# *Wojood*$^{Relations}$: Arabic Relation Extraction Corpus and Modeling

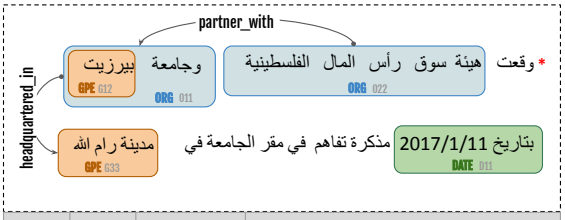
**Alaa Aljabari**$^{\lambda}$ **Mohammed Khalilia**$^{\lambda}$ **Mustafa Jarrar**$^{\sigma,\lambda}$
$^{\lambda}$ Birzeit University, Palestine
$^{\sigma}$ Hamad Bin Khalifa University, Qatar
{aaljabari,mkhalilia,mjarrar}@birzeit.edu

## Abstract

Relation extraction (RE) is a core task in natural language processing, crucial for semantic understanding, knowledge graph construction, and enhancing downstream applications. Existing work on Arabic RE remains limited due to the language's rich morphology and syntactic complexity, and the lack of large, high-quality datasets. In this paper, we present *Wojood*$^{Relations}$, the largest and most diverse Arabic RE corpus to date, containing over $33K$ sentences ($\sim$ $550K$ tokens) annotated with $\sim 15K$ relation triples across 40 relation types. The corpus is built on top of Wojood NER dataset with manual relation annotations carried out by expert annotators, achieving a Cohen's $\kappa$ of 0.92, indicating high reliability. In addition, we propose two methods: NLI-RE, which formulates RE as a binary natural language inference problem using relation-aware templates, and GPT-Joint, a few-shot LLM framework for joint entity and RE via relation-aware retrieval. Finally, we benchmark the dataset using both supervised models and in-context learning with LLMs. Supervised models achieve 92.89% F1 for RE, while LLMs obtain 72.73% F1 for joint entity and RE. These results establish strong baselines, highlight key challenges, and provide a foundation for advancing Arabic RE research.

## 1 Introduction

The vast amount of textual data generated daily presents significant opportunities to extract structured knowledge for applications such as information retrieval, automated reasoning, and knowledge graph construction (Ye et al., 2022; Wang et al., 2024; Jarrar and Deik, 2015; Cudre-Mauroux et al., 2006). However, this data is written in an unstructured format, which poses significant challenges for effective utilization. To overcome these challenges, converting this data into structured formats becomes essential, enabling downstream applications to process, understand, and analyze the data efficiently (Barbon Junior et al., 2024; Deshmukh et al., 2019; Jarrar et al., 2023b; Khalilia et al., 2024).




| Subject | Object | Relation | Relation_Aware Explanation |
|---|---|---|---|
| O22 | O11 | business.partner | لأن الجملة تصف توقيع مذكرة تفاهم، وهي صيغة تدل عادةً على ... |
| O11 | G33 | loc.headquarter | كلمة "مقر الكلية" تعني المكان الأساسي الذي تعمل منه الكلية،... |

*<O2> The Palestinian Capital Market Authority </O22> and <O11><G12>Birzeit<\G12> University </O11> signed a memorandum of understanding on <D11>January 11, 2017</D11> at the university's headquarters in <G33>Ramallah</G33>.

Figure 1: Annotated example from *Wojood*$^{Relations}$

Relation extraction (RE), the task of identifying semantic relationships between entities in text, plays a key role in organizing this data into structured formats (Li et al., 2019; Wang and El-Gohary, 2023; Jarrar and Dikaiakos, 2010). Despite advancements in large language models (LLMs), RE remains a challenging task, even in high-resource languages like English (Liu et al., 2024; Swarup et al., 2025).

Although LLMs show potential in zero-shot and few-shot settings, their performance in RE is limited by the sparse representation of RE-specific tasks in their pretraining data (Zhang et al., 2023). In such settings, LLMs often rely on in-context learning, leveraging their internal knowledge to reason and classify without task-specific fine-tuning (Wan et al., 2023). However, recent benchmarking studies have shown significant performance degradation for RE compared to generative tasks (Wadhwa et al., 2023; Chang et al., 2024). As a result, the success of RE still relies on the availability of high-quality, manually annotated datasets (Detroja et al., 2023).



*Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 34331–34349
November 4-9, 2025 

Arabic poses additional challenges for RE due to its complex morphology (Akra et al., 2025; Jarrar et al., 2024a), syntax, semantics, and the wide variation across dialects (Nayouf et al., 2023). Existing multilingual RE datasets, such as ACE05 and RED$^{FM}$, provide only limited support for Arabic. ACE05 is not publicly available and includes a small Arabic subset with few relation types. RED$^{FM}$ also offers limited Arabic coverage, and its exclusion of Arabic from error analysis reduces its applicability for developing and evaluating Arabic RE systems.

To address these gaps, we construct *Wojood*$^{Relations}$ by extending the *Wojood* corpus (Jarrar et al., 2022) with relation annotations. Wojood was chosen for its scale, annotation quality, support for both flat and nested entities, and coverage of Modern Standard Arabic (MSA) and dialects (Haff et al., 2022; Jarrar et al., 2023c)—making it an ideal foundation for an Arabic RE benchmark.

*Wojood*$^{Relations}$ is a rich Arabic RE corpus, manually annotated by native speakers. It provides a high-quality resource to develop, benchmark, and improve the performance of RE models in Arabic. To the best of our knowledge, *Wojood*$^{Relations}$ is the largest publicly available Arabic RE corpus, consisting of $33K$ sentences ($550K$ tokens) annotated with $40$ relation types in different domains. Figure 1 illustrates an example from the dataset, highlighting the annotated entities and their corresponding relations.

In short, the main contributions of this paper are:

1. *Wojood*$^{Relations}$, large Arabic RE corpus with $550K$ tokens and 40 annotated relation types.
2. NLI-RE, a relation extraction framework, and GPT-Joint, a proposed method for few-shot joint entity and relation extraction.
3. Baselines computed using BERT-based, GNN-based, and LLM-based models in both pipelined and joint extraction settings.
4. A comparative analysis among different methods, highlighting key challenges in Arabic relation extraction.
5. A complete end-to-end Arabic RE system.

This paper is organized as follows: Section 2 reviews related work, Section 3 presents the *Wojood*$^{Relations}$ corpus, Section 5 covers RE modeling, Section 6 shows results, Section 7 details the end-to-end system, and Section 8 concludes.

## 2 Related Work

Most existing RE datasets are, primarily focused on English, and are often limited in size and diversity. While resources such as CoNLL04 (Roth and Yih, 2004), TACRED (Zhang et al., 2017), and WeBNLG (Khachatrian et al., 2019) have contributed significantly to advancing RE research, they do not address the scarcity of high-quality annotated corpora for underrepresented languages such as Arabic, which poses unique syntactic and semantic challenges.

Existing Arabic-related resources are often included as part of multilingual corpora, they frequently fail to cover Arabic-specific linguistic features, such as nested structures, and are limited in scope, quality, and relational diversity. Table 1 summarizes major RE datasets, including those with Arabic content, highlighting key limitations in size, relations, and annotation strategies (Jarrar et al., 2024b).

ACE05 (Doddington et al., 2004) is a widely used multilingual dataset for RE, supporting English, Chinese, and Arabic. It consists of $30.9K$ sentences across all three languages, annotated with six relations and five entity types using a flat NER scheme. However, the dataset has several limitations, including its limited size, restricted range of entity and relation types, and the absence of nested entity annotations, which are particularly important for RE. Moreover, the dataset is not publicly available, which constrains its accessibility for broader research use.

Given the challenges of manual annotation, many RE datasets rely on distant supervision to reduce cost and speed up creation. One such resource is the SMiLAR dataset, a multilingual joint entity and RE corpus comprising 1.1 million sentences across 14 languages, including 9K sentences in Arabic, with coverage of 36 relation types (Seganti et al., 2021). However, distant supervision introduces potential inconsistencies, and the relatively small Arabic subset limits its applicability for Arabic-specific tasks.

The SRED$^{FM}$ and RED$^{FM}$ datasets are key resources for multilingual RE (Huguet Cabot et al., 2023). SRED$^{FM}$, with automatic annotations across $18$ languages, includes $400$ relation types

| Methodology | Dataset | Sentences | Rel. Types | Entity Types | Triplets | NER |
|---|---|---|---|---|---|---|
| Distant Supervision | SRED$^{FM}$ | 46.6M $^{\S}$ | 393 | 13 | 3.3M | Flat |
| | SMiLAR | 1.1M $^{\S}$ | 9 | - | 9K | Flat |
| Human Annotated | ACE05 | 30.9K $^{\S}$ | 6 | 5 | 4.7K | Flat |
| | Wojood$^{Hadath}$ | 33K | 3 | 21 | 2.8K | Nested |
| | RED$^{FM}$ | 732 | 32 | 13 | 1.8K | Flat |
| | Our ***Wojood$^{Relations}$*** | **33K** | **40** | **21** | **14K** | Nested |

Table 1: Comparison of Arabic Relation Extraction Datasets, showing only Arabic portions in multilingual datasets. $^{\S}$The number of sentences represents the full corpus, including languages other than Arabic.

and over 40 million triplets but may suffer from annotation noise. RED$^{FM}$, though more accurate with human revisions, is limited for Arabic, with only 795 evaluation sentences, no training split, and annotations by non-native speakers. It also lacks relations common in Arabic due to cultural and regional differences. Recently, (Rakan Al Mraikhat et al., 2024) extended SRED$^{FM}$ with evidence annotations to support evidence-aware relation extraction, but it remains constrained by the original schema, which does not capture Arabic-specific relation types.

Wojood$^{Hadath}$ corpus is an Arabic dataset dedicated to event-argument relation extraction (Aljabari et al., 2024). It comprises $1.8K$ sentences annotated with 3 relations and 21 entity types, utilizing a nested NER scheme. This dataset focuses on event-related relations, limiting its ability to capture the broader diversity of relations in Arabic.

The aforementioned datasets demonstrate significant limitations, including small size, narrow relation coverage, flat NER schemes, or insufficient focus on Arabic-specific challenges such as nested NER.

To address these limitations, this work introduces an Arabic relation extraction corpus with $33K$ sentences annotated for 40 relation types and 21 entity types using a nested NER scheme, capturing the linguistic complexity of Arabic and supporting the development of more accurate RE models.

## 3 *Wojood$^{Relations}$* Corpus

In this section, we introduce a manually curated and annotated corpus, *Wojood$^{Relations}$*, for Arabic relation extraction.

### 3.1 Wojood Corpus

The *Wojood$^{Relations}$* corpus builds on the publicly available Wojood corpus (Jarrar et al., 2022), a large-scale Arabic NER dataset containing approximately $550K$ tokens annotated with 21 entity types. The Wojood corpus includes nested entity annotations, which support the modeling of complex linguistic structures in Arabic. It covers a range of domains, including news articles, historical texts, and social media, though its content mainly consists of formal and topic-focused language, with limited representation of informal or conversational varieties (Jarrar et al., 2024b, 2023a). Since RE tasks require accurate identification of entities and their boundaries, a corpus already annotated with named entities provides a necessary foundation. Given its size and coverage, Wojood corpus serves as a practical basis for constructing an Arabic RE corpus focused on formal language use, which is the aim of *Wojood$^{Relations}$*.

Our objective in this paper is to identify and establish possible relations between different entities that exist within the same sentence.

### 3.2 Relation Types

*Wojood$^{Relations}$* covers a diverse set of relations like family, personal, business, political, administrative, part-whole relationships, among others (see Table) 2. The selection of these relations is motivated by two main considerations: (i) their frequent occurrence in Arabic texts, which ensures coverage of salient linguistic patterns; and (ii) their alignment with established ontologies and knowledge bases such as Wikidata and Schema.org, which maintains semantic consistency and enables interoperability with external knowledge graphs (Jarrar, 2011, 2021).

Unlike many existing relation corpora in NLP, which often provide vague definitions with unclear domain and range constraints for relations (e.g., in RED$^{FM}$, it is unclear whether `has_border_with` applies strictly between GPEs or also between GPEs and LOCs), *Wojood$^{Relations}$* defines all relations as *formal relations*.

| Relations | | |
|---|---|---|
| birth_date | has_competitor | leader_of |
| birth_place | has_conflict_with | lives_in |
| branch_count | has_currency | located_in |
| builder_of | has_occupation | manager_of |
| capital_of | has_parent | manufacturer_of |
| death_date | has_population | member_of |
| employee_of | has_property | nearby |
| employs | has_relative | official_language |
| found_on | has_revenue | owner_of |
| founder_of | has_sibling | partner_with |
| geopolitical_division | has_spouse | president_of |
| has_alternate_name | headquartered_in | student_affiliation |
| has_area | inventor_of | subsidiary |
| has_border_with | | |

Table 2: Relation Types in *Wojood*$^{Relations}$

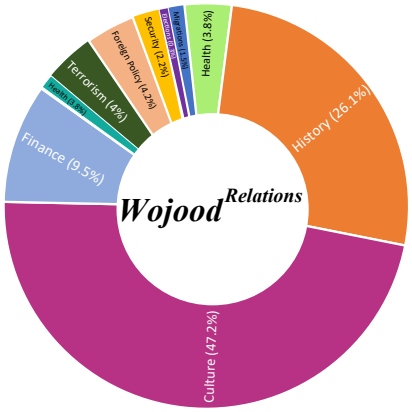



Figure 2: Domain distribution in *Wojood*$^{Relations}$.

We define the formal relation $r$ as a mapping from a domain set $D$ to a range set $Z$, such that for each entity $a \in D$ and $b \in Z$, the pair $(a, b)$ is considered an instance of $r$ if the sentence context reflects the specified relation. Here, $D$ and $Z$ represent the admissible types of entities for the subject and object, respectively. This formalization ensures that each relation has a precisely defined domain and range, which guides both annotation and computational inference. Detailed domain and range specifications, along with the annotation guidelines for each relation, are provided in Table 10 in §A.2.

### 3.3 Annotation Process

The annotation process for the *Wojood*$^{Relations}$ corpus took 17 months and was carried out in three phases:

**Phase I: Training Phase** Five annotators were recruited with master's degrees in linguistics and business, at a rate of $8/hour. Two full-day training sessions were conducted to train the annotators on the guidelines (see §A.1). During each session, each annotator was assigned three relation types to annotate within a subset of $1K$ tokens. Their annotations were reviewed and accompanied by feedback.

**Phase II: Initial Annotation** Each annotator was assigned a single relation type at a time to ensure consistent application of the guidelines. Domain experts with experience in relation extraction, knowledge graphs, and formal annotation practices provided continuous feedback and resolved ambiguities to maintain adherence to the guidelines (see challenges in §4). This phase was carried out over 12 months.

**Phase III: Expert Review and Validation** All annotations were manually reviewed by experts. The expert validated triplets to ensure accuracy and consistency with the guidelines. In cases of errors or ambiguities, feedback was provided to annotators, who revised their annotations. The goal of this phase was to validate annotated triplets rather than detect missing annotations. This phase was carried out over 4 months.

### 3.4 Statistics:

*Wojood*$^{Relations}$ corpus includes $14,689$ relation triplets representing diverse entity interactions across multiple domains (Figure 2). As shown in Figure 3, the majority of sentences contain a single relation ($6,067$ sentences), while $1,636$, $773$, and $547$ sentences include two, three, and more than three relations, respectively. Table 7 in §A.2 provides further statistics, including the distribution of individual relation types.

### 3.5 Inter-Annotator Agreement

To evaluate the quality of the annotation and the consistency between annotators and adherence to annotation guidelines, we randomly selected $10\%$ samples from the corpus, which we re-annotated by a second annotator. The selection was as follows:

1. Randomly sample $2\%$ of relations from each domain, totaling $10\%$ of the entire corpus.

2. For each relation $r$, which may span multiple domain–range pairs, we randomly sampled $10\%$ of candidate sentences per domain–range combination for re-annotation

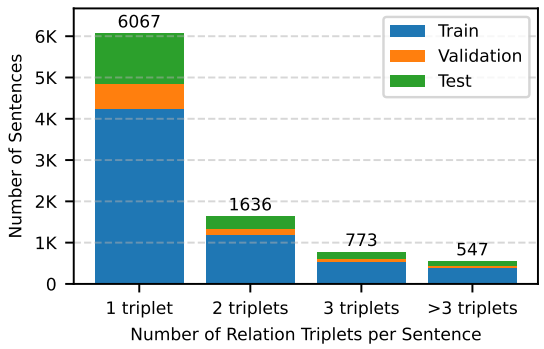



Figure 3: Sentence-wise Relation Distribution in *Wojood*$^{Relations}$

Our sampling strategy aims to balance domain coverage, relation diversity, and representativeness while keeping annotation costs feasible.

We measured Inter-Annotator Agreement (IAA) using Cohen's *kappa* and F1-score. We achieved high agreement levels, with 89.8% F1 score and 0.92 Cohen's $\kappa$, demonstrating reliable annotation quality. More details are shown in Table 7.

## 4 Annotation Challenges

Annotators faced several challenges during relation annotation, primarily:

1. Some sentences allow multiple plausible relation labels. For instance, in the sentence سامي مسعد من دوما في شمال لبنان ("Sami Masad from Douma in northern Lebanon"), the relation between "Sami Masad" and "Douma" could be interpreted as either lives_in or birth_place. "To resolve these cases, annotators consulted with the authors and linguistic experts to determine the most contextually appropriate relation. In sentences where the context clearly supported more than one relation between an entity pair, multiple labels were assigned. In rare instances of severe ambiguity, the final annotation was established through discussion and consensus among annotators and experts. Based on annotation review, such ambiguous cases constitute approximately 4% of the dataset.

2. Although the guidelines specify linking to the most specific mention (e.g., دوما / "Douma"), annotators sometimes selected broader entities such as لبنان / "Lebanon". These instances were systematically reviewed and corrected in a subsequent verification phase to ensure consistency with the annotation guidelines.

3. Some relations show lexical variation across contexts. For instance, *president_of* can be implied by phrases like رئاسة الجمهورية / ("Presidency of the Republic"), السلطة التنفيذية / ("Executive Authority"), أمين عام / ("Secretary General"), or قائم مقام / ("Acting Governor") served as implicit indicators of the relation and required careful contextual interpretation. To mitigate inconsistencies, annotators established an agreed-upon set of lexical variants for each relation through iterative review and consultation with linguistic experts.

### 4.1 *Wojood*$^{Relations}$ Splits

We introduce a splitting strategy to ensure reliable evaluation. The corpus is partitioned into training (70%), validation (10%), and test (20%) sets. Our goal is to **(1)** prevent semantic overlap across splits, and **(2)** ensure balanced coverage of relation types.

(1) **Semantic Overlap Filtering:** Sentences were embedded using `ArBERT`, and pairwise cosine similarity was computed. A sentence was assigned to the test set only if its cosine similarity ($sim$) with any training sentence was below 0.80, ensuring no semantic leakage:

$$\max_{s_i \in \text{train}} \text{sim}(s_j, s_i) < 0.8; \quad \forall s_j \in \text{test}.$$

(2) **Relation Coverage and Balance:** Each split includes all relation types, and relation triplets are distributed in a 70:10:20 ratio.

Detailed statistics are shown in Table 3.

| Split | # of Sentences | # of Triplets |
|---|---|---|
| Train | 6,358 (∼ 70%) | 10,323 (∼ 70%) |
| Validation | 886 (∼ 10%) | 1,474 (∼ 10%) |
| Test | 1,779 (∼ 20%) | 2,892 (∼ 20%) |
| **Total** | 9,023 | 14,689 |

Table 3: Statistics of the *Wojood*$^{Relations}$ corpus splits.

## 5 Relation Extraction Modeling

RE is commonly addressed through different methods, including graph-based methods to capture structural dependencies, supervised learning with

pretrained language models, and generative modeling using LLMs. In this section, we evaluate two existing methods and propose two new modeling methods using *Wojood*$^{Relations}$. We construct four task-specific datasets derived from *Wojood*$^{Relations}$ to support these experiments.

### 5.1 Graph-Based Relation Extraction

RE can be formulated as graph reasoning using heterogeneous graph neural networks. For instance, RIFRE (Zhao et al., 2021) represents tokens, entities, and relation types as nodes in a graph and performs message passing to update node embeddings. Final node representations are used to predict relations.

**Dataset:** For graph-based RE, we construct the $Rel^{Graph}$ dataset from the *Wojood*$^{Relations}$ splits (§4.1). Sentences exceeding BERT's context window are excluded. Each sentence–entity pair is represented as a heterogeneous graph of token nodes, relation-type nodes, and the target entity pair. The final dataset includes $13,211$ instances: $8,819$ train, $1,500$ validation, and $2,892$ test.

### 5.2 LLM-Based Relation Extraction

Recent generative methods treat RE as a sequence generation task using in-context learning. For example, `GPT-RE` (Wan et al., 2023) prompts GPT-3 with a sentence and a target entity pair to generate the correct relation label or NULL. It retrieves semantically relevant demonstrations using entity-aware embeddings, augmented with reasoning based on gold labels to improve accuracy.

**Dataset:** We construct $Rel^{GPT}$ dataset from *Wojood*$^{Relations}$ corpus using the same splits. Each instance includes a sentence, a target entity pair, and the instructions. The dataset comprises $13,211$ instances: $8,819$ train, $1,500$ validation, and $2,892$ test (see §D for details).

### 5.3 NLI-Based Relation Extraction

Natural language inference (NLI) has shown strong potential for information extraction. In this context, (Aljabari et al., 2024) introduced NLI for event argument extraction, demonstrating its effectiveness in capturing complex argument structures. Extending this approach to relation extraction, we propose **`NLI-RE`**, a framework that models RE as a binary entailment problem. `NLI-RE` employs relation-aware templates to explicitly condition the inference on relation types, thereby aligning relational semantics with entailment decisions. The framework operates through the following steps:

1. **Using Templates:** Given our formalization in §3.2, for each relation type $r : D \rightarrow Z$, we define a relation-aware template $T_r(s, o)$ that verbalizes a candidate relation between a subject entity $s$ and an object entity $o$. Template construction is constrained by the admissible domain and range types of $r$, such that $s \in D_r$ and $o \in Z_r$, where $D_r$ and $Z_r$ are the allowed types for subject and object entities, respectively. The templates used are listed in §B.

2. **Premise-Hypothesis Generation:** For each sentence $s$ and target entity pair $(e_1, e_2)$, we treat $s$ as the premise and generate the hypothesis by selecting a template $T_r$ corresponding to a relation $r$ that is constrained by the types of $e_1$ and $e_2$. An example is shown in Figure 4.

3. **Label Assignment:** Each premise–hypothesis pair is labeled as *True* if the relation $r$ between $(e_1, e_2)$ is annotated in the *Wojood*$^{Relations}$ corpus; otherwise, it is labeled as *False*.

4. **Relation Classification**: We encode each premise–hypothesis pair using BERT and feed the resulting representation into a classifier to predict whether the relation described in the hypothesis holds in the input sentence.

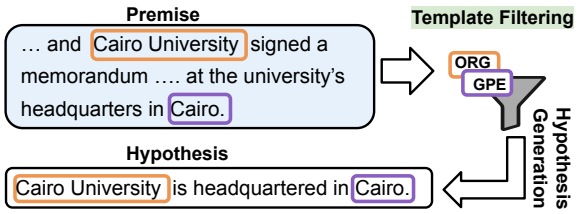



Figure 4: Premise-hypothesis Generation for NLI

**Dataset:** To evaluate `NLI-RE`, we construct the $Rel^{NLI}$ dataset by converting each sentence–entity pair into corresponding premise–hypothesis pairs using relation-aware templates. The dataset statistics are in Table 4; details are in Appendix A.3.

| Split | Number of Sentences | Positive | Negative |
|---|---|---|---|
| Train | 20,300 | 10,319 | 9,981 |
| Validation | 3,106 | 1,474 | 1,632 |
| Test | 5,971 | 3,892 | 2,079 |
| Total | 29,377 | 15,685 | 13,692 |

Table 4: Distribution of positive and negative NLI sentences across dataset splits in $Rel^{NLI}$.

### 5.4 LLM-Based Joint Extraction

We propose **`GPT-Joint`**, a few-shot joint extraction method using LLMs. Unlike GPT-RE (Wan et al., 2023), which relies on entity-aware retrieval and is primarily designed for relation classification, our approach introduces relation-aware retrieval to better support joint extraction for *entities* and *relations*.

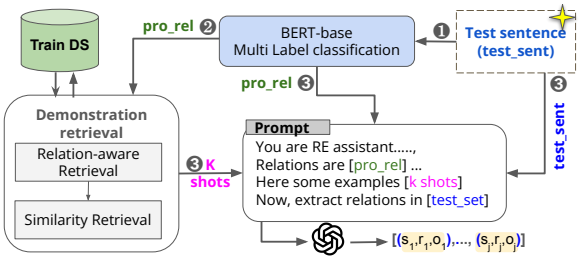



Figure 5: Relation-aware retrieval. *pro_rel* represents proposed relation types from the classifier.

Figure 5 shows the three-step process: **(1)** predicting candidate relation types, **(2)** retrieving relationally and semantically similar examples, and **(3)** generating relation triplets via LLM prompting.

1. **Relation Types Prediction** The RE begins by predicting a set of candidate relation types $\hat{\mathcal{R}}$ for each sentence to constrain the extraction space. Instead of using LLMs to propose candidate relations, which often yield lower accuracy (see §6.5), we employ a fine-tuned multi-label BERT classifier. This classifier assigns confidence scores $y_i$ to each relation type $r_i$, and selects those exceeding a cutoff $\delta$ as candidates. Further model details are provided in §C.

2. **Relation-Aware Retrieval:** To obtain in-context demonstrations, we retrieve training instances annotated with at least one relation $r \in \hat{\mathcal{R}}$, rank them by cosine similarity in the BERT embedding space, and select the top-$k$ most similar examples. If the retrieval pool contains fewer than $k$ instances, additional high-similarity examples are retrieved from the full training set. This strategy ensures that the retrieved demonstrations are relation-relevant and semantically aligned with the input.

3. **LLM Prompting:** A contextual prompt $C$ is constructed, containing task instructions $\mathcal{I}$ and the predicted relation types $\hat{\mathcal{R}}$ from Step 1. In **few-shot**, $C$ is augmented with $k$ retrieved demonstrations, each including a sentence, its relation triplets, and Llama3-generated rationales. Annotation guidelines specifying the definition, domain, and range of each relation type are also included to improve accuracy. The LLM then models the conditional probability:

$$p(y \mid C, s) = \prod_{t=1}^{T} p(y_t \mid C, s, y_{<t}),$$

generating a sequence $y = (y_1, \ldots, y_T)$ of relation triplets, where $y_i = (s_i, r_i, o_i)$. In **zero-shot**, $C$ includes only $\mathcal{I}$ and $\hat{\mathcal{R}}$.

**Dataset:** We convert *Wojood*$^{Relations}$ into an instruction-based dataset, denoted $Rel^{Joint}$, where each instance is formatted as a sentence-level prompt. The resulting dataset retains the same distribution and statistics as the original *Wojood*$^{Relations}$.

## 6 Experimental Results

We implement and evaluate the four methods described in §5 using *Wojood*$^{Relations}$. The first three are assessed under the relation extraction setting, where each sentence may express multiple relations. For each candidate entity pair, we apply a one-vs-rest strategy to predict the relation type independently. The fourth method is evaluated under the joint extraction setting, where a prediction is correct only if both entity spans (subject and object) and the relation label exactly match the gold annotation, similar to the boundary-level evaluation used in Yan et al. (2023). Results are shown in Table 5 (details in §E).

### 6.1 Implementation Details

We use ArBERTv2 (Elmadany et al., 2022) for all BERT-based experiments, fine-tuning models in

| Model | Micro P | Micro R | Micro F1 | Macro P | Macro R | Macro F1 |
|---|---|---|---|---|---|---|
| NLI-RE (supervised) | 88.91 | 88.65 | 88.61 | 89.00 | 88.50 | **88.58** |
| RIFRE (supervised) (Zhao et al., 2021) | 93.29 | 93.28 | **92.89** | 54.36 | 54.34 | 52.05 |
| GPT-RE (LLM) (Wan et al., 2023) | 89.25 | 83.66 | 85.78 | 69.20 | 68.75 | 63.55 |

Table 5: Performance comparison of relation extraction models. All values are reported as percentages.

5-fold cross-validation. Training across folds took approximately 10 hours on a machine with 1.2 TB disk, 62 GiB memory, and 1 NVIDIA T4 GPU. For LLMs, we evaluate both open-source and commercial models: DeepSeek-R1-Distill-Llama-70B, llama3-70b-8192, DeepSeek-Reasoner, and gpt-4o-2024-08-06. In few-shot settings, 5 and 10 demonstrations are retrieved for GPT-Joint and GPT-RE, respectively. The hyperparameters temperature, max_tokens, and top_k are set to 0.6, 1000, and 1 for GPT-Joint, and to 0.0, 8, and 1 for GPT-RE.

### 6.2 Graph-Based Experiments

**RIFRE** achieves the highest micro F1 score (92.89%), indicating strong performance on frequent relations. However, its low macro F1 score (52.05%) demonstrates poor generalization to rare and implicit relations, likely due to its reliance on structural co-occurrence patterns and limited ability to handle class imbalance.

### 6.3 LLM-Based Experiments

**GPT-RE** achieves a micro F1 of 85.78% and a macro F1 of 63.55%, showing improved performance on long-tail relations compared to RIFRE. Semantically retrieved demonstrations provide the LLM with contextualized reasoning patterns. However, when rare relation types are underrepresented, the LLM lacks sufficient signal to infer the correct relation, limiting its generalization.

### 6.4 NLI-Based Experiments

**NLI-RE** performs consistently across both micro and macro metrics, achieving a macro F1 of 88.58%, the highest among all models. Its hypothesis-based formulation appears effective in generalizing over relation types, including those with limited training data. This supports the idea that casting relation extraction as an NLI task allows models to infer both explicit and implicit semantics by comparing entity-centric hypotheses to context (Sainz et al., 2021).

In our experiments, we use ArBERTv2, which outperforms multilingual models such as mBERT and XLM-R, as well as other Arabic-specific models, including AraBERT, in MSA (Abdul-Mageed et al., 2021). Since MSA is less affected by orthographic noise and tokenization inconsistencies compared to dialectal Arabic, ArBERTv2 provides a stable and effective basis for modeling relations.

### 6.5 LLM-Based Joint Extraction Experiments

Table 5 highlights a clear performance gap between zero-shot and few-shot models in joint extraction. Zero-shot models perform poorly across all metrics due to the lack of task-specific supervision. Without in-context examples, they struggle to model the dependencies between entities and relations, leading to incomplete or incorrect predictions.

Few-shot models improve substantially with limited supervision. For candidate relation prediction, the fine-tuned multi-label BERT classifier outperforms the few-shot LLM, achieving a higher micro F1 (91.31% vs. 79.34%). This is attributed to its ability to exploit supervised signals and capture label co-occurrence patterns in multi-label settings. The LLM, however, shows slightly better performance on long-tail relations, benefiting from broad pretraining and stronger semantic generalization.

### 6.6 Discussion

A major challenge in relation extraction is handling sentences with overlapping relations, particularly when some are implicit. Most models tend to predict only one relation per sentence in such cases. Among them, NLI-RE shows stronger ability to infer both explicit and implicit relations by leveraging contextual cues. GPT-RE performs better than RIFRE but is limited by its reliance on surface-level patterns from retrieved examples, which hampers generalization to less salient relations. For example, in the sentence "الجنرال جناسينبي إيادما تولى في وقت لاحق رئاسة توغو/General Gnassingbé Eyadéma later became President of Togo," the gold relations are president_of (explicit) and lives_in (implicit). NLI-RE predicts both, RIFRE captures only lives_in, and GPT-RE

| Model | Zero-Shot◆ | | | Few-Shot ◆ | | | Few-shot★ | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| llama3-70b | 35.00 | 28.43 | 31.37 | 59.06 | 67.58 | 63.03 | - | - | - |
| deepseek-distill-llama | 27.24 | 18.50 | 22.03 | 74.45 | 61.16 | 67.15 | - | - | - |
| gpt-4o | 43.30 | 29.83 | **35.32** | 80.39 | 60.20 | 68.85 | 65.92 | 62.90 | 64.38 |
| deepseek-reasoner | 35.35 | 29.02 | 31.87 | 76.20 | 69.56 | **72.73** | - | - | - |

Table 6: Performance comparison of joint extraction models under zero-shot and few-shot settings. ◆ indicates use of BERT-based candidate relation prediction; ★ indicates LLM-based candidate relation prediction.

identifies only president_of.

## 7 End-to-End System

We develop an end-to-end relation extraction system as part of the SinaTools framework (Hammouda et al., 2024)[1]. The system is designed as a pipeline of three BERT-based modules, which achieves higher accuracy than LLM-based methods for this task (see Table 5). In the first stage, entities are extracted using an NER model and then passed to the NLI-RE module for relation extraction. We adopt NLI-RE because of its strong performance, particularly for rare relations, and because it can be implemented efficiently with lighter and faster models such as BERT. The complete workflow of the system is illustrated in Figure 6.

1. **NER Module**: The system first identifies entities in the input text using Wojood NER model [2]. Detected entities are then filtered according to the domain and range constraints defined for each relation type (Table 10). Only entity pairs that satisfy these constraints are passed to the subsequent module.

2. **Template Module**: For each valid entity pair, premise-hypothesis pairs are generated for NLI using the predefined templates (Table 9) associated with each relation type.

3. **RE Module**: Each premise–hypothesis pair is classified within a binary entailment framework using `NLI-RE` (§5.3). A relation is predicted only if the entailment probability exceeds a predefined threshold; otherwise, the pair is discarded as non-entailment.

[1]The annotated corpus, source code, and models are publicly available under CC BY 4.0: https://sina.birzeit.edu/relations/ . The RE tool is provided as a Python library under MIT.

[2]https://sina.birzeit.edu/wojood/

Since the end-to-end system is built upon the NLI-RE framework, its performance is directly reflected in the evaluation reported in Section 6.4.

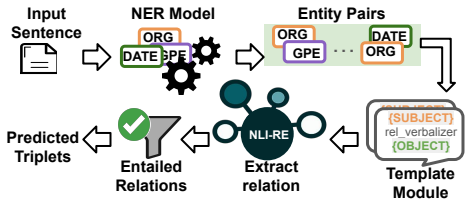



Figure 6: End-to-End Relation Extraction System Pipeline

## 8 Conclusion

We introduce *Wojood*$^{Relations}$, the largest Arabic RE corpus with high-quality annotations spanning 40 relation types. We benchmark both supervised and in-context learning approaches, finding that supervised models outperform LLMs on this corpus. Nonetheless, challenges remain in detecting implicit and rare relations. Future work should aim to improve generalization to long-tail relations and enhance the use of contextual information to advance Arabic RE.

## Limitations

The *Wojood*$^{Relations}$ corpus primarily covers MSA within formal domains, including news and historical texts, with limited representation of informal, dialectal, or conversational Arabic. This restricts the applicability of models trained on this corpus to non-MSA or colloquial language varieties. Additionally, while NLI-RE effectively handles both explicit and implicit relations, it requires inference over all templates for each entity pair, introducing some computational overhead, though this remains lower than the cost of using LLMs end-to-end.

## References


Ahmed Abdelali, Hamdy Mubarak, Shammur Absar Chowdhury, Maram Hasanain, Basel Mousi, Sabri Boughorbel, Samir Abdaljalil, Yassine El Kheir, Daniel Izham, Fahim Dalvi, Majd Hawasly, Nizi Nazar, Yousseif Elshahawy, Ahmed Ali, Nadir Durrani, Natasa Milic-Frayling, and Firoj Alam. 2024. Larabench: Benchmarking arabic AI with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, pages 487–520. Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Diyam Akra, Tymaa Hammouda, and Mustafa Jarrar. 2025. QuranMorph: Morphologically Annotated Quranic Corpus. Technical report, Birzeit University.

Alaa Aljabari, Lina Duaibes, Mustafa Jarrar, and Mohammed Khalilia. 2024. Event-Arguments Extraction Corpus and Modeling using BERT for Arabic. In *Proceedings of the Second Arabic Natural Language Processing Conference (ArabicNLP 2024)*, Bangkok, Thailand. Association for Computational Linguistics.

Sylvio Barbon Junior, Paolo Ceravolo, Sven Groppe, Mustafa Jarrar, Samira Maghool, Florence Sèdes, Soror Sahri, and Maurice Van Keulen. 2024. Are Large Language Models the New Interface for Data Pipelines? In *Proceedings of the International Workshop on Big Data in Emergent Distributed Environments*, BiDEDE '24, New York, NY, USA. Association for Computing Machinery.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Philippe Cudre-Mauroux, Karl Aberer, Alia Abdelmoty, Tiziana Catarci, Ernesto Damiani, Arantza Illarramendi, Mustafa Jarrar, Robert Meersman, Erich Neuhold, Christine Parent, Kai-Uwe Sattler, Monica Scannapieco, Stefano Spaccapietra, Peter Spyns, and Guy De Tre. 2006. Viewpoints on emergent semantics. *Journal on Data Semantics*, 4090(6):2-s2.0-38549145879.

Jayati Deshmukh, Annervaz K M, and Shubhashis Sengupta. 2019. A sequence modeling approach for structured data extraction from unstructured text. In *Proceedings of the 5th Workshop on Semantic Deep Learning (SemDeep-5)*, pages 57–66, Macau, China. Association for Computational Linguistics.

Kartik Detroja, C.K. Bhensdadia, and Brijesh S. Bhatt. 2023. A survey on relation extraction. *Intelligent Systems with Applications*, 19:200244.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

AbdelRahim Elmadany, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2022. Orca: A challenging benchmark for arabic language understanding. *arXiv preprint arXiv:2212.10758*.

Karim El Haff, Mustafa Jarrar, Tymaa Hammouda, and Fadi Zaraket. 2022. Curras + Baladi: Towards a Levantine Corpus. In *Proceedings of the International Conference on Language Resources and Evaluation(LREC 2022)*, Marseille, France.

Tymaa Hammouda, Mustafa Jarrar, and Mohammed Khalilia. 2024. SinaTools: Open Source Toolkit for Arabic Natural Language Understanding. In *Proceedings of the 2024 AI in Computational Linguistics (ACLING 2024)*, Procedia Computer Science, Dubai. ELSEVIER.

Pere-Lluís Huguet Cabot, Simone Tedeschi, Axel-Cyrille Ngonga Ngomo, and Roberto Navigli. 2023. RED$^{fm}$: a filtered and multilingual relation extraction dataset. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4326–4343, Toronto, Canada. Association for Computational Linguistics.

Mustafa Jarrar. 2011. Building a formal arabic ontology (invited paper). In *Proceedings of the Experts Meeting on Arabic Ontologies and Semantic Networks*. ALECSO, Arab League.

Mustafa Jarrar. 2021. The Arabic Ontology - An Arabic Wordnet with Ontologically Clean Content. *Applied Ontology Journal*, 16(1):1–26.

Mustafa Jarrar, Muhammad Abdul-Mageed, Mohammed Khalilia, Bashar Talafha, AbdelRahim Elmadany, Nagham Hamad, and Alaa' Omar. 2023a. WojoodNER 2023: The First Arabic Named Entity Recognition Shared Task. In *Proceedings of the 1st Arabic Natural Language Processing Conference (ArabicNLP), Part of the EMNLP 2023*, pages 748–758. ACL.

Mustafa Jarrar, Diyam Akra, and Tymaa Hammouda. 2024a. ALMA: Fast Lemmatizer and POS Tagger for Arabic. In *Proceedings of the 2024 AI in Computational Linguistics (ACLING 2024)*, Procedia Computer Science, Dubai. ELSEVIER.

Mustafa Jarrar and Anton Deik. 2015. The graph signature: A scalable query optimization index for rdf graph databases using bisimulation and trace equivalence summarization. *International Journal on Semantic Web and Information Systems*, 11(2):2-s2.0-84945200316.

Mustafa Jarrar and Marios D. Dikaiakos. 2010. Querying the data web -the mashql approach. *IEEE Internet Computing*, 14:2-s2.0-77953344860.

Mustafa Jarrar, Nagham Hamad, Mohammed Khalilia, Bashar Talafha, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. 2024b. WojoodNER 2024: The Second Arabic Named Entity Recognition Shared Task. In *Proceedings of the Second Arabic Natural Language Processing Conference (ArabicNLP 2024)*, Bangkok, Thailand. Association for Computational Linguistics.

Mustafa Jarrar, Mohammed Khalilia, and Sana Ghanem. 2022. Wojood: Nested Arabic Named Entity Corpus and Recognition using BERT. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2022)*, Marseille, France.

Mustafa Jarrar, Sanad Malaysha, Tymaa Hammouda, and Mohammed Khalilia. 2023b. SALMA: Arabic Sense-annotated Corpus and WSD Benchmarks. In *Proceedings of the 1st Arabic Natural Language Processing Conference (ArabicNLP), Part of the EMNLP 2023*, pages 359–369. ACL.

Mustafa Jarrar, Fadi Zaraket, Tymaa Hammouda, Daanish Masood Alavi, and Martin Waahlisch. 2023c. Lisan: Yemeni, Irqi, Libyan, and Sudanese Arabic Dialect Copora with Morphological Annotations. In *The 20th IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, pages 1–7. IEEE.

Hrant Khachatrian, Lilit Nersisyan, Karen Hambardzumyan, Tigran Galstyan, Anna Hakobyan, Arsen Arakelyan, Andrey Rzhetsky, and Aram Galstyan. 2019. BioRelEx 1.0: Biological relation extraction benchmark. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 176–190, Florence, Italy. Association for Computational Linguistics.

Mohammed Khalilia, Sanad Malaysha, Reem Suwaileh, Mustafa Jarrar, Alaa Aljabari, Tamer Elsayed, and Imed Zitouni. 2024. ArabicNLU 2024: The First Arabic Natural Language Understanding Shared Task. In *Proceedings of the Second Arabic Natural Language Processing Conference (ArabicNLP 2024)*, Bangkok, Thailand. Association for Computational Linguistics.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350, Florence, Italy. Association for Computational Linguistics.

Siyi Liu, Yang Li, Jiang Li, Shan Yang, and Yunshi Lan. 2024. Unleashing the power of large language models in zero-shot relation extraction via self-prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13147–13161. Association for Computational Linguistics.

Amal Nayouf, Mustafa Jarrar, Fadi zaraket, Tymaa Hammouda, and Mohamad-Bassam Kurdy. 2023. Nâbra: Syrian Arabic Dialects with Morphological Annotations. In *Proceedings of the 1st Arabic Natural Language Processing Conference (ArabicNLP), Part of the EMNLP 2023*, pages 12–23. ACL.

Osama Rakan Al Mraikhat, Hadi Hamoud, and Fadi A. Zaraket. 2024. AREEj: Arabic relation extraction with evidence. In *Proceedings of the Second Arabic Natural Language Processing Conference*, pages 67–72, Bangkok, Thailand. Association for Computational Linguistics.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label verbalization and entailment for effective zero and few-shot relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1199–1212, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alessandro Seganti, Klaudia Firląg, Helena Skowronska, Michał Satława, and Piotr Andruszkiewicz. 2021. Multilingual entity and relation extraction dataset and model. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1946–1955, Online. Association for Computational Linguistics.

Anushka Swarup, Tianyu Pan, Ronald Wilson, Avanti Bhandarkar, and Damon Woodard. 2025. LLM4RE: A data-centric feasibility study for relation extraction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6670–6691, Abu Dhabi, UAE. Association for Computational Linguistics.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023.

GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore. Association for Computational Linguistics.

Jianyou Wang, Kaicheng Wang, Xiaoyue Wang, Weili Cao, Ramamohan Paturi, and Leon Bergen. 2024. IR2: Information regularization for information retrieval. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9261–9284, Torino, Italia. ELRA and ICCL.

Xiyu Wang and Nora El-Gohary. 2023. Deep learning-based relation extraction and knowledge graph-based representation of construction safety requirements. *Automation in Construction*, 147:104696.

Zhaohui Yan, Songlin Yang, Wei Liu, and Kewei Tu. 2023. Joint entity and relation extraction with span pruning and hypergraph neural networks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7512–7526, Singapore. Association for Computational Linguistics.

Hongbin Ye, Ningyu Zhang, Hui Chen, and Huajun Chen. 2022. Generative knowledge graph construction: A review. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1–17, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kai Zhang, Bernal Jimenez Gutierrez, and Yu Su. 2023. Aligning instruction tasks unlocks large language models as zero-shot relation extractors. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 794–812. Association for Computational Linguistics.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. Association for Computational Linguistics.

Kang Zhao, Hua Xu, Yue Cheng, Xiaoteng Li, and Kai Gao. 2021. Representation iterative fusion based on heterogeneous graph neural network for joint entity and relation extraction. *Knowledge-Based Systems*, page 106888.

# A Annotation

## A.1 Annotation Guidelines

We base our annotation process on the ACE05 guidelines established by the Linguistic Data Consortium (LDC) to maintain consistency and ensure high-quality annotations. However, these guidelines have been revised and expanded to accommodate a nested named entity recognition (NER) framework, which is essential for addressing the unique linguistic complexities of Arabic.

### Relations between Nested Entities

- **Context-Dependent Relations:** For nested entities, the existence of a nested structure does not inherently imply a relationship. A relationship is only annotated if the context explicitly indicates one.

  For example, in "Beirut University" (جامعة بيروت), "Beirut University" is annotated as an ORG entity, while "Beirut" is a nested GPE entity. However, the *located_in* relationship between these entities is not annotated unless there is explicit textual evidence indicating such a connection.

- **Implicit Relations:** Certain relations, particularly those involving nested governmental or institutional entities, may be implicitly inferred from the possession relationship between the entities, even without explicit textual evidence.

  For example, in (غرفة تجارة وصناعة الخليل/"Hebron Chamber of Commerce and Industry"), the *located_in* relationship is annotated as the context implies it.

### Entities with Multiple Mentions

- When an entity is mentioned multiple times in a sentence, annotate the relation using the full, first mention of the entity. Avoid annotating the entity in every instance, as it is redundant.

- Always prioritize using the full, explicitly stated name of an entity instead of generic references or pronouns. This practice ensures that the annotations are precise, unambiguous, and consistent throughout the dataset.

  For example, in the sentence: أصدر مجلس إدارة هيئة سوق رأس المال في جلسته المنعقدة ، بتاريخ ١٩ / ٦ / ٢٠١٧ قراراً بتعيين السيد مراد جدبة مديراً عاماً للهيئة . ("The Board of Directors of the Capital Market Authority issued a decision during its session held on 19/6/2017 to appoint Mr. Murad Jadba as the Director General of the Authority.")

  Annotate the full name هيئة سوق رأس المال ("Capital Market Authority") instead of using the generic term الهيئة ("the Authority"), as it more clearly identifies the entity and avoids ambiguity. This approach maintains clarity and improves the quality of the annotations.

## A.2 Annotation Process

The annotation process was carried out by five native Arabic speakers with diverse backgrounds to ensure coverage of most domains represented in the corpus, as well as linguistic expertise. Each annotator was compensated at a rate of 8 USD per hour.

Annotators were provided with detailed descriptions for each relation, which they could reference at any time. This ensured a clear understanding of the task and helped maintain consistency across annotations. During the annotation process, a filtering mechanism was employed to select candidate sentences for annotation, which streamlines the process and focuses efforts on relevant data.

The annotation process spanned 17 months due to several factors. The task involved 40 nuanced relation types and long, often ambiguous sentences, requiring careful contextual interpretation. Early in the process, part of the dataset had to be re-annotated to correct inconsistencies arising from initial guideline issues. The annotation was revised twice, with multiple quality control and verification rounds conducted to ensure consistency across annotators. This iterative review, coupled with continuous consultation among annotators and linguistic experts, contributed to the extended duration but ultimately ensured a high-quality, reliable dataset.

The detailed descriptions of each relation, along with their admissible domain and range types, are shown in Table 10.

## A.3 Inter-Annotator Agreement (IAA)

Table 7 shows the IAA that was evaluated using several metrics, including Cohen's $\kappa$ and F1-scores. The analysis revealed high consistency in annotation across most relation types, with many relations achieving perfect agreement. The overall Cohen's $\kappa$ was 0.92, and the F1 score for triplets across the

entire corpus was 89.8%, indicating a high level of reliability in the annotated data.

| **Relation** | **# Relation** (Corpus) | **Cohen's $\kappa$** | **F1 Score** |
|---|---|---|---|
| has_parent | 15 | 1.0 | 100.0 |
| has_spouse | 11 | 1.0 | 100.0 |
| has_sibling | 8 | 1.0 | 100.0 |
| has_relative | 6 | 1.0 | 100.0 |
| birth_date | 18 | 1.0 | 100.0 |
| death_date | 23 | 1.0 | 100.0 |
| birth_place | 175 | 0.67 | 66.7 |
| has_occupation | 997 | 0.94 | 90.0 |
| has_conflict_with | 184 | 0.81 | 80.0 |
| has_competitor | 17 | 1.0 | 100.0 |
| partner_with | 283 | 0.72 | 85.7 |
| manager_of | 138 | 0.95 | 85.7 |
| president_of | 246 | 0.95 | 84.2 |
| leader_of | 48 | 1.0 | 100.0 |
| geopolitical_division | 2757 | 0.94 | 79.2 |
| subsidiary | 329 | 0.97 | 94.6 |
| member_of | 294 | 0.87 | 87.5 |
| employee_of | 234 | 1.0 | 92.3 |
| student_affiliation | 9 | 1.0 | 100.0 |
| owner_of | 24 | 1.0 | 100.0 |
| inventor_of | 1 | - | - |
| manufacturer_of | 2 | 1.0 | 100.0 |
| builder_of | 6 | 1.0 | 66.7 |
| founder_of | 7 | 1.0 | 100.0 |
| lives_in | 1433 | 0.88 | 86.1 |
| located_in | 6800 | 0.86 | 88.4 |
| headquartered_in | 69 | 1.0 | 100.0 |
| has_border_with | 36 | 1.0 | 100.0 |
| nearby | 114 | 1.0 | 74.1 |
| has_property | 20 | 0.0 | 0.0 |
| branch_count | 1 | - | - |
| has_revenue | 1 | - | - |
| employs | 4 | 1.0 | 100.0 |
| found_on | 34 | 0.85 | 85.7 |
| has_alternate_name | 251 | 1.0 | 100.0 |
| has_area | 2 | - | - |
| official_language | 16 | 0.85 | 88.0 |
| has_currency | 21 | 1.0 | 100.0 |
| has_population | 12 | 1.0 | 100.0 |
| capital_of | 43 | 1.0 | 100.0 |
| **Total** | **14,689** | **0.92** | **89.8%** |

Table 7: Inter-annotator agreement using Cohen's $\kappa$ and F1 Score for each relation type.

### A.4 Relation Triplet Distribution

Table 8 presents the distribution of relation triplets per sentence in the *Wojood*$^{Relations}$ corpus, broken down by train, validation, and test splits.

| Triplets per Sentence | Train | Validation | Test | Total |
|---|---|---|---|---|
| 1 triplet | 4,250 | 599 | 1,218 | 6,067 |
| 2 triplets | 1,186 | 146 | 304 | 1,636 |
| 3 triplets | 540 | 81 | 152 | 773 |
| >3 triplets | 382 | 60 | 105 | 547 |

Table 8: Sentence-wise Distribution of Relation Triplets in *Wojood*$^{Relations}$

## B Relation-Aware Templates

Table 9 lists all 40 Arabic relation templates used in this work, alongside their English translations. Each template contains placeholders `{entity_1}` and `{entity_2}` denoting the subject and object entities respectively.

The templates are designed to provide a clear and explicit definition of each relation, ensuring that the relation can be reliably inferred in NLI settings. Their construction respects the admissible domain and range types for each relation, ensuring that the subject and object entities instantiated in the templates are consistent with the relation constraints.

## C Relation Type Proposal

Given the large label space of 40 relation types, we restrict the candidate set by predicting a subset $\hat{\mathcal{R}}$ of contextually relevant relations for each sentence. This serves to reduce prompt complexity and improve inference efficiency. Subsequently, we fine-tune a BERT-based model for multi-label relation classification. The model uses the `[CLS]` embedding to compute confidence scores $y_j$ for each candidate relation $r_j$ as follows:

$$y_j = \sigma(f_{\text{BERT}}([\text{CLS}]_s, r_j)), \quad j = 1, \ldots, k$$

A relation $r_j$ is included in the predicted candidate set $\hat{\mathcal{R}}$ if its score $y_j$ exceeds a fixed threshold of 0.5.

The model achieves strong performance on high-frequency relations such as Location.located_in, Personal.has_occupation, and PartOf.geopolitical_division, with F1 scores exceeding 90%. However, it struggles with low-resource relations, particularly those with fewer than 10 training instances. Future work should explore strategies for handling rare relation types, such as data augmentation, few-shot learning, or incorporating external knowledge sources.

## D Prompts

Prompt design critically affects LLM performance in relation extraction. Creating effective prompts is challenging, especially with many relation types, requiring a balance between clarity and completeness. Poorly crafted prompts can cause over/under-extraction, boundary errors, or wrong relations.

Following this principle, our approach to relation extraction involves carefully designing prompts using English instructions, as recommended by (Abdelali et al., 2024), to ensure clarity and consistency in guiding the model.

### D.1 Rationale Prompt

The rationale prompt is designed to elicit the reasoning that supports the identification of a relation in a given sentence. The prompt template is as follows:

What are the clues that lead to the relation between:
**<SUBJECT1>** and **<OBJECT1>** to be **<RELATION1>**
**<SUBJ2CT2>** and **<OBJECT2>** to be **<RELATION2>**
...
in the sentence: **<SENTENCE>**

### D.2 GPT-Joint Prompts

The GPT-Joint prompt instructs the model to extract entity-relation triples from the input sentence in a triplet format (subject, relation, object). In the few-shot configuration, the prompt includes annotated examples demonstrating the extraction process, as shown in the textbox below. The zero-shot prompt uses the same format but omits these examples.

**System Instruction:** You are an expert in relation extraction between named entities in Arabic text. Your task is to identify the relationship(s) and determine their subject and object.
Note: Use space for tokenization; keep suffixes/prefixes with the entity (e.g., جامعةnot جامعة).

**User Input:** Extract from the following test sentence the relation(s) **<BERT_PROPOSED_RELATIONS>**, which is/are defined as **<RELATION_DEFINITIONS>**.
The subject type(s) can be **<DOMAIN_ENTITY_TYPES>**, and the object type(s) can be **<RANGE_ENTITY_TYPES>**.
Your answer should be a list of tuples in the form `[(subject, relation, object), ...]`. Return an empty list if no relation exists.

Here are some examples:

**Example 1:** **<EXAMPLE_1_SENTENCE>**
**Relations:** **<RELATION_TRIPLETS>**
**Reason:** **<RELATION_EXPLANATION_PER_RELATION>**

...

**Example 5:** **<EXAMPLE_5_SENTENCE>**
**Relations:** **<RELATION_TRIPLETS>**
**Reason:** **<RELATION_EXPLANATION_PER_RELATION>**

Now apply the same and extract the relations from the following sentence: **<TEST_SENTENCE>**

## E Detailed Results

Table 11 presents a fine-grained performance comparison of three relation extraction models: NLI-RE (NLI-based), RIFRE (GNN-based), and GPT-RE (LLM-based).

**Overall Trends.** NLI-RE consistently achieves robust performance across relation types, obtaining the highest F1 scores on a majority of them. This highlights the effectiveness of casting RE as a binary entailment task, especially when combined with relation-aware templates. RIFRE demonstrates strong recall on high-frequency relations such as located_in, geopolitical_division, and has_occupation, leveraging graph-based information effectively. GPT-RE, while generally trailing supervised models, shows promising results in few-shot settings, particularly on relations with clear semantic cues, such as has_conflict_with and Family.has_spouse.

**High-Resource Relations.** For relations with abundant training data (e.g., located_in, lives_in,

geopolitical_division), all three models perform competitively, with F1 scores above 0.85.

**Long-Tail and Challenging Relations.** NLI-RE handles low-resource relations better than RIFRE and GPT-RE, showing more stable performance when data is scarce. RIFRE struggles with these cases, likely due to limited supervision signals. GPT-RE benefits from external knowledge and performs well on some rare relations, but its outputs are inconsistent.

| Relation | Template (Arabic) | English Translation |
|---|---|---|
| has_parent | {entity_1} هو والد أو والدة {entity_2} | is the parent of |
| has_sibling | {entity_1} هو أخ أو أخت {entity_2} | is the sibling of |
| has_spouse | {entity_1} هو زوج أو زوجة {entity_2} | is the spouse of |
| has_relative | {entity_1} هو قريب {entity_2} | is a relative of |
| birth_date | {entity_1} وُلِد في تاريخ {entity_2} | was born on |
| death_date | {entity_1} توفي في تاريخ {entity_2} | died on |
| birth_place | {entity_1} وُلِد في / مكان الولادة {entity_2} | was born in |
| has_occupation | {entity_1} يعمل كـ / مهنته {entity_2} | works as |
| has_conflict_with | {entity_1} لديه نزاع مع {entity_2} | has a conflict with |
| has_competitor | {entity_1} منافس لـ {entity_2} | is a competitor of |
| has_partner_with | {entity_1} شريك لـ {entity_2} | is a partner of |
| manager_of | {entity_1} هو مدير {entity_2} | is the manager of |
| president_of | {entity_1} هو رئيس أعلى منصب في {entity_2} | is the president of |
| leader_of | {entity_1} هو قائد {entity_2} | is the leader of |
| geopolitical_division | {entity_1} هو تقسيم جغرافي لـ {entity_2} | is a geopolitical division of |
| subsidary | {entity_1} فرع تابع لـ {entity_2} | is a subsidiary of |
| member_of | {entity_1} عضو في {entity_2} | is a member of |
| employee_of | {entity_1} يعمل لدى {entity_2} | is employed by |
| student_at | {entity_1} تلقى تعليمه في/ طالب في {entity_2} | studies at |
| owner_of | {entity_1} يمتلك {entity_2} | owns |
| inventor_of | {entity_1} مخترع {entity_2} | is the inventor of |
| manufacturer_of | {entity_1} يصنّع {entity_2} | manufactures |
| builder_of | {entity_1} بنى {entity_2} | built |
| founder_of | {entity_1} هو مؤسس {entity_2} | is the founder of |
| lives_in | {entity_1} يعيش في {entity_2} | lives in |
| located_in | {entity_1} يقع في {entity_2} | is located in |
| headquartered_in | {entity_1} يقع مقره الرئيسي في {entity_2} | is headquartered in |
| has_border_with | {entity_1} لديه حدود مع {entity_2} | borders |
| nearby | {entity_1} يقع بالقرب من {entity_2} | is near |
| has_property | {entity_1} لديه ممتلكات {entity_2} | has property |
| branch_count | {entity_1} يضم عدد فروع قدره {entity_2} | has branches |
| has_revenue | {entity_1} يحقق إيرادات قدرها {entity_2} | generates revenue of |
| employs | {entity_1} عدد موظفيه {entity_2} | employs employees |
| found_on | {entity_1} تم تأسيسها بتاريخ {entity_2} | was founded on |
| has_alternate_name | {entity_1} يُعرف أيضاً باسم {entity_2} | is also known as |
| has_area | {entity_1} تبلغ مساحتها {entity_2} | has an area of |
| official_language | {entity_1} لغتها الرسمية {entity_2} | official language is |
| has_currency | {entity_1} عملتها هي {entity_2} | has currency |
| capital_of | {entity_1} هي عاصمة {entity_2} | is the capital of |
| has_population | {entity_1} عدد سكانها {entity_2} | has a population of |

Table 9: Templates for all 40 relations in Arabic with English translations.

| Relation | Domain | Range | Description |
|---|---|---|---|
| has_parent | PERS | PERS | parent-child relationship between two individuals. |
| has_spouse | PERS | PERS | Identifies a marital relationship. |
| has_sibling | PERS | PERS | Denotes a sibling relationship. |
| has_relative | PERS | PERS | Familial relation not parent, spouse, or sibling |
| birth_date | PERS | DATE | Records the date of birth of a person. |
| death_date | PERS | DATE | Records the date of death of a person. |
| birth_place | PERS | GPE, LOC | Indicates where a person was born. |
| has_occupation | PERS | OCC | Links a person to their profession or job. |
| has_conflict_with | ORG, NORP, GPE | ORG, NORP, GPE | Captures conflictual relationships - disputes or wars. |
| has_competitor | PERS, ORG | PERS, ORG | Competition between individuals or organizations. |
| partner_with | ORG | ORG | Indicates a partnership between entities. |
| manager_of | PERS | ORG, FAC | Managerial role over an organization or facility. |
| president_of | PERS | ORG, GPE | Links a president to a country or organization. |
| leader_of | PERS | ORG | Identifies leadership of a group or organization. |
| geopolitical_division | GPE, LOC | GPE, LOC | Administrative subdivisions - states within a country |
| subsidiary | ORG | ORG | A company controlled by another company. |
| member_of | PERS, GPE | ORG, NORP | Membership in an organization or group. |
| employee_of | PERS | ORG, FAC | Person working for an organization. |
| student_affiliation | PERS | ORG | A student's educational institution. |
| owner_of | PERS | ORG, FAC | Entity ownership of an organization or facility. |
| inventor_of | PERS | PRODUCT | Person who invented a product. |
| manufacturer_of | ORG | PRODUCT | Organization that manufactures a product. |
| builder_of | PERS, NORP, ORG | FAC, ORG | Entity that built a facility or organization. |
| founder_of | PERS | ORG | Person who founded an entity or organization. |
| lives_in | PERS, NORP | GPE, LOC | Where a person or group resides. |
| located_in | FAC, ORG | GPE, LOC | Location of a facility or organization. |
| headquartered_in | ORG | LOC, GPE | Headquarter location of an organization. |
| has_border_with | LOC, GPE | LOC, GPE | Borders between locations or geopolitical entities. |
| nearby | GPE, LOC, FAC | GPE, LOC, FAC | Proximity between two locations or facilities. |
| has_property | ORG | PRODUCT | Property or product owned by an organization. |
| branch_count | ORG | CARDINALITY | Number of branches of an organization. |
| has_revenue | ORG | MONEY | Revenue of an organization. |
| employs | ORG | CARDINALITY | Number of employees an organization has. |
| found_on | ORG | DATE, TIME | Founding date or time of an entity. |
| has_alternate_name | ORG, FAC | ORG, FAC | Alternative names or aliases for an entity. |
| has_area | GPE, LOC | QUANTITY | Area covered by a location or geopolitical entity. |
| official_language | GPE, LOC | LANGUAGE | Official language of a country or region. |
| has_currency | GPE, LOC | CURRENCY | Currency used by a geopolitical entity. |
| has_population | GPE | CARDINALITY | Population of a geopolitical entity. |
| capital_of | GPE | GPE | Links a capital city to its country. |

Table 10: List of Relation Types, each with Domain, Range, and Descriptions in *Wojood*$^{Relations}$

| Relation | NLI-RE(NLI-Based) | | | RIFRE (GNN-Based) | | | GPT-RE (LLM-Based) | | | Support |
|---|---|---|---|---|---|---|---|---|---|---|
| | P. | R. | F1 | P. | R. | F1 | P. | R. | F1 | |
| leader_of | 1.00 | 1.00 | **1.00** | 0.67 | 0.33 | **0.44** | 0.16 | 0.50 | **0.24** | 6 |
| manager_of | 0.86 | 0.86 | **0.83** | 0.38 | 0.79 | **0.51** | 0.38 | 0.38 | **0.38** | 24 |
| president_of | 0.84 | 0.84 | **0.82** | 0.59 | 0.70 | **0.64** | 0.63 | 0.43 | **0.51** | 46 |
| employee_of | 0.81 | 0.81 | **0.80** | 0.91 | 0.46 | **0.61** | 0.64 | 0.59 | **0.61** | 46 |
| member_of | 0.84 | 0.84 | **0.84** | 1.00 | 0.66 | **0.80** | 0.27 | 0.66 | **0.38** | 53 |
| owner_of | 0.82 | 0.75 | **0.71** | 0.00 | 0.00 | **0.00** | 0.67 | 0.40 | **0.50** | 5 |
| student_at | 1.00 | 1.00 | **1.00** | 0.00 | 0.00 | **0.00** | 0.50 | 1.00 | **0.67** | 1 |
| has_competitor | 0.64 | 0.67 | **0.65** | 0.00 | 0.00 | **0.00** | 0.50 | 0.33 | **0.40** | 3 |
| has_conflict_with | 0.70 | 0.71 | **0.67** | 0.90 | 0.69 | **0.78** | 0.97 | 0.77 | **0.86** | 39 |
| has_partner_with | 0.85 | 0.85 | **0.81** | 0.87 | 0.91 | **0.89** | 0.85 | 0.72 | **0.78** | 57 |
| has_parent | 0.83 | 0.67 | **0.67** | 0.20 | 1.00 | **0.33** | 1.00 | 0.50 | **0.67** | 2 |
| has_relative | 0.00 | 0.00 | **0.00** | 0.00 | 0.00 | **0.00** | 0.20 | 1.00 | **0.33** | 2 |
| has_sibling | 0.44 | 0.67 | **0.53** | 0.00 | 0.00 | **0.00** | 1.00 | 1.00 | **1.00** | 2 |
| has_spouse | 0.50 | 0.50 | **0.50** | 0.00 | 0.00 | **0.00** | 1.00 | 1.00 | **1.00** | 3 |
| capital_of | 0.88 | 0.83 | **0.83** | 0.25 | 0.17 | **0.20** | 0.50 | 0.17 | **0.25** | 6 |
| has_currency | 1.00 | 0.40 | **0.57** | 0.71 | 1.00 | **0.83** | 1.00 | 0.60 | **0.75** | 5 |
| has_population | 1.00 | 1.00 | **1.00** | 1.00 | 1.00 | **1.00** | 0.67 | 1.00 | **0.80** | 2 |
| official_language | 0.00 | 0.00 | **0.00** | 0.00 | 0.00 | **0.00** | 1.00 | 1.00 | **1.00** | 2 |
| has_border_with | 0.49 | 0.70 | **0.58** | 0.00 | 0.00 | **0.00** | 0.30 | 0.50 | **0.38** | 6 |
| headquartered_in | 0.90 | 0.88 | **0.87** | 0.69 | 0.82 | **0.75** | 0.39 | 0.82 | **0.53** | 11 |
| lives_in | 0.88 | 0.88 | **0.88** | 0.96 | 0.95 | **0.95** | 0.93 | 0.85 | **0.89** | 279 |
| located_in | 0.92 | 0.92 | **0.92** | 0.98 | 0.99 | **0.99** | 0.92 | 0.91 | **0.92** | 1306 |
| nearby | 0.85 | 0.84 | **0.81** | 0.86 | 0.63 | **0.73** | 0.90 | 0.68 | **0.78** | 38 |
| employs | 1.00 | 1.00 | **1.00** | 1.00 | 1.00 | **1.00** | 0.14 | 1.00 | **0.25** | 1 |
| found_on | 0.92 | 0.75 | **0.79** | 1.00 | 1.00 | **1.00** | 1.00 | 0.29 | **0.44** | 7 |
| has_alternate_name | 0.90 | 0.90 | **0.89** | 0.78 | 0.95 | **0.85** | 0.83 | 0.83 | **0.83** | 40 |
| has_property | 0.60 | 0.60 | **0.60** | 0.50 | 0.50 | **0.50** | 1.00 | 0.25 | **0.40** | 4 |
| geopolitical_division | 0.93 | 0.93 | **0.93** | 0.96 | 0.98 | **0.97** | 0.95 | 0.86 | **0.90** | 576 |
| subsidary | 0.85 | 0.84 | **0.84** | 0.82 | 0.88 | **0.85** | 0.94 | 0.77 | **0.85** | 64 |
| birth_date | 0.33 | 0.57 | **0.42** | 1.00 | 0.25 | **0.40** | 0.94 | 0.77 | **0.84** | 4 |
| birth_place | 1.00 | 0.97 | **0.99** | 0.75 | 0.92 | **0.82** | 1.00 | 0.50 | **0.67** | 36 |
| death_date | 1.00 | 1.00 | **1.00** | 0.83 | 1.00 | **0.91** | 0.77 | 0.94 | **0.85** | 5 |
| has_occupation | 0.85 | 0.84 | **0.83** | 0.95 | 1.00 | **0.97** | 0.94 | 0.70 | **0.81** | 203 |
| builder_of | 1.00 | 1.00 | **1.00** | 0.00 | 0.00 | **0.00** | 1.00 | 1.00 | **1.00** | 2 |
| founder_of | 0.00 | 0.00 | **0.00** | 0.00 | 0.00 | **0.00** | 1.00 | 1.00 | **1.00** | 2 |
| manufacturer_of | 0.25 | 0.50 | **0.33** | 0.00 | 0.00 | **0.00** | 0.17 | 0.50 | **0.25** | 2 |

Table 11: Performance breakdown of supervised relation extraction models across individual relation types