person/group)
or
(nsubj → cop →
ADJ/NOUN_insult)
```

Schema:

```
nsubj → root → (ccomp |
xcomp)
```

In other words, the negative characterization is realized inside an embedded predicative structure.

Example:

„. . . твърди, че **този министър е некадърен**.“ (bg)
*“. . . claims that **this minister is incompetent**.”* (eng)

Coordinated Discrediting. This pattern uses coordination to stack multiple negative modifiers for the same target, creating an accumulative discrediting effect.

Structure:

```
(amod → NOUN_person)
cc/conj
(amod → NOUN_person)
```

Schema:

```
amod + conj(+)
```

Example:

„**Те са злобни, завистливи и дребни души**.“ (bg)
*“**They are spiteful, envious, and petty people**.”* (eng)

Passive Discrediting. In the passive pattern, the target appears as a passive subject, while the negative evaluation is contributed indirectly by a dependent phrase.

Structure:

```
[NOUN_person] → nsubj:pass →
[VERB_root]
(obl / nmod contains a negative
qualifier)
```

Schema:

```
nsubj:pass + NEG
(obl/nmod/xcomp)
```

Example:
„. . . партията заедно с коалиционните си партньори **да бъдат обявени за антидемократични**“. (bg)
“. . . that the party, together with its coalition partners, ***be declared anti-democratic****.”* (eng)

6 Methodology and Experiments

6.1 Feature Representation and Classification

The linguistic analysis identifies five recurring dependency configurations that realize ad hominem attacks. In the qualitative description, labels such as `ADJ_neg`, `NOUN_insult`, and `NEG_PRED` are used only as explanatory abstractions for evaluative content in representative examples. They are not implemented as manually curated lexical resources in the classification experiments. The structural models therefore do not rely on a hand-built negative lexicon or rule-based matching of insult words. Instead, the constructions are operationalized through dependency-derived features extracted from the Stanza-generated UD parses and represented with three feature groups:

- **UD relation features:** counts and length-normalized ratios of selected dependency relations (e.g., `amod`, `nsubj`, `cop`, `nsubj:pass`, `ccomp/xcomp`, `nmod`, `obl`, `case`, `conj/cc`).
- **UPOS distribution features:** counts and ratios of UPOS tags (e.g., `ADJ`, `NOUN`, `PROPN`, `VERB`, `AUX`, `ADV`, `PRON`, `PUNCT`), capturing coarse grammatical composition.
- **Template-shape indicators:** binary features approximating the structural frames of the five constructions (attributive, predicative, complement-clause embedding, coordination, passive) without lexical constraints, i.e., based on the presence and configuration of the relevant dependency relations rather than membership in a negativity lexicon. This reduces brittleness due to lexicon coverage, morphological variation, and lemma/wordform mismatch, while preserving the linguistic motivation of the templates.

For classification, we compare three feature settings: (i) structural features only, (ii) lexical features using TF–IDF over word unigrams and bigrams, and (iii) a combined representation concatenating TF–IDF and structural features.

We evaluate three model families—Logistic Regression (LR), Decision Trees (DT), and Support Vector Machines (SVM)—to cover both linear and non-linear decision boundaries while keeping the analysis interpretable.

6.2 Evaluation Protocols

Because the dataset is small and class-imbalanced, we report results using two evaluation protocols.

Protocol A: Preliminary single-split evaluation (illustrative). We use a stratified 80/20 train–test split (`stratify=y`) to preserve the original class distribution. Hyperparameters are selected via 5-fold stratified `GridSearchCV` conducted only on the training partition. After selecting the best configuration, the model is retrained on the full training set and evaluated once on the held-out test set. This provides an interpretable snapshot but is sensitive to the particular split. Results from the preliminary single-split protocol are reported in Table 1.

Protocol B: Main repeated nested cross-validation (primary). To reduce variance and avoid optimistic bias from hyperparameter tuning on small data, we use repeated stratified nested cross-validation with an outer evaluation loop (5 folds $\times$ 20 repeats) and an inner model-selection loop (5-fold stratified `GridSearchCV`) applied to each outer training fold, optimizing F1 for the positive class. We report mean $\pm$ standard deviation across outer folds for F1, precision, recall, and ROC–AUC. Our primary performance estimates are reported in Table 2 (mean $\pm$ std across outer folds).

7 Results

Table 1 and Table 2 report results for three feature settings—structural, TF–IDF, and combined—across Logistic Regression (LR), Decision Tree (DT), and Support Vector Machine (SVM). We report F1, precision, recall (positive class), and ROC–AUC.

7.1 Preliminary Single-Split Evaluation (Illustrative)

Under the single stratified 80/20 split (Table 1), structural features appear highly competitive and in several cases outperform TF–IDF in F1. For example, DT and SVM with structural features yield the highest F1 values in this protocol (DT:

Feature set	Model	F1	Prec.	Rec.	ROC-AUC
Structural only	Logistic Regression	0.4390	0.3000	0.8182	0.6400
	Decision Tree	0.5185	0.4375	0.6364	0.6455
	SVC	0.5385	0.4667	0.6364	0.6145
TF–IDF only	Logistic Regression	0.2222	0.2857	0.1818	0.6036
	Decision Tree	0.3158	0.3750	0.2727	0.5364
	SVC	0.3158	0.3750	0.2727	0.6255
Combined	Logistic Regression	0.3810	0.4000	0.3636	0.7200
	Decision Tree	0.3810	0.4000	0.3636	0.5764
	SVC	0.5385	0.4667	0.6364	0.6618

Table 1: Illustrative results from a single stratified 80/20 train–test split (`stratify=y`). Hyperparameters are selected via 5-fold stratified `GridSearchCV` on the training partition only; evaluation is performed once on the held-out test set. Due to small test size, results are split-sensitive; the primary evaluation is reported in Table 2.

F1 = 0.5185; SVM: F1 = 0.5385), while TF–IDF alone is lower (DT/SVM: F1 = 0.3158). For LR, structural features produce markedly higher recall than TF–IDF (0.8182 vs. 0.1818), though at the expense of specificity, consistent with a tendency to flag positives aggressively. In the same setting, the combined representation improves LR ROC–AUC (0.7200) relative to both structural (0.6400) and TF–IDF (0.6036), suggesting that lexical and structural signals can interact beneficially in at least some configurations.

These results support the plausibility that dependency-derived structure captures discriminative information; however, because the test set is small, this protocol is vulnerable to split-dependent variance, and the magnitude of apparent gains should not be treated as definitive.

7.2 Main Evaluation: Repeated Stratified Nested Cross-Validation (Primary)

The primary results are those from repeated stratified nested cross-validation in Table 2, which reduces variance and controls for optimistic bias during hyperparameter tuning. Under this protocol, TF–IDF provides the strongest overall baseline, particularly for Logistic Regression (F1 = 0.548±0.074; ROC–AUC = 0.698±0.093), exceeding the corresponding structural-only LR model (F1 = 0.489±0.099; ROC–AUC = 0.655±0.081). This indicates that, across many train/test partitions, lexical information provides a consistently strong signal in the present dataset.

Structural features remain competitive: across LR/DT/SVM, structural-only F1 lies in the 0.474–0.489 range, with ROC–AUC around 0.63–0.66. The combined representation does not yield a consistent F1 improvement over TF–IDF alone. For LR, combining TF–IDF with structural features produces similar F1 (0.491±0.105) and slightly lower ROC–AUC (0.688±0.088) compared to TF–IDF alone, but it changes the error profile: it increases precision relative to the TF–IDF LR baseline (0.470±0.121 vs. 0.414±0.060) while lowering recall (0.561±0.184 vs. 0.818±0.130). This pattern suggests that structural cues can contribute complementary information, but in this dataset their contribution is expressed more as a precision–recall rebalancing than a clear increase in average F1.

7.3 Interpreting the Discrepancy Between Protocols

A key observation is that the single-split evaluation suggests larger structural advantages than the repeated nested cross-validation. This discrepancy is expected in small datasets: a single held-out test set can, by chance, align well (or poorly) with particular feature types, inflating apparent differences. In contrast, repeated nested cross-validation averages performance over many distinct train/test partitions and thus provides a more reliable estimate of expected generalization. The attenuation of "structural > lexical" under the main protocol therefore indicates that variance from sampling is substantial in this regime, and conclusions should be drawn from the repeated nested results.

7.4 Summary of Findings

Across models, the main evaluation shows that lexical TF–IDF features are a strong and robust baseline for ad hominem detection in Bulgarian political social media sentences. Dependency-derived structural features are competitive and encode syntactic regularities motivated by the linguistic analysis, but they do not consistently outperform lexical features when evaluated under repeated stratified nested cross-validation. The

Feature set	Model	F1	Prec.	Rec.	ROC-AUC
Structural only	Logistic Regression	0.489 ± 0.099	0.435 ± 0.098	0.597 ± 0.182	0.655 ± 0.081
	Decision Tree	0.474 ± 0.140	0.468 ± 0.131	0.502 ± 0.181	0.632 ± 0.098
	SVC	0.478 ± 0.091	0.407 ± 0.085	0.615 ± 0.171	0.639 ± 0.082
TF–IDF only	Logistic Regression	0.548 ± 0.074	0.414 ± 0.060	0.818 ± 0.130	0.698 ± 0.093
	Decision Tree	0.418 ± 0.123	0.454 ± 0.124	0.410 ± 0.165	0.604 ± 0.083
	SVC	0.396 ± 0.157	0.350 ± 0.150	0.570 ± 0.340	0.509 ± 0.222
Combined	Logistic Regression	0.491 ± 0.105	0.470 ± 0.121	0.561 ± 0.184	0.688 ± 0.088
	Decision Tree	0.446 ± 0.124	0.455 ± 0.135	0.454 ± 0.154	0.612 ± 0.084
	SVC	0.477 ± 0.120	0.460 ± 0.132	0.538 ± 0.185	0.674 ± 0.097

Table 2: Primary evaluation: repeated stratified nested cross-validation (outer: 5 folds $\times$ 20 repeats; inner: 5-fold stratified `GridSearchCV` within each outer training fold; mean $\pm$ std over 100 outer folds). Metrics are reported for the positive class (ad hominem).

combined representation yields performance comparable to the best single-view setting and can shift the precision–recall balance in interpretable ways, supporting the view that syntactic structure contributes information that is not purely lexical, even if its benefit is modest in the current dataset.

8 Conclusion and Future Work

This paper investigated whether linguistically motivated dependency-based structure can support ad hominem detection in Bulgarian political social media beyond standard lexical baselines. Based on a linguistic analysis, we operationalised five recurring UD configurations via three structural feature groups (UD relation ratios, UPOS distributions, and template-shape indicators) and compared them to TF–IDF and a combined representation across LR, DT, and SVM classifiers. Under the primary repeated stratified nested cross-validation protocol, TF–IDF provides the strongest and most stable baseline, while dependency-derived structural features remain competitive and capture syntactic regularities consistent with the linguistic patterns. Although combining structural and lexical features does not yield a consistent average F1 improvement, it can shift the precision–recall profile in an interpretable way, indicating that syntactic cues contribute complementary information rather than merely duplicating lexical signal.

Future work will (i) extend the study to larger and more diverse Bulgarian political datasets and move from sentence-only classification to context-aware modelling, (ii) refine the structural representation from coarse relation counts to dependency paths and small subgraph motifs that more directly encode the five constructions, and (iii) incorporate large language models in complementary roles: as data-efficient baselines under the same strict evaluation, as annotation and validation assistants that propose candidate ad hominem spans or subtypes with uncertainty for human review, and as structured explanation models constrained to output target–predicate–frame summaries aligned with UD evidence, enabling more systematic error analysis and more reliable template refinement.

9 Limitations

The experiments operate in a small-data regime (177 sentences; 53 positives), which makes estimates sensitive to sampling; we mitigate this by reporting repeated nested cross-validation, but variance remains inherent in this setting. The task is formulated at sentence level and does not incorporate discourse context, including preceding turns, target coreference, speaker identity, or pragmatic intent, even though ad hominem is often context-dependent. No automatic coreference resolution is performed; therefore, examples involving pronouns are interpreted only within the local sentence-level analysis, and cases where the attacked target is recoverable only from prior discourse are likely to remain difficult for the present models. Some errors are therefore attributable to missing discourse context rather than purely lexical or syntactic ambiguity. Structural features depend on UD parses produced by general Bulgarian parsing models, and social-media style (elliptical syntax, unconventional punctuation, slang) can introduce parse noise that directly affects dependency-derived features. Finally, our template-shape indicators intentionally approximate the five linguistic configurations without enforcing explicit polarity or insult-lexicon constraints; this improves robustness to morphology and vocabulary variation, but it also allows structurally similar non-attacks, which can increase false positives and weaken the one-to-one correspondence between the linguistic patterns and their computational operationalisation.

At the same time, the strength of TF–IDF should not be interpreted as evidence that lexical features alone provide a robust general solution to ad hominem detection. In a small and imbalanced political dataset, TF–IDF may capture dataset-specific lexical regularities, topic markers, named entities, or highly frequent evaluative expressions associated with the positive class. Repeated stratified nested cross-validation reduces split-dependent variance and tuning bias, but it does not eliminate possible lexical or topical bias in the corpus itself. For this reason, the TF–IDF results are treated as a strong in-domain baseline rather than as evidence of reliable generalization to broader political discourse.

Acknowledgments

This research is part of the GATE project funded by the Horizon 2020 WIDESPREAD-2018-2020 TEAMING Phase 2 Programme under grant agreement no. 857155, the Programme "Research, Innovation and Digitalization for Smart Transformation"2021-2027 (PRIDST) under grant agreement no. BG16RFPR002-1.014-0010-C01 and, the project BROD (Bulgarian-Romanian Observatory of Digital Media) funded by the Digital Europe Programme of the European Union under contract number 101226153.

References

Fermín Cruz, José A. Troyano, Fernando Enríquez, and F. Javier Ortega. 2023. Automatic detection and generation of fallacies using large language models based on deep learning. In *Proceedings of the DEFALAC Workshop on Fallacy Detection*, pages 1–10.

Giovanni Da San Martino, Alberto Barrón-Cedeño, and Preslav Nakov. 2020. Semeval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation (SemEval-2020)*, pages 1377–1414.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys*, 51(4):85:1–85:30.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. Before name-calling: Dynamics and triggers of ad hominem fallacies in web argumentation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2018)*, pages 386–396.

Gabrijela Kišiček. 2013. The Political Discourse on Croatia's EU Accession: A Rhetorical Analysis of the Presentation of the European Union among Supporters and Opponents of the EU. In Gabrijela Kišiček and Igor Ž. Žagar, editors, *What Do We Know about the World? Rhetorical and Argumentative Perspectives*, pages 181–199. Educational Research Institute, Ljubljana. Accessed: 2026-04-15.

Gabrijela Kišiček. 2018. Ad hominem arguments in political discourse. In *Proceedings of the Conference on Argumentation and Language*, pages 55–70.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. *Dependency Parsing*. Morgan & Claypool Publishers, San Rafael, CA.

John Lawrence and Chris Reed. 2019. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2020)*, pages 4034–4043, Marseille, France. European Language Resources Association.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Saumya Sahai, Oana Balalau, and Roxana Horincar. 2021. Breaking down the invisible wall of informal fallacies in online discussions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10.

Kiril Simov, Petya Osenova, Milena Slavcheva, Sia Kolkovska, Elisaveta Balabanova, Dimitar Doikoff, Krassimira Ivanova, Alexander Simov, and Milen Kouylekov. 2002. Building a linguistically interpreted corpus of Bulgarian: the BulTreeBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Milan Straka. 2018. Udpipe 2.0 prototype at conll 2018 ud shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207.

Irina Temnikova, Ruslana Margova, Stefan Minkov, Tsvetina Stefanova, Nevena Grigorova, Silvia Gargova, and Venelin Kovatchev. 2025. Automatic Detection of the Bulgarian Evidential Renarrative. *Computational Linguistics in Bulgaria*.

Christopher W. Tindale. 2007. *Fallacies and Argument Appraisal*. Cambridge University Press.

Universal Dependencies. UD Bulgarian BTB. Treebank webpage. Accessed: 2026-04-15.

Universal Dependencies, GitHub. UD_Bulgarian-BTB. GitHub repository. Accessed: 2026-04-15.

Douglas Walton. 1998. *Ad Hominem Arguments*. University of Alabama Press.

Yin Wenjie and Arkaitz Zubiaga. 2021. Towards generalisable hate speech detection: A review on abusive language detection. *PeerJ Computer Science*, 7:e598.