iMETRE: Incorporating Markers of Entity Types for Relation Extraction

N Harsha Vardhan
International Institute of Information Technology
Hyderabad, India
nemani.v@research.iiit.ac.in

Manav Chaudhary
International Institute of Information Technology
Hyderabad, India
manav.chaudhary@research.iiit.ac.in

ABSTRACT

Sentence-level relation extraction (RE) aims to identify the relationship between 2 entities given a contextual sentence. While there have been many attempts to solve this problem, the current solutions have a lot of room to improve. In this paper, we approach the task of relationship extraction in the financial dataset REFinD [2]. Our approach incorporates typed entity markers representations, and various models finetuned on the dataset, which has allowed us to achieve an F_1 score of 69.65% on the validation set. Through this paper, we discuss various approaches and possible limitations.

KEYWORDS

REFinD, Relation Extraction, Natural Language Processing, Finance, Information Retrieval

ACM Reference Format:
N Harsha Vardhan and Manav Chaudhary. 2023. iMETRE: Incorporating Markers of Entity Types for Relation Extraction. In *Proceedings of The 4th Workshop on Knowledge Discovery from Unstructured Data in Financial Services (KDF '23).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

To extract meaningful semantic relationships from textual data, in the field of natural language processing (NLP) and information retrieval, relation extraction (RE) plays a defining role. The task involves the categorization of the connections between two entities of specific types, such as persons, organizations, or locations, into various semantic categories like financial relationships, employment associations, or geographical affiliations, among others.

Although large-scale datasets produced from generic knowledge sources like Wikipedia, online texts, and news articles have significantly improved the performance of existing models for relation extraction, these resources frequently fall short in capturing domain-specific hurdles. Financial text documents (such as financial reports and Securities and Exchange Commission (SEC) filings) require complicated extraction methods, presenting a unique set of difficulties that necessitate domain-specific extraction methods. These documents have entities and relations that involve but are not limited to numbers, money, dates, legal information, and claims. Additionally, the sentences in such financial documents are lengthier, more complex, and cover a broader range of entity distances.

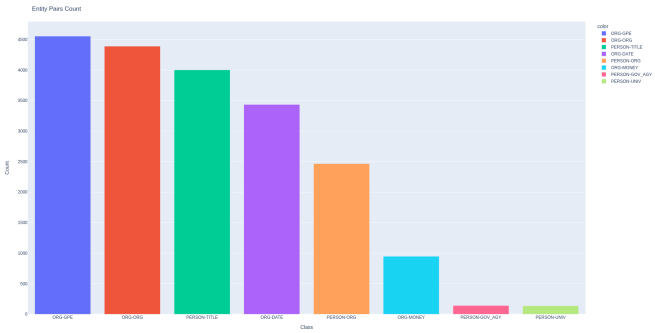

Figure 1: Count of Entity Pairs in the ReFind Dataset

The REFinD dataset[2] is the largest relation extraction dataset for financial documents to date and contains 29K instances and 22 relations among 8 types of entity pairs. REFinD is a domain-specific financial relation extraction dataset created using raw text from various 10-X reports (including 10-K, 10-Q, etc., broadly known as 10-X) of publicly traded companies obtained from US Securities and Exchange Commission (SEC) which is a rich and complex data source.

Compared to other well-known datasets for relation extraction such as TACRED (with 32 relations and a majority (79.5%) of which being NO_RELATION instances), REFinD is a much more balanced dataset with only 45.5% being NO_RELATION and has a higher number of instances for each relation it covers. Among financial datasets, both FinRED [6] and CorpusFR are smaller compared to REFinD and have fewer relation types. REFinD also contains longer sentences with an average sentence length of 53 words and average contextual complexity of 11 words between entity pairs.

This research endeavor aims to bridge the gap between general-purpose relation extraction datasets and the unique demands of the finance domain. By utilizing REFinD, researchers and practitioners in the field of financial NLP can access a rich and extensive resource that facilitates the extraction of financial entities and their intricate relationships from the text. This dataset holds great potential for advancing various tasks, including constructing knowledge graphs, question-answering systems, and semantic search engines tailored to the financial domain.

The main contributions of this paper include the following:

- **Markers for Entity Types:** This approach incorporates both entity types and spans into the context for classification,

KDF '23, July 27, 2023, Taipei, Taiwan

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

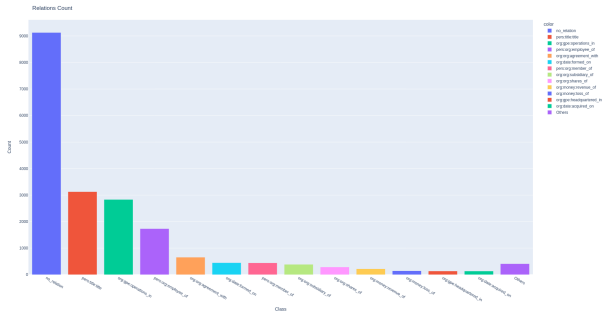

Figure 2: Count of Relations in the ReFind Dataset

leading to accurate relationship predictions. Typed Entity Markers are used in PLM-based RE models to mark entity spans and types using punctuation [10], enhancing contextual embeddings and improving the model's understanding of entity relationships.

- **Model Exploration:** We explore and compare the performance of various transformer-based models, including BERT-Base[1], RoBERTa-Base[3], DistilBERT[4], LUKE-Base[8], XLNet-Base[9], and FLANG-DistilBERT[5], in the context of sentence-level relation extraction and the ReFinD dataset, which provides valuable insights into the effectiveness of different models in capturing entity relationships in the financial domain.
- **Entity Type Pair Classification:** To address the weak influence of entity types and the ambiguity between semantically close relationships, we discuss a novel approach of dividing the classification task into eight tasks based on entity type pairs. This approach leads to improved classification accuracy and reduces error rates by considering the specific characteristics of each entity type pair.

2 METHODOLOGY

In this section, we first formally define the problem in the context of the ReFinD dataset while also presenting various model architectures and experiments performed along with the quantified metrics.

2.1 Problem Definition

In this task, we focus on sentence-level Relation Extraction. Specifically , given a sentence x and an entity pair (e_1, e_2), the objective is to predict the relationship r between e_1 and e_2 in the sentence x.

2.2 Model Architecture

The Relation Extraction(RE) classifier is based on the Permutated Language Modelling (PLM) objective-based RE models. Given the input sentence x, we mark the entity spans and entity types using a variation of the typed entity marker[7]. To this end, we mark an entity type and span using punctuation marks [10]. Specifically, the subject entity is enclosed with " @ " and the object entity with " # ". The entity types are also represented using the label text, which is prepended to the entity spans and is enclosed by " * " for subjects and " ∧ " for objects. For instance, given a text, the modified text would be ..."@ *subj-type* SUBJ @ . . . # ∧ obj-type ∧ OBJ # "....This allows us to include both; the type of entities, as well as the entity spans into context for classification. This is a variation of many such similar techniques, entity masks, and markers which are well-defined solutions in the task of RE.

We then feed the generated sentence to a model to get contextual embeddings and modify the model's last layer to a softmax classifier to obtain inferences. In inference, the class with the highest softmax probability is predicted as the relationship. We use a batch size of 8 and finetune the model for the training dataset for five epochs. Also, the reported metric on the test set is F_1, while on the validation set is a stricter F_1 metric. More specifically, while calculating F_1, we remove all instances where the model correctly predicts NO_RELATION and use the remaining instances to calculate F_1.

2.3 Experiments

In this section, we evaluate and report proposed techniques on the REFinD dataset. We primarily tried two approaches: one, we switch the models while keeping the configurations constant for better metrics, and two, we individually try to model pair-pair relation extraction accurately and use multiple models for inference which depend on the involved entity type pair.

2.3.1 Switching Architectures. Using the above data processing methods for capturing entity spans and types, we employ various models, namely BERT-Base, RoBERTa-Base, DistilBert, LUKE-Base, XLNET-Base, and FLANG-DistilBERT, for capturing entity spans and types, each of which is trained on different datasets and subsequently fine-tuned. The results obtained from these models have been compiled and are presented in Table 1.

- BERT-Base[1] : BERT is a transformer-based model which utilizes bidirectional context to generate word representations. BERT-Base is trained on general text like English Wikipedia and books.
- RoBERTa-Base[3] : Robustly Optimized BERT is an optimized variant of BERT that achieves improved performance by utilizing better training procedures, dynamic masking, and pretraining on a larger corpus.
- DistilBERT[4] : DistilBERT is a condensed version of BERT that retains most of its performance while reducing the model size and computational requirements using the concept of knowledge distillation, where a smaller model is trained to mimic the behavior of the larger BERT model, resulting in a compact yet efficient representation.
- LUKE-Base[8] : LUKE is a new pre-trained contextualized representation of words and entities based on transformer.LUKE-base utilizes a hybrid architecture combining transformer-based contextualized and knowledge-based embeddings, resulting in improved performance on entity-related tasks.
- XLNet-Base[9] : XLNET is a novel generalized permutation-based training objective that helps the model to consider all possible permutations of input during training, allowing the model to capture both bidirectional contexts which helps in differentiating between Directional Ambiguity between entities.

- FLANG-DistilBERT[5] : FLANG-DistilBERT is built by further training the DistilBERT language model in the finance domain with improved performance over previous models due to the use of domain knowledge and vocabulary.

Table 1: Scores across Models

Model	Test Set	
	Micro F_1 Score	Accuracy
BERT-Base	0.73	0.77
RoBERTa-Base	0.74	0.75
DistilBERT	0.73	0.78
XLNET-Base	0.75	0.79
FLANG-DistilBERT	0.75	0.79
LUKE-Base	0.72	0.76

Since each of the models has been trained on different data and was different in terms of architecture, there have been slight increments in metrics across models.

2.3.2 Classification based on Entity Type Pair. One of the shortcomings of treating the task as a 22-label classification problem is that the classification is weakly affected by the entity types. This is coupled with the fact that some of the relationships are semantically close and ambiguous. For example, $member\_of$, $employee\_of$, and $founder\_of$ for entity pairs belonging to the PER-ORG, PER-UNIV, and PER-GOV groups are often confused. Since the data distribution is also non-uniform across classes, this increases the error rate. Hence we divided the classification task into eight classification tasks, one for each entity type pair, keeping a constant model DistilBERT for fine-tuning. The results obtained have been compiled and presented in Table 2.

Table 2: Cumulative Accuracy, F_1 Score, and Additional Metric for each Entity Pair using DistilBERT

Entity Pair	Test Set		REFinD F_1 Baseline
	F_1 Score	Accuracy	
ORG-GPE	0.79	0.76	0.85
ORG-ORG	0.69	0.69	0.41
PERS-TITLE	0.88	0.82	0.90
ORG-DATE	0.91	0.91	0.81
PERS-ORG	0.70	0.71	0.67
ORG-MONEY	0.87	0.87	0.78
PERS-UNIV	0.65	0.62	0.61
PERS-GOV_AGY	0.72	0.66	0.22

Although we can see better classification metrics in some entity pair classes, the overall F_1 score is relatively low. This is primarily due to the unique operationalization of the F_1 used for grading on the validation set as well as the disparity along entity-pair classes. The performance also increases in entity pair classes since some relationships are semantically similar, and even the usage of entity markers fails to compensate adequately, while segregation of such classes shows classwise better results.

Apart from these, given the unique operationalization of the F_1 score used for grading in the shared task, approaches for an architecture that leverages the high % of *NO_RELATION* entries, improvements with larger models of the architectures above mentioned can be some potential ideas of interest.

3 CONCLUSION

We see that using a modification of the classical entity mask and marker approaches which are slightly flawed due to a lack of understanding of numerical inferences (using entity markers (quote MTB)), while the proposed punctuated entity markers are better at capturing context within relations of entities as well as detecting the entity spans.

We also observe that although the results seem consistent and equitable across models, XLNET-base outperforms the others. This can be mainly attributed to the bidirectional nature of training data which, in essence, captures the importance of sequencing between entities for relation prediction, and due to structured textual forms in REFinD dataset, permutation-based attention masking might have also played a crucial role. This also achieves an F_1 score of 69.65% on the validation dataset.

Also, from the eight-class approach, we can see an individual increase in some classes to a high degree. This could be attributed to better separation between semantically similar results. Although this approach still fails to outperform overall F_1, across models, mainly due to disparity in support of the classes, it still might be better for pair-specific classifications.

4 FUTURE WORK

Theoretically, using Larger Models instead of Distilled and Base Models should improve the results. Also, techniques like data augmentation to inflate classes with low support might also increase the overall efficiency and viability of the models and dataset. Some other approaches of differentiating based on entity type pairs and integrating into the final layer might also yield some interesting results.

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[2] Simerjot Kaur, Charese Smiley, Akshat Gupta, Joy Sain, Dongsheng Wang, Suchetha Siddagangappa, Toyin Aguda, and Sameena Shah. 2023. REFinD: Relation Extraction Financial Dataset. *arXiv preprint arXiv:2305.18322* (2023).

[3] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[4] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).

[5] Raj Sanjay Shah, Kunal Chawla, Dheeraj Eidnani, Agam Shah, Wendi Du, Sudheer Chava, Natraj Raman, Charese Smiley, Jiaao Chen, and Diyi Yang. 2022. WHEN FLUE MEETS FLANG: Benchmarks and Large Pre-trained Language Model for Financial Domain. *arXiv preprint arXiv:2211.00083* (2022).

[6] Soumya Sharma, Tapas Nayak, Arusarka Bose, Ajay Kumar Meena, Koustuv Dasgupta, Niloy Ganguly, and Pawan Goyal. 2022. FinRED: A Dataset for Relation Extraction in Financial Domain. In *Companion Proceedings of the Web Conference 2022*. 595–597.

[7] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. *arXiv preprint arXiv:1906.03158* (2019).

[8] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. *arXiv preprint arXiv:2010.01057* (2020).
[9] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* 32 (2019).
[10] Wenxuan Zhou and Muhao Chen. 2021. An improved baseline for sentence-level relation extraction. *arXiv preprint arXiv:2102.01373* (2021).

A PERFORMANCE OF MODELS

Table 3: F_1 Scores across Models and Relations: Part 1

Class	Performance on the Test Set		
	BERT-Base	RoBERTa-Base	DistilBERT
Class 0	0.80	0.78	0.81
Class 1	0.86	0.83	0.87
Class 2	0.85	0.85	0.85
Class 3	0.15	0.14	0.13
Class 4	0.78	0.74	0.79
Class 5	0.93	0.67	0.77
Class 6	0.00	0.00	0.11
Class 7	0.84	0.00	0.83
Class 8	0.71	0.55	0.74
Class 9	0.56	0.63	0.73
Class 10	0.46	0.44	0.46
Class 11	0.92	0.00	0.93
Class 12	0.46	0.00	0.49
Class 13	0.67	0.00	0.73
Class 14	0.57	0.00	0.00
Class 15	0.49	0.59	0.49
Class 16	0.25	0.21	0.25
Class 17	0.45	0.15	0.50
Class 18	0.90	0.91	0.90
Class 19	0.71	0.52	0.65
Class 20	0.39	0.38	0.36
Class 21	0.62	0.00	0.55

Table 4: F_1 Scores across Models and Relations: Part 2

Class	Performance on the Test Set		
	XLNet-Base	FLANG-DistilBERT	LUKE-Base
Class 0	0.81	0.82	0.79
Class 1	0.90	0.86	0.79
Class 2	0.86	0.86	0.86
Class 3	0.20	0.20	0.17
Class 4	0.80	0.79	0.76
Class 5	0.88	0.86	0.86
Class 6	0.18	0.10	0.00
Class 7	0.87	0.87	0.72
Class 8	0.81	0.74	0.69
Class 9	0.76	0.76	0.33
Class 10	0.43	0.54	0.50
Class 11	1.00	1.00	0.67
Class 12	0.46	0.45	0.30
Class 13	0.62	0.73	0.00
Class 14	0.00	0.00	0.00
Class 15	0.53	0.52	0.57
Class 16	0.22	0.21	0.17
Class 17	0.52	0.49	0.26
Class 18	0.91	0.91	0.91
Class 19	0.73	0.73	0.69
Class 20	0.34	0.38	0.42
Class 21	0.67	0.62	0.50