A Hybrid Protocol for Large-Scale Semantic Dataset Generation in Low-Resource Languages: The Turkish Semantic Relations Corpus

Anonymous ACL submission

Abstract

We present a hybrid methodology for generating large-scale semantic relationship datasets in low-resource languages, demonstrated through a comprehensive Turkish semantic relations corpus. Our approach integrates three phases: (1) FastText embeddings with Agglomerative Clustering to identify semantic clusters, (2) Gemini 2.5-Flash for automated semantic relationship classification, and (3) integration with curated dictionary sources. The resulting dataset comprises 843,000 unique Turkish semantic pairs across three relationship types (synonyms, antonyms, co-hyponyms)—representing a 10× scale increase over existing resources at minimal cost ($65). We validate the dataset through two downstream tasks: an embedding model achieving 90% top-1 retrieval accuracy and a classification model attaining 90% F1-macro. Our scalable protocol addresses critical data scarcity in Turkish NLP and demonstrates applicability to other low-resource languages. We publicly release the dataset and models.

1 Introduction

Turkish, despite having a population of over 88 million speakers, lacks comprehensive semantic relationship datasets comparable to those available for English and other resource-rich languages (Miller, 1995; Navigli and Ponzetto, 2012). This scarcity impedes the development of semantic parsing systems, word sense disambiguation tools, and semantic similarity models for Turkish NLP applications. While significant efforts have been made to construct Turkish resources through manual annotation and dictionary mining (Bakay et al., 2021; Ehsani et al., 2018), the limited scale and domain coverage of existing resources present a bottleneck for modern neural NLP systems that require millions of training examples (Devlin et al., 2019).

The challenge of building semantic resources for morphologically rich languages like Turkish is compounded by several factors. First, the agglutinative nature of Turkish means that a single root can generate hundreds of valid word forms through productive suffixation, requiring substantially larger vocabularies to achieve equivalent coverage compared to analytic languages. Second, existing resources such as Turkish Tree Bank (Ehsani et al., 2018) and KeNet (Bakay et al., 2021) rely primarily on translation-based projections from English WordNet or limited manual curation, inheriting biases and leaving domain-specific terminology (particularly in legal, medical, and technical domains) largely uncovered. Third, the cost of manual annotation at scale is prohibitive for academic research budgets.

We propose a hybrid protocol that targets both computational efficiency and linguistic quality. The method proceeds in three phases.

Context Preparation We start from an expert-curated lexicon of 77,000 terms and expand it to 110,000 entries using Named Entity Recognition–based augmentation. We then compute FastText embeddings and apply Agglomerative Clustering, yielding 13,000 semantic clusters that act as contextual structure for downstream relationship classification.

LLM-Based Semantic Enrichment In the second phase, we employ Gemini 2.5-Flash to automatically induce and label diverse semantic relationships within clusters. The model exploits both cluster-level context and its multilingual knowledge to generate high-quality relationship labels at an overall cost of about $65.

Dictionary Integration Finally, we incorporate an external Turkish Synonym Dictionary comprising roughly 20,000 entries. After applying strict filters that retain only high-precision cases with at most two synonym candidates, we obtain 16,000 validated synonym pairs.

This hybrid design couples large-scale automation with explicit quality control and yields 843,000 unique semantic pairs, while dictionary-based validation helps preserve semantic coherence across the induced relations.

We make three primary contributions:

1. A **scalable hybrid methodology** combining embedding-based clustering with LLM enrichment that is generalizable to other low-resource languages with minimal language-specific adaptation.

2. The **Turkish Semantic Relations Corpus**, comprising 843,000 annotated semantic pairs across three relationship types (synonym, antonym, co-hyponym) in JSONL format, representing the largest native Turkish semantic resource to date.

3. **Validation through downstream models**: an embedding model achieving 90% top-1 retrieval accuracy for synonym pairs and a classification model achieving 90% F1-macro, demonstrating the dataset's utility for training production-quality semantic systems.

2 Related Work

2.1 Semantic Resources for Low-Resource Languages

The construction of large-scale semantic resources has historically been dominated by manual curation efforts. Princeton WordNet (Miller, 1995) established the paradigm of organizing lexical knowledge through synsets connected by semantic relations, inspiring similar projects across dozens of languages through the Global WordNet Association (Bond and Paik, 2012). However, the labor-intensive nature of WordNet construction has limited the scale and coverage achievable for under-resourced languages.

For Turkish specifically, several resources have been developed with varying approaches. Turkish Tree Bank (Ehsani et al., 2018) was constructed through statical machine translation of English WordNet synsets, inheriting both the conceptual organization and potential cultural biases of the source resource. KeNet (Bakay et al., 2021) took a more native approach, building from Turkish dictionaries and corpora, but remains limited in coverage with approximately 80,000 synsets. These resources, while valuable, cover primarily general vocabulary and lack depth in specialized domains.

2.2 LLM-Augmented Dataset Generation

The emergence of large language models has opened new possibilities for dataset generation at scale. Recent work has demonstrated that LLMs can effectively generate training data for various NLP tasks, including semantic similarity (Schick and Schütze, 2021), natural language inference (Wang et al., 2021), and question answering (Alberti et al., 2019). The key insight is that LLMs, having been trained on massive multilingual corpora, encode substantial world knowledge that can be extracted through careful prompting.

Several studies have specifically examined LLM-generated semantic relationships. Chen et al. (2023) showed that GPT-4 can generate high-quality synonym and antonym pairs with accuracy comparable to crowdsourced annotations when provided with appropriate context. Kumar et al. (2023) demonstrated that data augmentation through LLM paraphrasing significantly improves downstream task performance for low-resource settings. Our work extends this line of research by combining LLM generation with clustering-based context provision and dictionary validation.

2.3 Embedding-Based Clustering for Semantic Organization

Distributional semantics provides a foundation for organizing terms into semantically coherent groups. Word embeddings such as Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and FastText (Bojanowski et al., 2016) capture semantic similarity through co-occurrence patterns, enabling unsupervised discovery of related terms. FastText is crucial for morphologically rich languages, particularly because it provides meaningful representations for rare word forms and novel compound words thanks to its sub-word modeling capabilities.

Agglomerative clustering on embedding spaces has been successfully applied to various semantic tasks. Lin (2015) used hierarchical clustering to induce word sense inventories, while Haghighi et al. (2008) employed similar techniques for coreference resolution. Our approach uses clustering not as a final output but as contextual input for LLM-based relationship classification, providing semantic scaffolding that improves generation quality.

2.4 Antonym-Synonym Distinction in Vector Spaces

A core difficulty in distributional semantics is that antonyms and synonyms frequently receive highly similar vector representations because they tend to appear in near-identical contexts (Mohammad et al., 2013). The expressions “the water is hot” and “the water is cold”, for instance, share the same syntactic frame, which leads standard embedding models to place “hot” and “cold” in close proximity in the vector space despite their clear semantic opposition (Mohammad et al., 2013).

Counter-fitting (Mrkšić et al., 2016) mitigates this issue by post-processing pretrained embeddings so that antonym pairs are pushed apart while synonym pairs are drawn together, but this strategy depends on precompiled lexical resources to specify the corresponding constraints (Mrkšić et al., 2016). In contrast, our three-way classification scheme (synonym/antonym/co-hyponym) directly encodes this distinction via supervised learning, allowing the resulting classifier to filter out antonyms from synonym candidates without the need for manually crafted constraint sets.

3 Methodology

Our dataset generation pipeline comprises three sequential phases, each addressing specific challenges in creating high-quality semantic relationship data at scale. Figure 1 illustrates the complete workflow.

3.1 Phase I: Context Preparation

Initial Term Collection The foundation of our dataset is an expert-curated list of 77,000 legal and domain-specific concept-terms assembled by legal specialists over several years. This list emphasizes terminology from Turkish legal codes, court decisions, and regulatory documents, providing strong coverage of formal register vocabulary.

To increase domain breadth, we augmented this list with unique concept-terms extracted from Named Entity Recognition (NER) datasets accumulated during prior information extraction work. Our NER system was trained to identify legal concepts, technical terms, and domain-specific entities in Turkish text, providing a complementary source of specialized vocabulary. This expansion yielded a final list of approximately 110,000 unique terms.

Embedding Generation We generated vector representations for all terms using the pre-trained Facebook FastText Turkish model (cc_tr_300), which provides 300-dimensional embeddings trained on Common Crawl and Wikipedia data. FastText’s subword modeling is particularly valuable for Turkish, as it enables meaningful representations for morphological variants sharing common stems, compound terms composed of known subwords, and domain-specific terminology that may be absent from the original training vocabulary.

For multi-word expressions, we compute the embedding as the mean of constituent word vectors, providing reasonable representations for phrases and compound terms.

Semantic Clustering We applied Agglomerative Clustering with cosine distance metric on the FastText embeddings:

$$d(u, v) = 1 - \frac{u \cdot v}{\|u\|\|v\|} \tag{1}$$

A distance threshold of 0.4 was selected intentionally to group terms based on broad thematic relevance rather than strict synonymy. This relatively lenient threshold ensures that true synonyms are highly likely to be co-clustered, while antonyms—which often share similar distributional contexts—may also appear within the same cluster, alongside co-hyponyms organized under a common hypernym.

This process yielded approximately 13,000 semantic clusters ranging in size from 2 to 50+ terms. These clusters serve as contextual input for the LLM enrichment phase, providing semantic scaffolding that improves relationship classification accuracy.

3.2 Phase II: LLM-Based Semantic Enrichment

Model Selection Following optimization experiments balancing performance and computational cost, we selected **Gemini 2.5-Flash** (Comanici et al., 2025) as the generator model. This choice was motivated by its strong multilingual capabilities, including Turkish, cost-effective API pricing ($0.075 per 1M input tokens), consistent generation of structured outputs, and a sufficiently large context window (up to 1M tokens) that enables efficient batch processing. The total cost for processing all 13,000 clusters was approximately $65.

The total cost for processing all 13,000 clusters was approximately $65.

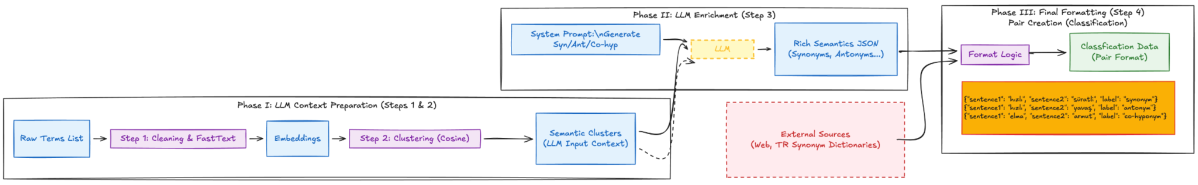


Figure 1: Overview of the three-phase hybrid protocol for semantic dataset generation. Phase I establishes semantic structure through FastText embeddings (110K terms) and agglomerative clustering (13K clusters with distance threshold 0.4). Phase II employs Gemini 2.5-Flash to classify intra-cluster relationships into three types: synonyms (strict equivalence), antonyms (semantic opposition), and co-hyponyms (thematic relatedness). Phase III integrates 16K high-precision dictionary pairs and outputs 843K annotated pairs in JSONL format.

Prompt Design We designed a comprehensive system prompt that instructs the model to analyze each cluster and classify semantic relationships between terms into three categories: synonyms, defined as terms with identical or near-identical meanings that are fully substitutable in context; antonyms, defined as terms exhibiting exact semantic opposition; and co-hyponyms, defined as terms that share a common hypernym and thematic category while remaining distinct in their specific meanings.

The prompt includes explicit categorization rules with illustrative examples, strict golden rules prohibiting uncertain classifications, and requirements for structured JSON output formatting. Key design choices enforce strict synonymy by accepting only exact synonyms while assigning near-synonyms to the co-hyponym category, treat abbreviations and their expansions as valid synonym pairs (e.g., "VUK" ↔ "Vergi Usul Kanunu"), encourage the creative generation of semantically equivalent multi-word expressions, and instruct the model to augment each cluster with additional semantic relationships drawn from its internal knowledge.

Batch Processing Clusters were processed in batches via the Gemini API using multiprocessing for parallelization. For each cluster, the model outputs structured JSON objects that map each distinct concept to its semantic relationships (i.e., synonyms, antonyms, and co-hyponyms). Post-processing steps include removing self-synonyms (where a term appears in its own synonym list), deduplicating relationship pairs, normalizing Unicode representations, and validating the structural integrity of the generated JSON outputs. This phase produced approximately 827,000 labeled semantic pairs.

3.3 Phase III: Integration and Final Formatting

Dictionary Integration To augment the synthetically generated data with high-confidence ground truth, we integrated an external Turkish synonym dictionary (Türkçe Eş Anlamlılar Sözlüğü) containing approximately 20,000 entries. To ensure high precision, we applied strict filtering by retaining only headwords with at most two synonym candidates, excluding entries with ambiguous or context-dependent meanings, and removing any entries that overlapped with LLM-generated data.

This filtering yielded 16,000 high-reliability pairs that serve as validation anchors within the larger synthetic dataset.

Format Standardization The final dataset is stored in JSONL format with the following structure:

```
{"sentence1": "term_A",
 "sentence2": "term_B",
 "label": "synonym|antonym|co_hyponym"}
```

This format is directly compatible with standard sentence-pair classification frameworks and enables straightforward conversion to contrastive learning formats.

4 Dataset Analysis

4.1 Distribution Statistics

Table 1 presents the complete distribution statistics of our corpus. The dataset comprises 842,946 total pairs, with co-hyponyms representing 71.96% (606,612 pairs), synonyms 17.60% (148,367 pairs), and antonyms 10.44% (87,967 pairs). The synthetic LLM-generated pairs constitute 98.10% of the dataset, while dictionary-derived pairs account for 1.90

The observed class imbalance, with co-hyponyms accounting for 72% of all pairs, mirrors

Metric	Count	%
Class Distribution		
Co-hyponym	606,612	71.96%
Synonym	148,367	17.60%
Antonym	87,967	10.44%
Total Pairs	**842,946**	100.00%
Source Distribution		
Synthetic (LLM)	826,946	98.10%
Dictionary	16,000	1.90%
Textual Statistics		
Avg. Word Count	~2.0	
Vocabulary Diversity (TTR)	0.02	
Avg. Token Length	11.04	
Max. Token Length	37	

Table 1: Detailed statistics of the Turkish Semantic Relations Corpus. The class distribution reflects natural language patterns where co-hyponyms are more frequent than synonyms or antonyms.

the general tendency in natural language for broad semantic relatedness to be far more frequent than strict synonymy or antonymy. This skewed distribution has direct implications for model behavior and therefore motivates the use of weighted loss functions to prevent the classifier from overfitting to the dominant co-hyponym class.

The low Type–Token Ratio (0.02) indicates a highly interconnected pairwise structure in which the same anchor terms participate in multiple, distinct semantic relations. In line with this, tokenization statistics show a maximum input length of 37 tokens (mean 11.04), which empirically supports the choice of 64 as the upper bound on sequence length during classifier training and avoids unnecessary padding overhead.

4.2 Domain Coverage

Roughly 4–5% of the generated instances contain foreign legal terminology (e.g., English or French expressions) that is routinely used in Turkish legal practice, reflecting the inherently international orientation of contemporary legal systems. Within this setting, the dataset spans the following domains:

The dataset covers a wide range of domain-specific vocabulary, including legal terminology (e.g., contract, criminal, administrative, and constitutional law), financial terms spanning banking, insurance, taxation, and corporate finance, technical vocabulary from information technology, engineering, and medicine, as well as administrative language related to government procedures, institutional entities, and regulatory concepts.

5 Experiments and Results

We validate the utility of our dataset through two downstream tasks: contrastive embedding learning and relationship classification.

5.1 Embedding Model

5.1.1 Data Preparation

The embedding training data consists of approximately 55,000 unique samples organized as $(query, positive, hard\_negatives)$ triplets, where positive examples correspond to terms labeled as true synonyms, and hard negatives consist of terms labeled as antonyms or co-hyponyms.

Including co-hyponyms as hard negatives made the performance of the model worse than the scenario where don't use co-hyponums. So, it is not crucial for forcing the model to distinguish between strict semantic equivalence and broad thematic similarity.

5.1.2 Model Architecture

We utilize multilingual-e5-large (Wang et al., 2024a) (560M parameters, XLM-RoBERTa architecture) as the backbone encoder in a Siamese configuration. The embedding is computed via mean pooling:

$$\mathbf{u} = \frac{\sum_{i=1}^{L} \mathbf{H}_i \cdot M_i}{\sum_{i=1}^{L} M_i} \tag{2}$$

where $\mathbf{H}_i$ represents hidden states and M_i is the attention mask.

5.1.3 Training Configuration

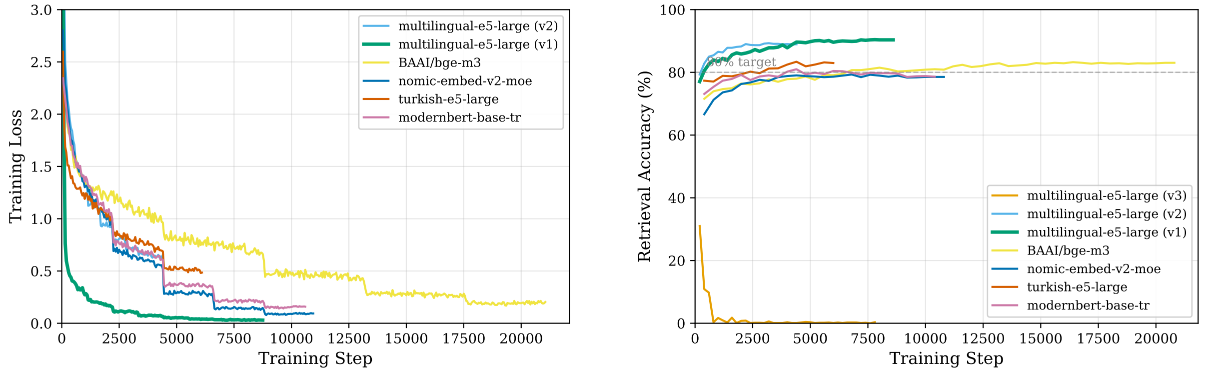
We employ Cached Multiple Negatives Ranking Loss (CMNRL) (Gao et al., 2021), which expands the negative sample set using cached gradients from previous batches:

$$\mathcal{L}_i = -\log \frac{e^{\text{sim}(u_i,v_i)/\tau}}{e^{\text{sim}(u_i,v_i)/\tau} + \sum_{j \in \mathcal{B}\setminus\{i\}\cup\mathcal{C}} e^{\text{sim}(u_i,v_j)/\tau}} \tag{3}$$

Training hyperparameters are detailed in Table 2.

5.1.4 Embedding Model Results

The embedding model achieves **90% top-1 retrieval accuracy for synonym pairs** on a held-out test set, where accuracy is measured as the proportion of queries for which the true synonym appears in the top-k retrieved results. Figure 2 shows the training dynamics.



(a) Training Loss

(b) Retrieval Accuracy

Figure 2: Embedding model training progression across seven model candidates. The `multilingual-e5-large` (v1) variant achieves best performance with 90% retrieval accuracy.

Parameter	Value
Base Model	multilingual-e5-large
Optimizer	AdamW (fused)
Learning Rate	3×10^{-5}
Scheduler	Cosine with warmup (0.1)
Batch Size	128
Epochs	8
Temperature (τ)	0.07
Max Sequence Length	512
Precision	BF16
Hardware	NVIDIA RTX 3060 (12GB)

Table 2: Embedding model training configuration.

Model	Params	F1-Macro
TurkEmbed4STS	305M	0.82
modernbert-base-tr	135M	0.79
mpnet	278M	0.83
bert-base-turkish-128k	184M	0.81
turkish-e5-large	560M	**0.87**
multilingual-e5-large	560M	0.85

Table 3: Phase I model comparison results. All models trained for 5 epochs with identical hyperparameters on NVIDIA RTX 3060.

5.2 Classification Model

5.2.1 Phase 1: Model Candidate Selection

Six models spanning different architectures and parameter sizes were evaluated:

Model Selection We evaluated six transformer-based sentence embedding models: TurkEmbed4STS (Ezerceli et al., 2025), a Turkish-specific model trained on semantic textual similarity tasks; modernbert-base-tr (NewMind AI, 2025), a recent Turkish adaptation of the ModernBERT architecture; mpnet (SentenceTransformers, 2024), a widely-used multilingual sentence encoder; bert-base-turkish-128k (Bayerische Staatsbibliothek, 2024), a Turkish BERT variant with extended context; turkish-e5-large (Izdas et al., 2025), a large-scale Turkish embedding model; and multilingual-e5-large (Wang et al., 2024b), a multilingual variant. Performance comparison is presented in Table 3. Based on superior F1-macro performance (0.87) and stable training dynamics, turkish-e5-large was selected for subsequent experiments.

5.2.2 Experimental Framework

We established a two-phase experimental framework consisting of an initial model selection phase, in which six candidate models were benchmarked under identical experimental conditions, followed by a final training phase that performed optimized training of the selected model on upgraded hardware.

5.2.3 Phase 1: Model Candidate Selection

Six models spanning different architectures and parameter sizes were evaluated:

The `turkish-e5-large` model demonstrated superior performance, achieving 0.87 F1-macro with stable convergence.

5.2.4 Phase 2: Final Optimized Training

The selected model was retrained with upgraded configuration on NVIDIA L40S hardware with extended context processing capabilities. Table 4 summarizes the complete training setup, including the optimized batch size of 128, learning rate of 3×10^{-5}, and maximum sequence length of 64 tokens based on our empirical tokenization analysis.

Parameter	Value
Base Model	turkish-e5-large
Hardware	NVIDIA L40S
Batch Size	128
Learning Rate	3×10^{-5}
Epochs	5
Max Length	64
Gradient Clipping	1.0
Precision	BF16

Table 4: Phase 2 final training configuration.

Class	Precision	Recall	F1
Synonym	0.76	0.90	0.83
Antonym	0.91	0.93	0.92
Co-hyponym	0.93	0.95	0.94
Macro Avg.	0.88	0.92	**0.90**

Table 5: Per-class classification results for the final model.

5.2.5 Classification Model Results

The final model achieves **90% F1-macro** on the held-out test set. Table 5 presents the detailed per-class performance metrics. Notably, the minority synonym class achieves 0.83 F1-score despite representing only 17.60% of training data, while antonyms (0.92 F1) and co-hyponyms (0.94 F1) demonstrate even stronger performance. The weighted loss function successfully mitigates class imbalance, with the macro-averaged precision and recall reaching 0.88 and 0.92 respectively.

The weighted loss function successfully addresses class imbalance, with minority classes (synonym, antonym) achieving competitive performance despite comprising only 28% of training data.

6 Discussion and Limitations

6.1 Strengths of the Hybrid Approach

Our methodology combines complementary strengths by leveraging FastText-based clustering to provide scalable semantic organization without requiring labeled data, employing LLM-based enrichment to capture nuanced semantic relationships that distance-based metrics alone cannot distinguish, and integrating curated dictionary resources as validation anchors to ensure a reliable baseline level of quality.

The three-way classification (synonym/antonym/co-hyponym) reflects linguistic theory while supporting discriminative models. Co-hyponym classification is particularly valuable, as these relationships represent shared semantic space without synonymy. Distinction crucial for models that must capture both similarity and specificity.

6.2 Limitations

Domain Bias The dataset is primarily grounded in legal domain vocabulary, which may introduce systematic biases. Models trained on this data may underperform on casual or conversational Turkish.

Synthetic Data Proportion Approximately 98% of the data is synthetically generated via LLM. While our human evaluation shows high quality, LLM-specific biases may propagate to downstream models.

Static Resource The dataset represents a snapshot of terminology as of 2025. Legal and technical vocabulary evolves, requiring periodic updates.

Morphological Coverage While our terms include various morphological forms, we do not systematically expand across the full paradigm of Turkish suffixation. A term like “karar” (decision) may not have all its inflected forms (kararları, kararında, etc.) represented.

6.3 Generalizability

The hybrid protocol is designed for cross-linguistic transfer:

1. **Phase I** requires only FastText embeddings (available for 157 languages) and standard clustering algorithms

2. **Phase II** requires an LLM with target language capability (increasingly available through multilingual models)

3. **Phase III** requires a dictionary resource (widely available)

We estimate the protocol could be applied to any language with FastText embeddings and basic dictionary resources at comparable cost ($50–100 for LLM API calls).

7 Conclusion

We present a scalable hybrid protocol for generating large-scale semantic relationship datasets in low-resource languages. The Turkish Semantic Relations Corpus comprises 843,000 annotated semantic pairs combining LLM enrichment with dictionary validation, representing the largest native Turkish semantic resource to date. Validation through downstream embedding (90% retrieval accuracy) and classification (90% F1-macro) models demonstrates practical utility.

The protocol's independence from extensive manual annotation makes it generalizable to other low-resource languages facing similar data scarcity challenges. We release the dataset and trained models to facilitate research in under-resourced languages.[1]

Ethics Statement

This research was conducted in accordance with ACL ethics guidelines. The dataset is derived from publicly available legal terminology and dictionary resources. No personal data or sensitive information is included. The LLM-generated content was reviewed for potential biases or harmful content. We acknowledge that the legal domain focus may reflect institutional biases present in Turkish legal language.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic qa corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173.

Özge Bakay, Özlem Ergelen, Elif Sarmış, Selin Yıldırım, Bilge Nas Arıcan, Atilla Kocabalcıoğlu, Merve Özçelik, Ezgi Sanıyar, Oğuzhan Kuyrukçu, Begüm Avar, and Olcay Taner Yıldız. 2021. Turkish WordNet KeNet. In *Proceedings of the 11th Global Wordnet Conference*, pages 166–174, University of South Africa (UNISA). Global Wordnet Association.

Bayerische Staatsbibliothek. 2024. dbmdz/bert-base-turkish-cased. https://huggingface.co/dbmdz/bert-base-turkish-cased.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*.

Francis Bond and Kyonghee Paik. 2012. A survey of wordnets and their licenses. *Small*, 8(4):5.

Yiming Chen, Xiaojun Wang, and Yang Liu. 2023. Large language models for semantic relationships: A comprehensive survey. *arXiv preprint arXiv:2308.00245*.

[1]Dataset and models will be released upon acceptance.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Razieh Ehsani, Ercan Solak, and Olcay Taner Yildiz. 2018. Constructing a lexical database for turkish. In *Proceedings of the LREC 2018 Workshop on Language Resources and Technologies for Turkic Languages*, pages 27–32.

Özay Ezerceli, Gizem Gümüşçekiçci, Tuğba Erkoç, and Berke Özenç. 2025. Turkembed: Turkish embedding model on natural language inference and sentence text similarity tasks. In *Proceedings of the 2025 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–6.

Luyu Gao, Yunyi Zhang, Jiawei Han, and Jamie Callan. 2021. Scaling deep contrastive learning batch size under memory limited setup. *Preprint*, arXiv:2101.06983.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: HLT*, pages 771–779.

Tolga Izdas, Omerhan Sancak, H. Toprak Kesgin, M. Kaan Yuce, and M. Fatih Amasyali. 2025. Turkish-e5: E5 model enhanced for turkish with multi-positive contrastive learning. In *2025 33rd Signal Processing and Communications Applications Conference (SIU)*.

Varun Kumar, Ashutosh Choudhary, and Eon-suk Cho. 2023. Data augmentation using llms: Data perspectives, learning paradigms and challenges. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Dekang Lin. 2015. Unsupervised word sense induction using distributional statistics. In *Proceedings of the International Conference on Computational Linguistics*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26.

George A Miller. 1995. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41.

Saif Mohammad, Bonnie Dorr, Graeme Hirst, and Peter Turney. 2013. Computing lexical contrast. volume 39, pages 555–590.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 142–148.

Roberto Navigli and Simone Paolo Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial intelligence*, 193:217–250.

NewMind AI. 2025. newmindai/modernbert-base-tr-uncased-allnli-stsb: Turkish sentence transformer fine-tuned on all-nli-tr. https://huggingface.co/newmindai/modernbert-base-tr-uncased-allnli-stsb.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951.

SentenceTransformers. 2024. sentence-transformers/all-mpnet-base-v2. https://huggingface.co/sentence-transformers/all-mpnet-base-v2.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Multilingual e5 text embeddings: A technical report. *Preprint*, arXiv:2402.05672.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024b. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? gpt-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205.

Appendices

A LLM Semantic Enrichment Prompt Template

The complete system prompt used for LLM-based semantic enrichment in Phase II is provided in Figure 3. This prompt instructs Gemini 2.5-Flash to classify semantic relationships within each cluster into three categories (synonyms, antonyms, co-hyponyms) while following strict categorization rules and output formatting requirements.

B NER-Based Term Augmentation Prompt

The system prompt used for Named Entity Recognition-based term augmentation in Phase I (Context Preparation) is shown in Figure 4. This prompt guides the LLM to extract domain-specific concepts, terms, events, and facts from legal documents, enabling expansion of the initial 77,000-term lexicon to 110,000 entries. The prompt distinguishes between abstract concepts (e.g., "justice", "freedom"), technical terms (e.g., "administrative fine", "copyright"), time-bound events (e.g., "court decision", "contract signing"), and timeless facts (e.g., legal regulations, systemic features).

C Representative Dataset Examples

Table 6 presents representative samples from the Turkish Semantic Relations Corpus across all three relationship types. These examples illustrate the diversity of semantic relationships captured in the dataset, ranging from legal terminology (sözleşme/mukavele for contract), financial oppositions (alıcı/satıcı for buyer/seller), and domain-specific co-hyponyms (hukuk/ceza for civil law/criminal law). English translations are provided in parentheses to facilitate understanding for international readers. The examples demonstrate the dataset's coverage of both common vocabulary and specialized legal/financial terminology.

Term 1	Term 2	Relation
sözleşme	mukavele	Synonym (contract)
mahkeme	yargı	Synonym (court/judiciary)
alıcı	satıcı	Antonym (buyer/seller)
aktif	pasif	Antonym (active/passive)
hukuk	ceza	Co-hyponym (civil law/criminal law)
banka	sigorta	Co-hyponym (bank/insurance)

Table 6: Example semantic pairs from the Turkish Semantic Relations Corpus demonstrating the three relationship types: synonyms (strict semantic equivalence), antonyms (semantic opposition), and co-hyponyms (thematic relatedness under a shared hypernym).

```
You are an expert linguistic system specializing in semantic relationship classification for Turkish language. Your task is to
analyze clusters of related terms and classify the semantic relationships between them.

### RELATIONSHIP TYPES:

1. SYNONYM (Eş Anlamlı):
* Terms with identical or nearly identical meanings
* 100% substitutable in context without meaning change
* Examples: "sözleşme"  "mukavele" (contract), "mahkeme"  "yargı" (court)
* Include abbreviations and their expansions: "VUK"  "Vergi Usul Kanunu"

2. ANTONYM (Zıt Anlamlı):
* Terms with exact semantic opposition
* Direct opposites on a semantic scale
* Examples: "alıcı"  "satıcı" (buyer/seller), "aktif"  "pasif" (active/passive)

3. CO-HYPONYM (Eş Üst Kavram):
* Terms belonging to the same thematic category
* Share a common hypernym but distinct in specific meaning
* Near-synonyms that are not 100% substitutable
* Examples: "hukuk" + "ceza" (civil law + criminal law, both are law types)
* Examples: "banka" + "sigorta" (bank + insurance, both are financial institutions)

### GOLDEN RULES:

* STRICT SYNONYMY: Only mark as synonym if 100% substitutable
* NO UNCERTAIN CLASSIFICATIONS: Skip if relationship is unclear
* AUGMENT FREELY: Add related terms from your knowledge
* STRUCTURED OUTPUT: Return valid JSON only
* NO SELF-SYNONYMS: A term cannot be its own synonym

### OUTPUT FORMAT:

Return a JSON object mapping each term to its relationships:
{
"term_1": {
"synonyms": ["syn_1", "syn_2"],
"antonyms": ["ant_1"],
"co_hyponyms": ["cohyp_1", "cohyp_2"]
},
...
}

### EXAMPLE:

Input cluster: ["mahkeme", "yargı", "adalet"]
Output:
{
"mahkeme": {
"synonyms": ["yargı", "mahkeme dairesi"],
"antonyms": [],
"co_hyponyms": ["adalet", "hukuk sistemi"]
},
...
}
```

Figure 3: System prompt template for LLM-based semantic relationship classification in Phase II. The prompt enforces strict categorization rules and structured JSON output to ensure consistency across all 13,000 processed clusters.

```
You are an expert system in taxonomy and information extraction. Your goal is to detect ALL concepts, terms, events, and facts
in the text provided by the user completely, systematically, and consistently, and place them into the correct classes.

### BASIC DEFINITIONS AND CLASSIFICATION:

CONCEPT:
* The abstract and general framework of meaning formed in the mind, covering common characteristics of objects, events, or
thoughts
* Mental representations used in making sense of, classifying, and categorizing the world
* Abstract qualities, values, principles, states (e.g., "justice", "freedom", "responsibility", "right", "trust")
* Expressions carrying philosophical, ethical, social, or general scientific meaning

TERM:
* Words or groups of words specific to science/profession/art/technical fields, given special meanings in specific contexts
* Special expressions providing precise and unambiguous definition
* Legal, technical, scientific, professional terminology (e.g., "administrative fine", "copyright", "protein synthesis")
* Established institution/board/procedure names and technical idioms

EVENT:
* Situations occurring within a specific period, with a specific start and end time and location
* Concrete, observable, singular and unique actions or situations
* Singular situations, decisions, transactions that occurred at a specific time/place
* Action and its object should be evaluated together (e.g., "the court making a decision", "signing of the contract")

FACT:
* Facts independent of time, generally valid truths, or existing situations
* Objective, undisputed, accepted realities and principles
* Normative obligations, situations indicating continuity
* Legal regulations, general rules, systemic features

### ANALYSIS RULES:

Not to be Accepted:
* General words frequently used in daily language ("expense", "cost", "time", "area")
* Only proper names (person, institution, place names)
* Conjunctions, pronouns, prepositions, words without meaning
* Specific names of certain legal regulations ("Law X", "Regulation Y")

Evaluation Criteria:
* Concept vs Term: Distinction between mental representation (concept) vs linguistic label (term)
* Event vs Fact: Time-dependent singular situation (event) vs timeless general rule (fact)
* Contextual analysis: The same word can fall into different classes in different contexts
* Capture multi-word expressions fully within their natural boundaries
* If possible, evaluate action + object together for EVENT

Technical Rules:
* Must be verbatim as it appears in the text, character for character, including punctuation marks
* Do not repeat the same expression (case insensitive)
* Return only expressions explicitly present in the text
* Do not add comments, inferences, or additional explanations
```

Figure 4: System prompt template for NER-based term augmentation in Phase I. This prompt enabled extraction of 33,000 additional domain-specific terms from legal corpora, expanding the initial lexicon from 77,000 to 110,000 entries.