IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 22, 2021, accepted March 21, 2021, date of publication April 15, 2021, date of current version April 30, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3073428

Improving Distantly-Supervised Relation Extraction Through BERT-Based Label and Instance Embeddings

DESPINA CHRISTOU AND GRIGORIOS TSOUMAKAS
School of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

Corresponding author: Despina Christou (christoud@csd.auth.gr)

This work was supported by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, through the call RESEARCH–CREATE-INNOVATE, in the context of the Exploitation of Cultural Assets with computer-assisted Recognition, Labeling and meta-data Enrichment (ECARLE) Project under Project T1EDK-05580.

ABSTRACT Distantly-supervised relation extraction (RE) is an effective method to scale RE to large corpora but suffers from noisy labels. Existing approaches try to alleviate noise through multi-instance learning and by providing additional information but manage to recognize mainly the top frequent relations, neglecting those in the long-tail. We propose REDSandT (Relation Extraction with Distant Supervision and Transformers), a novel distantly-supervised transformer-based RE method that manages to capture a wider set of relations through highly informative instance and label embeddings for RE by exploiting BERT's pre-trained model, and the relationship between labels and entities, respectively. We guide REDSandT to focus solely on relational tokens by fine-tuning BERT on a structured input, including the sub-tree connecting an entity pair and the entities' types. Using the extracted informative vectors, we shape label embeddings, which we also use as an attention mechanism over instances to further reduce noise. Finally, we represent sentences by concatenating relation and instance embeddings. Experiments in the two benchmark datasets for distantly-supervised RE, NYT-10 and GDS, show that REDSandT captures a broader set of relations with higher confidence, achieving a state-of-the-art AUC (0.424) in NYT-10 and an excellent AUC (0.862) in GDS.

INDEX TERMS Relation extraction, distant supervision, BERT, label embeddings, relation attention, entity information.

I. INTRODUCTION

Relation Extraction (RE) aims to detect semantic relationships between entity pairs in natural texts and has proven to be crucial in various natural language processing (NLP) applications, including question answering, and knowledge-base (KB) population.

Most RE methods follow a supervised approach, with the required number of labeled training data rendering the whole process time and labor-intensive. To automatically construct datasets for RE, [1] proposed to use distant supervision (DS) from a KB, assuming that if two entities exhibit a relationship in a KB, then all sentences mentioning these entities express this relation. Inevitably, this assumption generates false-positives and leads distantly-created datasets to contain erroneous labels. To alleviate the *wrong labeling problem*, [2] relaxed this assumption so that it does not hold for all instances and along with [3], [4] proposed multi-instance based learning. Under this setting, classification shifts from instance-level to bag-level, with a bag consisting of all instances that contain a specific entity pair.

Current state-of-the-art RE methods try to reduce the effect of noisy instances by: i) identifying valid instances through multi-instance learning and selective attention [5], ii) reducing inner-sentence noise by capturing long-range dependencies using syntactic information from dependency parses [1], [6], [7], specialized models like piecewise CNN (PCNN) and graph CNN (GCNN), or word-level attention [6], and iii) enhancing model effectiveness using external knowledge (i.e. KB entity types [8], entity descriptions [9], [10], relation phrases [8]) or transfer knowledge from pre-trained models [11].

The study of the above approaches led us to the following core observations. First, among all models used in the

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Wei Tsai.

literature, the use of a pretrained transformer-based language model (LM) can help in recognizing a broader set of relations, even though at the expense of time and computational resources, and second, the relationship between label and entities can entail valuable information but rarely used over external knowledge. Driven by these observations we inspired to develop a novel transformer-based model that can efficiently capture instance and label embeddings in less complexity so as to drive RE in recognizing a broader set of relations.

We propose REDSandT (Relation Extraction with Distant Supervision and Transformers), a novel transformer-based RE model for distant supervision. To handle the problem of noisy instances, we guide REDSandT to focus solely on relational tokens by fine-tuning BERT on a structured input, including the sub-tree connecting an entity pair (STP) and the entities' types. The input's RE-specific formation, along with BERT's knowledge from unsupervised pre-training, results in REDSandT generating informative vectors. Using these vectors, we shape relation embeddings representing the entities' distance in vector space. Relation embeddings are then used as relation-wise attention over instance representation to reduce the effect of less-informative tokens. Finally, REDSandT encodes sentences by concatenating relation and weighted-instance embeddings, with relation classification to occur at bag-level as a weighted sum over its sentences' predictions.

We chose BERT over other transformer-based models because it considers bidirectionality while training. We assume that this characteristic is important to efficiently capture entities' interactions without requiring an additional task that importantly increases complexity (i.e. fine-tuning an auxiliary objective in GPT [11]).

The main contributions of this paper can be summarized as follows:

- We extend BERT to handle multi-instance learning to directly fine-tune the model in a DS setting and reduce error accumulation.
- Relation embeddings captured through BERT fine-tuned on our RE-specific input help to recognize a wider set of relations, including relations in the long-tail.
- Suppressing the input sentence to its relational tokens through STP encoding allowed us to capture informative instance embeddings while preserving low complexity to train our model on modest hardware.
- Experiments on the NYT-10 dataset show REDSandT to surpass state-of-the-art models [8], [11], [12] in AUC (1.0 & 0.2 & 39 units respectively) and performance at higher recall values, while achieving a 7-10% improvement in P@{100, 200, 300} over [11].
- Experiments on the GDS dataset show REDSandT to surpass the state-of-the-art RESIDE [8] in P@{100, 200, 300} values.
- We make our code publicly available at https://github.com/DespinaChristou/REDSandT.

II. REDSANDT

Given a bag of sentences $\{s_1, s_2, \ldots, s_n\}$ that concern a specific entity pair, REDSandT generates a probability distribution on the set of possible relations. REDSandT utilizes BERT pre-trained LM to capture the semantic and syntactic features of sentences by transferring pre-trained commonsense knowledge. We extend BERT to handle multi-instance learning, and we fine-tune the model to classify the relation linking the entity pair given the associated sentences.

During fine-tuning, we employ a structured, RE-specific input to minimize architectural changes to the model [13]. Each sentence is adapted to a structured text, including the sentences' tokens connecting the entity pair (STP) along with the entities types. We transform the input into a (sub-)word-level distributed representation using BPE and positional embeddings from BERT fine-tuned on our corpora. Then, we form final sentence representation by concatenating relation embedding and sentence representation weighted with the relation embedding. Lastly, we use attention over the bag's sentences to shape bag representation, which is then fed to a softmax layer to get the bag 's relation distribution.

REDSandT can be summarized in three components, namely sentence encoder, bag encoder, and model training. Each component is described in detail in the following sections with the overall architecture shown in Fig. 1 and 3.

A. SENTENCE ENCODER

Given a sentence x and an entity pair $\langle h, t \rangle$, REDSandT constructs a distributed representation of the sentence by concatenating relation and instance embeddings. Overall sentence encoding is represented in Fig. 1, with following sections to examine the sentence encoder parts in a bottom-up way.

1) INPUT REPRESENTATION

Relation extraction requires a structured input that can sufficiently capture the latent relation between an entity pair and its surrounding text. Our input representation encodes each sentence as a sequence of tokens, depicted in the very bottom of Fig. 1.

It starts with the head entity type and token(s) followed by delimiter [H-SEP], continues with the tail entity type, and token(s) followed by delimiter [T-SEP] and ends with the token sequence of the sentence's STP path. The whole input starts and ends with special delimiters [CLS] and [SEP], respectively. In BERT, [CLS] typically acts as a pooling token representing the whole sequence for downstream tasks, such as RE.

Several other sentence encodings were attempted[1] with the presented one to perform the best. Moreover, the ablation studies in section IV-B, reveal the importance of encoding entities' types and compressing the original sentence to the below-presented STP path. Below, we present in brief how we form the sub-tree parse of the input and the entity types.

[1]Trials included encoding overall sentence tokens, STP tokens only, SDP ([14]) tokens only, using common $\langle h, t \rangle$ delimiter, using single delimiter between entities and STP, removing entity type information.

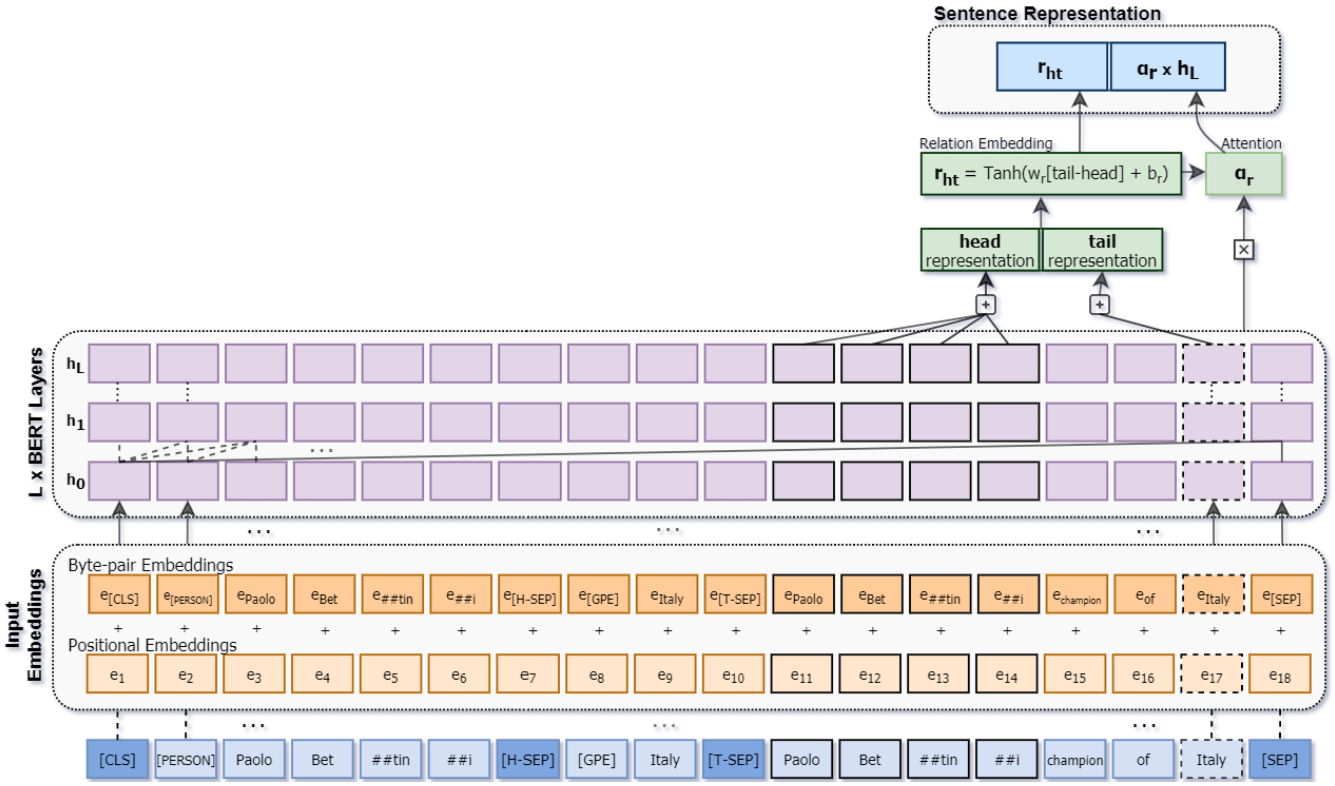

FIGURE 1. Sentence Representation in REDSandT. The input embedding h_0 to BERT is created by summing over the positional and byte pair embeddings for each token in the structured input. States h_t are obtained by self-attending over the states of the previous layer h_{t-1}. Final sentence representation is obtained by concatenating the relation embedding r_{ht}, and the final fine-tuned BERT layer h_L weighted with relation attention α_r. Head and tail tokens participating in the relation embedding formation are marked with bold and dashed lines respectively.

Text The defending champion, Paolo Bettini of Italy, won the Tour of Lombardy yesterday for his 11th Pro Tour title.
STP Paolo Bettini champion of Italy
SDP Paolo Bettini of Italy

FIGURE 2. Example of STP and SDP sentence encoding.

a: SUB-TREE PARSE OF INPUT SENTENCE

We utilize the sub-tree parse (STP) of the input sentence in order to reduce the noisy words within sentence and focus on the relational tokens. As seen in Fig. 2, STP preserves the path of the sentence that connects the two entities with their least common ancestor (LCA)'s parent. Compared to other implementations [7], who shape the final STP sequence by re-assigning the participating tokens into their original sequence order, we preserve the tokens' order within STP achieving a grammatical normalization of the original sentence.

b: ENTITY TYPE SPECIAL TOKENS

In the extent that every relation puts some constraint on the type of participating entities [8], [15], we incorporate the entity type in the model's structured input (see bottom of Fig. 1).

Precisely, we incorporate 18 generic entity types, captured from recognizing NYT-10 sentence's entities with the spaCy model.[2] We assume these types KB-independent and easily accessible with our experiments in section IV-B indicating their inclusion to improve performance.

[2]https://spacy.io/models/en

2) INPUT EMBEDDINGS

The input embedding h_0 to BERT is created by summing over the positional and byte pair embeddings for each token in the structured input.

a: BYTE-PAIR TOKENS ENCODING

To make use of sub-word information, we tokenize input using byte-pair encoding (BPE) [16]. We particularly use the tokenizer from the pre-trained model (30,000 tokens), which we extend with 20 task-specific tokens (e.g., [H-SEP], [T-SEP], and the 18 entity type tokens). Added tokens serve a special meaning in the input representation, thus are not split into sub-words by the tokenizer.

b: POSITIONAL ENCODING

Positional encoding is an essential part of BERT's attention mechanism. Precisely, BERT learns a unique position embedding to represent each of the input (sub-word) token positions within the sequence.

3) SENTENCE REPRESENTATION

Input sequence is transformed into feature vectors (h_L) using BERT's pre-trained language model, fine-tuned in our task. In spite of common practice to represent the sentence by the [CLS] vector in h_L [11], we argue that not all words contribute equally to sentence representation.

By encoding the underlying relation as a function of the examining entities and by giving attention to vectors related to this underlying relation, we can further reduce sentence noise and improve precision. Core modules constitute the relation embedding, entities-wise attention, and relation attention. We examine them below.

a: RELATION EMBEDDING

: We formulate relation embeddings using the TransE model [17]. TransE model regards the embedding of the underlying relation l as the distance (difference) between h and t embeddings ($l_i = t_i - h_i$), assuming that a relation r holds between an entity pair (h, t). Then, we shape relation embedding for each sentence i by applying a linear transformation on the head and tail entities vectors, activated through a Tanh layer to capture possible nonlinearities:

$$l_i = Tanh(w_l(t_i - h_i) + b_l), \tag{1}$$

where w_l is the underlying relation weight matrix and $b_l \in \Re^{d_t}$ is the bias vector. We mark relation embedding as l because it represents the possible underlying relation between the two entities and not the actual relationship r. Head h_i and tail t_i embeddings reflect only the entities' related tokens, which we capture by selecting solely the head and tail token embeddings from h_L as shown in Fig. 1.

b: RELATION ATTENTION

Even though REDSandT is trained on STP that naturally preserves only relational tokens, we wanted to further reduce possible left noise on sentence-level. For this reason, we use a relation attention to emphasize on sentence tokens that are mostly related to the underlying relation l_i. We calculate relation attention α_r by comparing each sentence representation from the last layer L against the learned representation l_i for each sentence i:

$$\alpha_r = \frac{exp(h_{iL} l_i)}{\sum_{j=1}^{n} exp(h_{jL} l_i)} \tag{2}$$

Then, we weight BERT' s last hidden layer $h_L \in \Re^{d_h}$ with relation embedding:

$$h'_L = \sum_{t=1}^{T} \alpha_r \cdot h_{it} \tag{3}$$

Finally, **sentence representation** $s_i \in \Re^{d_h * 2}$ is computed as the concatenation of the relation embedding l_i and the sentence's weighted hidden representation h'_L:

$$s_i = \left[l_i \; ; \; h'_L\right] \tag{4}$$

Several other representation techniques were tested, with the presented method to outperform.

B. BAG ENCODER

Bag encoding, i.e., aggregation of sentence representations in a bag, comes to reduce noise generated by the erroneously annotated relations accompanying DS. Assuming that not all sentences contribute equally to the bag representation, we use selective attention [5] to emphasize on sentences that better express the underlying relation.

$$B = \sum_i \alpha_i s_i, \tag{5}$$

As seen, selective attention represents bag as a weighted sum of the individual sentences. Attention α_i is calculated by comparing each sentence representation against a learned representation r:

$$\alpha_i = \frac{exp(s_i r)}{\sum_{j=1}^{n} exp(s_j r)} \tag{6}$$

Finally, bag representation B is fed to a softmax classifier to obtain the probability distribution over the relations.

$$p(r) = Softmax(W_r \cdot B + b_r), \tag{7}$$

where W_r is the relation weight matrix and $b_r \in \Re^{d_r}$ is the bias vector.

C. TRAINING

REDSandT utilizes a transformer model, precisely BERT, which fine-tunes on our specific setup to capture the semantic features of relational sentences. Below, we present the overall process.

1) MODEL PRE-TRAINING

For our experiments, we use the pre-trained *bert-base-cased* language model [18], which consists of 12 layers, 12 attention heads, and 110M parameters, with each layer being a bidirectional Transformer encoder [19]. The model is trained on cased English text of BooksCorpus and Wikipedia with a total of 800M and 2.5K words respectively. BERT is pre-trained using two unsupervised tasks: masked LM and next sentence prediction, with masked LM being its core novelty as it allows the previously impossible bidirectional training.

2) MODEL FINE-TUNING

We initialize REDSandT model' s weights with the pre-trained BERT model, and we fine-tune its last four layers under the multi-instance learning setting presented in Fig. 3, given the specific input shown in Fig. 1. We end up fine-tuning only the last four layers after experimentation.

During fine-tuning, we optimize the following objective:

$$L(D) = \sum_{i=1}^{|B|} logP(l_i|B_i; \theta), \tag{8}$$

where for all entity pair bags $|B|$ in the dataset, we want to maximize the probability of correctly predicting the bag's relation given its sentences' representation and parameters.

III. EXPERIMENTAL SETUP

A. DATASETS

We conduct experiments on the two widely-used benchmark datasets for distantly-supervised RE, namely NYT-10 [2] and GDS [20]. NYT-10 dataset was built by aligning triples in

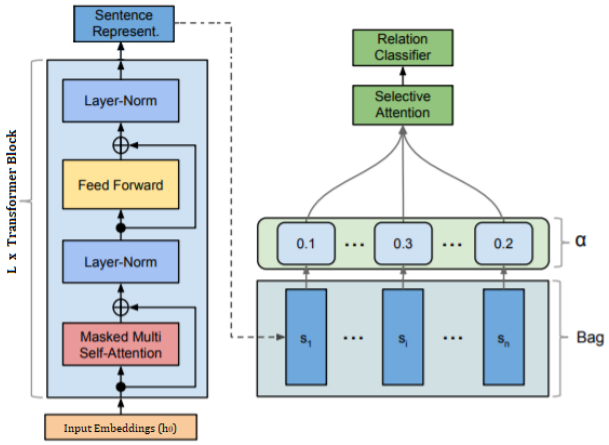

FIGURE 3. Transformer architecture (left) and training framework (right). Sentence representation s_i is formed as shown in Fig. 1.

TABLE 1. Datasets' Statistics.

Dataset	Split	# Sentences	# Entity Pairs
NYT-10	train	522,611	281,270
(53 relations)	test	172,448	96,678
GDS	train	11,297	6,498
(5 relations)	val	1,864	1,082
	test	5,663	3,247

Freebase to the NYT corpus, consists of many instances and relations, but contains much noise coming both from incorrect labels and unlabelled instances (73.8% of train instances and 96.3% of test instances exhibit no relation ("NA")), whereas GDS was built to precisely overcome DS noise (only 25% of the {train, val, test} instances exhibit no relation ("NA")), but consists of few instances and relations. You can view datasets' statistics in Fig. 1.

Moreover, we provide an enhanced version of both datasets, including STP encoding of the input sentences as well as the head and tail entity types to facilitate future implementations.

B. HYPER-PARAMETER SETTINGS

In our experiments we utilize *bert-base-cased* model with hidden layer dimension $D_h = 768$, while we fine-tune the model with *max_seq_length* $D_t = 64$. We use the Adam optimization scheme [21] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a cosine learning rate decay schedule with warm-up over 0.1% of training updates. Also, we minimize loss using cross entropy criterion weighted on dataset's classes to handle the unbalanced training set.

Regarding dataset-specific model's hyper-parameters, we manually tune them on the training (NYT-10) and validation (GDS) set based on AUC score. Fig. 2 presents the applied searching space and selected values for the dataset-specific hyper-parameters.

Experiments conducted on a PC with 32.00 GB RAM, Intel i7-7800X CPU@ 3.5GHz and NVIDIA's GeForce GTX 1080 with 8GB. Training time takes about 100min/epoch and 3min/epoch for the NYT-10 and GDS datasets, respectively. Our code will be publicly available upon acceptance.

TABLE 2. Dataset-specific model hyper-parameters.

Hyper-parameter	Search Space	NYT-10	GDS
Batch Size	[8, 16, 32]	32	32
Epochs	[3, 4]	3	3
Dropout	[0.2, 0.4, 0.5, 0.6]	0.4	0.4
Learning Rate	$[2e^{-4}, 2e^{-5}, 5e^{-5}, 5.5e^{-5}]$	$2e^{-5}$	$5.5e^{-5}$
Weight Decay	[0.01, 0.001]	0.001	0.001
Fine-tuned layers	[last(2, 4, 8), all]	last 4	last 4

C. BASELINE MODELS

To show the proposed method's effectiveness, we compare against five strong baselines in the two benchmark datasets for distantly-supervised RE. Precisely, we compare REDSandT to [1], [5], [8], [11], [12] in the NYT-10 dataset and to [8], [12] in the GDS dataset. The above choice was taken considering the publicly available source codes and available precision-recall results. Our reported results are those presented in the original papers or were captured using the open-source, released codes.

1) FEATURE-BASED METHODS

- **Mintz** [1]: A multi-class logistic regression model under distant supervision setting.
- **PCNN+ATT** [5]: A CNN model with instance-level attention

2) NEURAL NETWORK METHODS

- **RESIDE** [8]: A NN model that uses several side information (entity types,[3] relational phrases) and employs Graph-CNN to capture syntactic information of instances.
- **BERT-SIDE** [12]: A related approach to RESIDE. Simplifies its complex sentence encoding with BERT embeddings. No fine-tuning is applied.
- **DISTRE** [11]: A transformer model, GPT fine-tuned for RE with an auxiliary objective under the distant supervision setting.

D. EVALUATION CRITERIA

We report top-N precision ($P@N$, with $N \in \{100, 200, 300, 500\}$), precision-recall curves, and relation distribution on top-300 predicted relations. Moreover, we present several ablation studies and we show the relation-wise attention's effect on dataset's instances.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. COMPARISON WITH BASELINES

For evaluating the effectiveness of our model, we present for both datasets the precision-recall (P-R) curves (Fig. 4, 5), AUC and precision at various points in the P-R curve (Fig. 3, 4). Also, we report for the NYT-10 dataset the distribution over relation types for the top-300 predictions (Fig. 5) to review our model's efficiency in extracting a wider set of relations in top-N predictions.

[3]Compared to our 18 KB-independent entity types, authors use 38 Freebase-specific entity types.

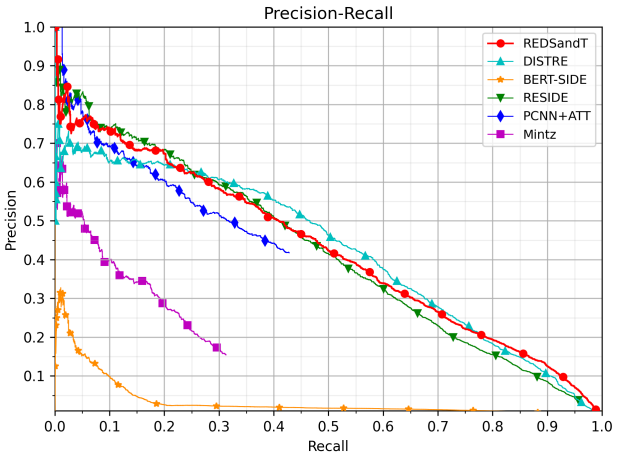

FIGURE 4. Precision-Recall curves in NYT-10 dataset.

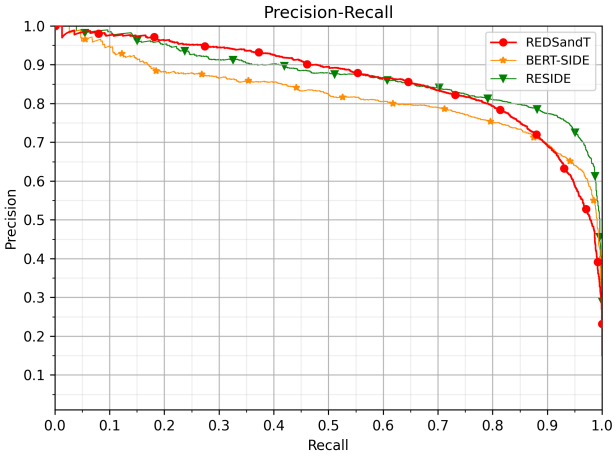

FIGURE 5. Precision-Recall curves in GDS dataset.

Fig. 4 compares the precision-recall curves of REDSandT in NYT-10 dataset against both feature-based and neural-based baseline models. We observe that: (1) The fine-tuned on distantly-supervised RE task NN-based approaches outperform the feature-based approaches (Mintz, PCNN+ATT), showing human-designed features limitation against neural networks' automatically extracted features. (2) BERT-SIDE, which is not fine-tuned on the task, presents a sudden decline in precision quickly enough, which its authors attribute to the high percentage of negatives (96.3% of test instances) that the model cannot efficiently handle. (3) RESIDE, DISTRE, and REDSandT achieve better performance than PCNN+ATT, which even exhibiting the highest precision in the beginning soon follows an abrupt decline. This reveals the importance of both side-information (i.e., entity types and relation alias), and transfer knowledge. (4) RESIDE performs the best in low recalls and generally performs well, which we attribute to the multitude of side-information given. (5) Although DISTRE exhibits 3.5% greater precision in medium-level recalls, it presents 2-12% lower precision in recall values <0.25 compared to RESIDE, and REDSandT. (6) Our model shows a stable behavior, with a steady, downward trend, acting similar to RESIDE at the low and medium recalls and surpassing all baselines in the very high recall values. We believe the reason is that we use potential label information as an additional feature and as attention over the instance tokens. The learned label embeddings are of high quality since they carry common-knowledge from the pre-trained model fine-tuned on the specific dataset and task. Moreover, the chosen pre-trained model, BERT, considers bidirectionality while training, being thus able to efficiently capture head and tail interaction. Respectively, Fig. 5 compares the P-R curves of REDSandT in the GDS dataset against the RESIDE-based approaches (RESIDE, BERT-SIDE). We observe that: (1) REDSandT and RESIDE perform almost equally, with REDSandT outperforming in the up-to-0.8 level recalls, while RESIDE surpasses all models in the highest-level recalls. (2) BERT-SIDE shows the most inferior performance among the three models, but it seems

TABLE 3. AUC and P@N evaluation results at NYT-10 dataset.

Methods	AUC	P@100	P@300	P@500
Mintz	0.107	52.3	45.0	39.7
PCNN+ATT	0.341	73.0	67.3	63.6
RESIDE	0.415	0.83	0.71	69.7
BERT-SIDE	0.034	1.00	1.00	1.00
DISTRE	0.422	68.0	65.3	65.0
REDSandT	**0.424**	78.0	73.0	67.6

to better handle the negatives (only 25% of test instances) in this dataset compared to its performance on NYT-10. (3) The extensive side-information of the RESIDE-based approaches adds value only in the very-high (>0.8) recall values.

Fig. 3, and 4, which present AUC and precision at various points in the P-R curve for both datasets, reveal that our model preserves the state-of-the-art AUC in the NYT-10 dataset and only 1.1 units less than RESIDE in the GDS dataset. Precisely, in the NYT-10 dataset, our model's precision is between that of RESIDE and DISTRE. REDSandT's s precision does not exceed RESIDE', even though it is close enough, suggesting that additional side-information or freebase-specific entities would improve performance. Additionally, REDSandT surpasses DISTRE's precision, which we attribute to our selected pre-trained model that efficiently captures label embeddings. Also, even though BERT-SIDE exhibits the greatest (100%) precision in top-{100-500} predictions, it presents far opposite results in AUC, showing that it is efficient in much fewer top-N predictions to the rest models. Meanwhile, in the GDS dataset, our model surpasses RESIDE in top precision values but exhibits 1.1 units less in AUC, following the P-R curve analysis. Moreover, similarly to NYT-10, BERT-SIDE achieves the highest precision in top-{100-500} predictions but shows the least AUC. Consequently, our model is more consistent across the various points of the P-R curve regardless of the dataset.

Fig. 5 shows the distribution over relation types for the top 300 predictions of REDSandT and baseline models in the NYT-10. REDSandT encompasses 10 distinct relation types, two of which (*/place_founded*, */geographic_distribution*) are

TABLE 4. AUC and P@N evaluation results at GDS dataset.

Methods	AUC	P@100	P@200	P@300
BERT-SIDE	0.819	1.00	0.99	0.99
RESIDE	0.872	0.95	0.93	0.92
REDSandT	0.862	0.99	0.98	0.98

TABLE 5. Relation distribution over the top 300 predictions for REDSandT and comparison models in NYT-10 dataset.

Relation	redsandt	distre	reside	pcnn+att
/location/contains	176	168	182	214
/person/company	38	31	26	19
/person/nationality	26	32	65	59
/admin_div/country	25	13	12	6
/neighborhood_of	22	10	3	2
/person/children	5	-	6	-
/team/location	4	2	-	-
/founders	2	2	6	-
/place_founded	1	-	-	-
/geo_distribution	1	-	-	-
/country/capital	-	17	-	-
/person/place_lived	-	22	-	-

not recognized by none of rest models. PCNN+ATT predictions are highly biased towards a set of only four relation types, while RESIDE captures three additional types. DISTRE and REDSandT manage to recognize more types than all models, emphasizing the contribution of transfer knowledge. Moreover, REDSandT correctly not recognizes */location/country/capital* relation that DISTRE does, as their authors found most errors to arise from the specific predicted relation in manual evaluation. Meanwhile, we highlight REDSandT' s effectiveness in recognizing relations in the long-tail. Particularly, our model captures, */founders* (1.47%), */neighborhood_of* (1.06%), */person/children* (0.47%), and */sports_team/location* (0.16%) relations. Relations are listed in descending order regarding population in test set with respective percentage referenced in parentheses.

We also manually inspect the top-300 predictions of REDSandT in the GDS dataset. All four relations including those in the long-tail were captured, while none "NA" relation was predicted.

B. ABLATION STUDIES

To assess the effectiveness of the different modules of REDSandT, we create four ablation models:

- **REDSandT w/o ET**: Removes entity types from input sentence representation.
- **REDSandT w/o** r_{ht}: Removes relation embedding and relation attention. We represent sentence using the [CLS] token of BERT's last hidden layer h_L.
- **REDSandT w/o** a_r: Removes relation attention on instance tokens.
- **REDSandT w. SDP**: Replaces STP with SDP [14] in sentence encoding.

As shown in Fig. 6, 7 all modules contribute to final model' s effectiveness, in both datasets. Starting to the NYT-10, the greatest impact comes from relation embeddings with their removal resulting in the highest AUC (2 units) and P@300 (5.3%) drop. Meanwhile, P@100 goes up to 80%

TABLE 6. AUC and P@N of variant models on NYT-10 dataset.

Metrics	AUC	P@100	P@200	P@300
REDSandT w/o r_{ht}	0.404	80.0	72.0	67.7
REDSandT w/o ET	0.415	78.0	74.0	71.3
REDSandT w. SDP	0.418	75.0	71.0	69.7
REDSandT w/o a_r	0.422	75.0	76.0	71.0
REDSandT	0.424	78.0	75.0	73.0

TABLE 7. AUC and P@N of variant models on GDS dataset.

Metrics	AUC	P@100	P@200	P@300
REDSandT w/o a_r	0.79	0.88	0.87	0.84
REDSandT w/o r_{ht}	0.83	0.95	0.92	0.90
REDSandT w/o ET	0.84	0.96	0.97	0.96
REDSandT	0.86	0.99	0.98	0.98

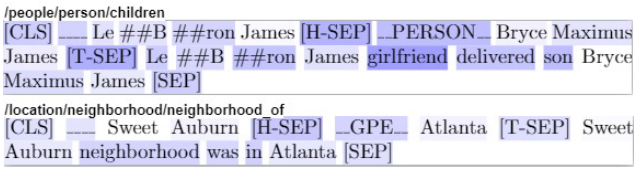

FIGURE 6. Relation attention weights in challenging long-tail relations. In the top example we see an instance of a *"/people/person/children"* relation, whereas in the bottom we view an instance of a *"/location/neighborhood/neighborhood_of"* relation.

with inspection of top-300 predictions revealing a focus on 5 relation types only, with */location/contains* to make up the 79% of these. Simple integration of entity types in input representation is the next most important feature that boosts our model. Next, "REDSandT w. SDP", shows STP's superiority, while a manual inspection in the model's top 300 predictions prove SDP's weakness to recognize relations in the long tail, with focus given on */person/nationality* relation. Finally, removing the relation attention over instance tokens exhibits the least effect in AUC (0.002) and precision (∼2%). Meanwhile, we notice that model focuses solely on 8 relation types in the top 300 predictions.

Respectively, the greatest impact in the GDS dataset comes from the relation attention mechanism with its removal to induce a 7-unit drop in AUC and a 11-14% drop in P@{100-300}. Manual inspection in top-300 predictions revealed "NA" relation to be among the top predicted. Next significant decline comes by removing relation embedding, which from manual inspection leads in recognizing only the most-frequent relation (*/people/person/place_of_birth*) in top-300 predictions ignoring all rest relations. At last, subtracting entity type information shows the least drop in performance, suggesting that our model's robustness regardless side-information.

C. CASE STUDY: EFFECT OF RELATION ATTENTION

Fig. 6 shows a visualization[4] of the relation attention weights, highlighting the different parts of the sentence that drive relation extraction, for two long-tail relations in the NYT-10 dataset. In both cases, we see that the special tokens preserve important information, while also the entity type is

[4]Obtained with https://github.com/jiesutd/Text-Attention-Heatmap-Visualization

given more weight than the entity itself. Moreover, we see which tokens affect more the relation. Tokens ''girlfriend'', ''son'', and the repetition of name ''James'' are predictive of the ''*children*'' relation, while tokens ''neighborhood'', ''was'', ''in'', along with a GPE entity type show a probable ''*neighborhood_of*'' relation.

V. RELATED WORK

Our work is related to distant supervision, neural relation extraction (mainly pre-trained LMs), sub-tree parse of input, label embedding, and entity type side information.

A. DISTANT SUPERVISION

DS plays a key role in RE, as it satisfies its need for extensive training data, easily and inexpensively. The use of DS [22], [23] to generate large training data for RE was proposed by [1], who assumed that all sentences that include an entity pair, which exhibits a relationship in a KB, express the same relation. However, this assumption comes with noisy labels, especially when the KB is not directly related to the domain at hand. Multi-instance learning methods were proposed to alleviate the issue, by conducting relation classification at the bag level, with a bag including instances that mention the same entity pair [2], [3].

B. NEURAL RELATION EXTRACTION

While the performance of the above approaches heavily relies on handcrafted features (POS tags, named entity tags, morphological features, etc.), the advent of neural networks (NNs) in RE set the focus on model architecture. Initially, a CNN-based method was proposed by [24] to automatically capture the semantics of sentences, while PCNN [25] became the common architecture to embed sentences. PCNN is used in several approaches that handle DS noisy patterns, such as intra-bag attention [5], inter-bag attention [26], soft labeling [27], [28] and adversarial training [29], [30]. Moreover, Graph-CNNs proved an effective way to encode syntactic information from text [8].

The latest development of pre-trained LMs relying on transformer architecture [19] has shown to capture semantic and syntactic features better [13], with [31] proving that pretrained LMs significantly improve the performance in text classification tasks, prevent overfitting, and increase sample efficiency. Moreover, works [32], [33] that fine-tune the pre-trained LM models (most of them BERT [18]) have shown that simple NNs built on top of pretrained transformer-based models improve performance. Meanwhile, DISTRE model [11] extended GPT [13] to the DS setting by incorporating a multi-instance training mechanism, proving that pre-trained LMs provide a stronger signal for DS than specific linguistic and side-information features [8]. To this extent, we take advantage of the knowledge that these models carry to capture label embeddings and boost RE.

C. SIDE INFORMATION

Apart from model architecture, several methods propose additional information to further reduce noise. For example, RESIDE [8] model uses relation phrases and incorporates Freebase-specific entity types achieving state-of-the-art precision at higher recall values, whereas [9], [10] use entity descriptors to enhance entity and label embeddings, respectively.

D. SUB-PARSES OF INPUT

Shortest-dependency path (SDP) [14] has been proved important in reducing irrelevant to RE words, while preserving the sub-path of the sentence that connects the two entities with their least common ancestor's parent (STP) [7] can further reduce the noise within sentences. Contrary to [7], who shape the final STP sequence by re-assigning the participating tokens into their original sequence order, we preserve the tokens' order within the STP to maintain the emerged grammar information.

E. LABEL EMBEDDING

Label embeddings aim to embed labels in the same space with word vectors. The idea comes from computer vision, with [28] to introduce them in text classification and [10] to use them as attention-mechanism over relational tokens in distantly-supervised RE. We make use of the TransE [17] model to shape label embeddings as the entities' distance in BERT's vector space, and we show that their use both as a feature and as attention over sentences significantly improves RE.

VI. CONCLUSION

We presented a novel transformer-based relation extraction model for distant supervision. REDSandT manages to acquire high-informative instance and label embeddings and is efficient at handling the noisy labeling problem of DS. REDSandT captures high-informative embeddings for RE by fine-tuning BERT on a RE-specific structured input that focuses solely on relational arguments, including the sub-tree connecting the entities along with entities' types. Then, it utilizes these vectors to encode label embeddings, which are also used as attention mechanism over instances to reduce the effect of less-informative tokens. Finally, relation extraction occurs at bag-level by concatenating label and weighted instance embeddings. Extensive experiments on the NYT-10 and GDS datasets illustrate REDSandT's effectiveness over existing baselines in current literature. Precisely, REDSandT manages to recognize relations that other methods fail to detect, including relations in the long-tail. Future work includes an investigation of whether additional information, such as entity descriptors, influence REDSandT's performance and to what extent, while also whether the special token embeddings can act as global embeddings for RE.

REFERENCES

[1] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, ''Distant supervision for relation extraction without labeled data,'' in *Proc. Joint Conf. 47th Annu. Meeting ACL 4th Int. Joint Conf. Natural Lang. Process. AFNLP (ACL-IJCNLP)*, col. 2, 2009, pp. 1003–1011.

[2] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, vol. 6323. 2010, pp. 148–163.
[3] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics: Hum. Lang. Technol.*, vol. 1, 2011, pp. 541–550.
[4] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "Multi-instance multi-label learning for relation extraction," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.*, 2012, pp. 455–465.
[5] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2016, pp. 2124–2133.
[6] Z. He, W. Chen, Z. Li, M. Zhang, W. Zhang, and M. Zhang, "SEE: Syntax-aware entity embedding for neural relation extraction," in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 5795–5802.
[7] T. Liu, X. Zhang, W. Zhou, and W. Jia, "Neural relation extraction via inner-sentence noise reduction and transfer learning," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Stroudsburg, PA, USA, 2018, pp. 2195–2204.
[8] S. Vashishth, R. Joshi, S. S. Prayaga, C. Bhattacharyya, and P. Talukdar, "RESIDE: Improving distantly-supervised neural relation extraction using side information," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 1257–1266.
[9] H. She, B. Wu, B. Wang, and R. Chi, "Distant supervision for relation extraction with hierarchical attention and entity descriptions," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8.
[10] L. Hu, L. Zhang, C. Shi, L. Nie, W. Guan, and C. Yang, "Improving distantly-supervised relation extraction with joint label embedding," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, p. 3821.
[11] C. Alt, M. Hübner, and L. Hennig, "Fine-tuning pre-trained transformer language models to distantly supervised relation extraction," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1388–1398.
[12] J. Moreira, C. Oliveira, D. Macedo, C. Zanchettin, and L. Barbosa, "Distantly-supervised neural relation extraction with side information using BERT," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–7.
[13] A. Radford and T. Salimans. (2018). *Improving Language Understanding by Generative Pre-Training*. [Online]. Available: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf
[14] Y. Xu, L. Mou, G. Li, Y. Chen, H. Peng, and Z. Jin, "Classifying relations via long short term memory networks along shortest dependency paths," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1785–1794.
[15] Y. Liu, K. Liu, L. Xu, and J. Zhao, "Exploring fine-grained entity type constraints for distantly supervised relation extraction," in *Proc. 25th Int. Conf. Comput. Linguistics, COLING: Tech. Papers*, 2014, pp. 2107–2116.
[16] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2016, pp. 1715–1725.
[17] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2787–2795.
[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805
[19] A. Vaswani, G. Brain, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, K. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[20] S. Jat, S. Khandelwal, and P. Talukdar, "Improving distantly supervised relation extraction using word and entity based attention," 2018, *arXiv:1804.06987*. [Online]. Available: http://arxiv.org/abs/1804.06987
[21] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.
[22] M. Craven and J. Kumlien, "Constructing biological knowledge bases by extracting information from text sources," *Proc. 7th Int. Conf. Intell. Syst. Mol. Biol.*, 1999, pp. 77–86.
[23] R. Snow, D. Jurafsky, and A. Y. Ng, "Learning syntactic patterns for automatic hypernym discovery," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005, pp. 1297–1304.
[24] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proc. 25th Int. Conf. Comput. Linguistics, Proc. COLING: Tech. Papers*, 2014, pp. 2335–2344.
[25] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1753–1762.
[26] Z.-X. Ye and Z.-H. Ling, "Distant supervision relation extraction with intra-bag and inter-bag attentions," in *Proc. Conf. North*, vol. 1, 2019, pp. 2810–2819.
[27] T. Liu, K. Wang, B. Chang, and Z. Sui, "A soft-label method for noise-tolerant distantly supervised relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1790–1795.
[28] G. Wang, C. Li, W. Wang, Y. Zhang, D. Shen, X. Zhang, R. Henao, and L. Carin, "Joint embedding of words and labels for text classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 2321–2331.
[29] Y. Wu, D. Bamman, and S. Russell, "Adversarial training for relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1778–1783.
[30] P. Qin, W. Xu, and W. Y. Wang, "DSGAN: Generative adversarial training for distant supervision relation extraction," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics, Long Papers*, vol. 1, 2018, pp. 496–505.
[31] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2018, pp. 328–339.
[32] P. Shi and J. Lin, "Simple BERT models for relation extraction and semantic role labeling," 2019, *arXiv:1904.05255*. [Online]. Available: http://arxiv.org/abs/1904.05255
[33] X. Han and L. Wang, "A novel document-level relation extraction method based on BERT and entity information," *IEEE Access*, vol. 8, pp. 96912–96919, 2020.

DESPINA CHRISTOU received the B.S. degree from the University of Macedonia, Thessaloniki, Greece, in 2015, and the M.S. degree in artificial intelligence from The University of Edinburgh, U.K., in 2016. She is currently pursuing the Ph.D. degree in natural language processing with the Aristotle University of Thessaloniki (AUTH), Greece.

Her current research interests include deep learning models for knowledge-centered NLP tasks, such as relation extraction, entity linking, and reasoning. She is interested in investigating deep learning models under the peculiarities of creative writing by studying the above tasks on literature corpora. She is involved in two national projects dealing with semantic analysis and metadata enrichment of modern Greek corpora, while she has worked as a Machine Learning Scientist and a Consultant in the private sector.

GRIGORIOS TSOUMAKAS received the degree in computer science from the Aristotle University of Thessaloniki (AUTH), Greece, in 1999, the M.S. degree in artificial intelligence from The University of Edinburgh, U.K., in 2000, and the Ph.D. degree in computer science from AUTH, in 2005.

He is currently an Associate Professor of machine learning and knowledge discovery with the School of Informatics, AUTH. His research expertise focuses on supervised learning techniques (ensemble methods and multi-target prediction) and text mining (semantic indexing, keyphrase extraction, and summarization). He has published more than 100 research articles and according to Google Scholar, he has more than 13,000 citations and an H-index of 44. He is an Advocate of applied research that matters and has worked as a machine learning and data mining developer, researcher, and consultant in several national, international, and private sector funded research and development projects. In February 2019, he co-founded Medoid AI, a spin-off company of the Aristotle University of Thessaloniki, developing custom AI solutions based on machine learning technology.

Dr. Tsoumakas is a Senior Member of the ACM and an Action Editor of the *Data Mining and Knowledge Discovery* journal.