Rehoboam at the NTCIR-15 SHINRA2020-ML Task

by

Tushar Abhishek, Ayush Agarwal, Anubhav Sharma, Vasudeva Varma, Manish Gupta

in

The 15th NTCIR (2019 - 2020) Evaluation of Information Access Technologies
: 1
-6

Tokyo,JAPAN

Report No: IIIT/TR/2020/-1

Centre for Language Technologies Research Centre
International Institute of Information Technology
Hyderabad - 500 032, INDIA
December 2020

Rehoboam at the NTCIR-15 SHINRA2020-ML Task

Tushar Abhishek
IIIT-Hyderabad, India
tushar.abhishek@research.iiit.ac.in

Ayush Agarwal
Delhi Technological University, India
ayush286@gmail.com

Anubhav Sharma
IIIT-Hyderabad, India
anubhav.sharma@research.iiit.ac.in

Vasudeva Varma
IIIT-Hyderabad, India
vv@iiit.ac.in

Manish Gupta*
IIIT-Hyderabad, India
manish.gupta@iiit.ac.in

ABSTRACT

Maintaining a unified ontology across various languages is expected to result in effective and consistent organization of Wikipedia entities. Such organization of the Wikipedia knowledge base (KB) will in turn improve the effectiveness of various KB oriented multi-lingual downstream tasks like entity linking, question answering, fact checking, etc. As a first step toward a unified ontology, it is important to classify Wikipedia entities into consistent fine-grained categories across 30 languages. While there is existing work on fine-grained entity categorization for rich-resource languages, there is hardly any such work for consistent classification across multiple low-resource languages. Wikipedia webpage format variations, content imbalance per page, imbalance with respect to categories across languages make the problem challenging. We model this problem as a document classification task. We propose a novel architecture, RNN_GNN_XLM-R, which leverages the strengths of various popular deep learning architectures. Across ten participant teams at the NTCIR-15 Shinra 2020-ML Classification Task, our proposed model stands second in the overall evaluation.

Keywords: Document Classification, Named-Entity Recognition, XLM Roberta, Multi-lingual Modeling, RNN_GNN_XLM-R

TEAM NAME

Rehoboam (RH312)

TASK

Shinra 2020-ML Classification Task

1 INTRODUCTION

Wikipedia consists of a large number of entity-centric articles. Entity mining on such comprehensive resource has been useful in many natural language processing (NLP) tasks. To maximize the use of such knowledge, resources created from Wikipedia need to be structured for inference, reasoning, or any other purposes in many NLP applications. The current structured knowledge bases such as DBpedia, Wikidata, Freebase, YAGO, and Wikidata among others, are created mostly by bottom-up crowdsourcing, which may cause a significant amount of undesirable inconsistency in the structure of the knowledge base.

To resolve the issues and with a final goal to structure the knowledge in Wikipedia and create the KB as accurately as possible, Shinra ML Task was defined as to classify 30 language Wikipedia entities into 219 categories defined in Extended Named Entity (ENE) (ver.8.0)[1] (a four-layer ontology for names, time, numbers and concepts), using categorized Japanese Wikipedia pages and the inter-language links to the corresponding pages in target languages.

Language	Entity Name	English Translation	Fine-grained category
Japanese	東京都	Tokyo	Province
Hindi	प्रशांति निलयम	Prasanthi Nilayam	Worship_Place
Russian	Звонящий в полночь	Caller at midnight	Broadcast_Program
Spanish	Sala de crisis de la Casa Blanca	White House Crisis Room	Facility_Part
French	Sneaker Pimps	Sneaker Pimps	Show_Organization

Table 1: Entity Categorization Examples

A few examples of such entities with their fine-grained categorization are shown in Table 1.

Recently, several Transformer [15]-based multi-lingual models across a wide range of NLP tasks have been proposed. This includes models for multi-lingual translation, summarization, question generation, sentiment analysis, etc. In this work, we focus on multi-lingual text categorization. Models like mBERT [7], XLM [10], XLM-R [6] and Unicoder [8], InfoXLM [4] are therefore relevant to our problem. However, to the best of our knowledge, this is the first work on multi-lingual categorization of Wikipedia entities using Transformer-based models.

We initially experimented with mono-lingual entity categorization for Hindi using various models like MPAD (Message Passing Attention Networks for Document Understanding) [12], MAGNET (Multi-Label Text Classification using Attention-based Graph Neural Network) [13], DTMT (Deep Transition RNN-based Architecture for Neural Machine Translation) [11] and a named entity recognition (NER) based model. Next, we investigated the performance of multi-lingual models like mBERT and XLM-R on entity categorization for 29 languages[1]. Finally, we propose a novel multi-lingual approach, RNN_GNN_XLM-R, which helps us get the best reported results.

Across 30 languages, our proposed model achieves a micro-F1 average score of 73.7. Across ten participant teams at the NTCIR-15 Shinra 2020-ML Classification Task [14], our proposed model stands second in the overall evaluation.

*The author is also an applied scientist at Microsoft

[1] All 30 languages except for Greek because of input dataset format differences

We discuss details of the dataset in Section 2. We discuss our baseline methods and our best proposed method, RNN_GNN_XLM-R, in Section 3. We present detailed evaluation results in Section 4. Finally, we conclude with a brief summary in Section 5.

2 DATASET

2.1 Original Dataset

The original dataset comprises on 5.07 million Wikipedia entities pages categorized into one of the 219 categories across 30 languages. For more details, please refer the overview of SHINRA2020-ML Task [14].

2.2 Preprocessing

We use the Wikipedia Cirrus Dump provided for each language for training, validation and testing. We use the 'text' section of the dump for experimenting with our models as an input. We perform masking of numerical values, removing white-spaces and punctuation characters. We also remove characters of other languages from the 'text' field, and the hyperlinks.

2.3 Taxonomy

The dataset has been annotated in a hierarchy of four levels. Each data sample can have multiple labels. The label can belong to any of the four levels. The topmost level consists of five categories Name, Timex, Numex, Concept, and Ignored. Name, Timex and Numex are further sub-categorized to three more levels while the last two topmost categories are leaf nodes in the type hierarchy.

2.4 Multi-lingual dataset selection

We carefully selected a set of 3.8M entity pages covering a subset of 18 languages out of 30 to compose multi-lingual dataset for faster training. To maintain language diversity, we used syntactic word orders as the primary criteria of selection and the number of training data available in each language as secondary criteria. Within the syntactic word ordering, only subject-verb-object (SVO) and subject-object-verb (SOV) order were preferred as they cover most of the world's spoken languages. Obtained multi-lingual dataset was randomly shuffled without replacement to maintain a proper mix of different languages in the training batch. We used a stratified split ratio of 80:20 to create training and validation datasets. Languages selected for creation of multi-lingual datasets are listed in Table 2

SVO order		SOV order	
Languages	**Training Data**	**Languages**	**Training Data**
Chinese	267,107	Dutch	199,983
English	439,354	German	274,732
French	318,828	Hindi	30,547
Finnish	144,750	Hungarian	120,295
Italian	270,295	Korean	190,807
Polish	225,552	Persian	169,053
Portuguese	217,896	Turkish	111,592
Russian	253,012		
Spanish	257,835		
Swedish	180,948		
Vietnamese	116,280		

Table 2: Languages selected for creation of multi-lingual dataset along with training data statistics.

3 FINE-GRAINED ENTITY-TYPE CLASSIFICATION METHODS

We tried the following state-of-the-art methods as our baselines. Since most (~98%) of the entity pages (especially in Hindi) have single labels, we experiment with Message Passing Attention Networks for Document Understanding (MPAD) [12] which is a state-of-the-art approach for single label document classification. Next, we experiment with multi-label text classification methods like Attention-based Graph Neural Network (MAGNET) [13]. Further, since the task is very entity-centric, we designed an entity-specific CNN-RNN model. All of the above approaches used Gated Recurrent Units (GRUs) [5], so we moved to the DTMT encoder approach which further acts as an enhancement over our RNN models. Finally, we propose a new architecture, RNN_GNN_XLM-R, for the task which is a combination of RNN, GNN and Transformer modules. In the following, we provide a brief description of these models.

3.1 MPAD

Message Passing Attention Networks for Document Understanding (MPAD) [12] represent documents as word co-occurrence networks. They used GNNs (graph neural networks) with an enhanced COMBINE step to create intermediate representation for each node in the graph. These node representations then pooled in the READOUT step to create document representation. MPAD performs best on 5 out 10 standard text classification datasets and has competitive results with the state-of-the-art on others.

3.2 DTMT

Meng et al. [11] enhance the RNN based Neural Machine Translation (NMT) by increasing the transition depth between consecutive hidden states and build a novel Deep Transition RNN-based Architecture for Neural Machine Translation. It reinforces the hidden-to-hidden transition with multiple non-linear transformations and additionally captures linear transformation path throughout this deep transition to avoid the gradient vanishing problem. We used the bi-directional encoder from this architecture and passed the final hidden state of the final encoder token to a linear layer to classify sentences into multi-label classes.

3.3 MAGNET

Multi-Label Text Classification using Attention-based Graph Neural Network (MAGNET) [13] uses feature matrix and correlation matrix obtained from GNNs in order to capture dependencies between the labels and classifiers for the downstream task. Each classification label is represented as nodes in the graph and dependencies among them are learned through GNNs implicitly. It achieves state-of-the-art performance over 5 different multi-label classification datasets.

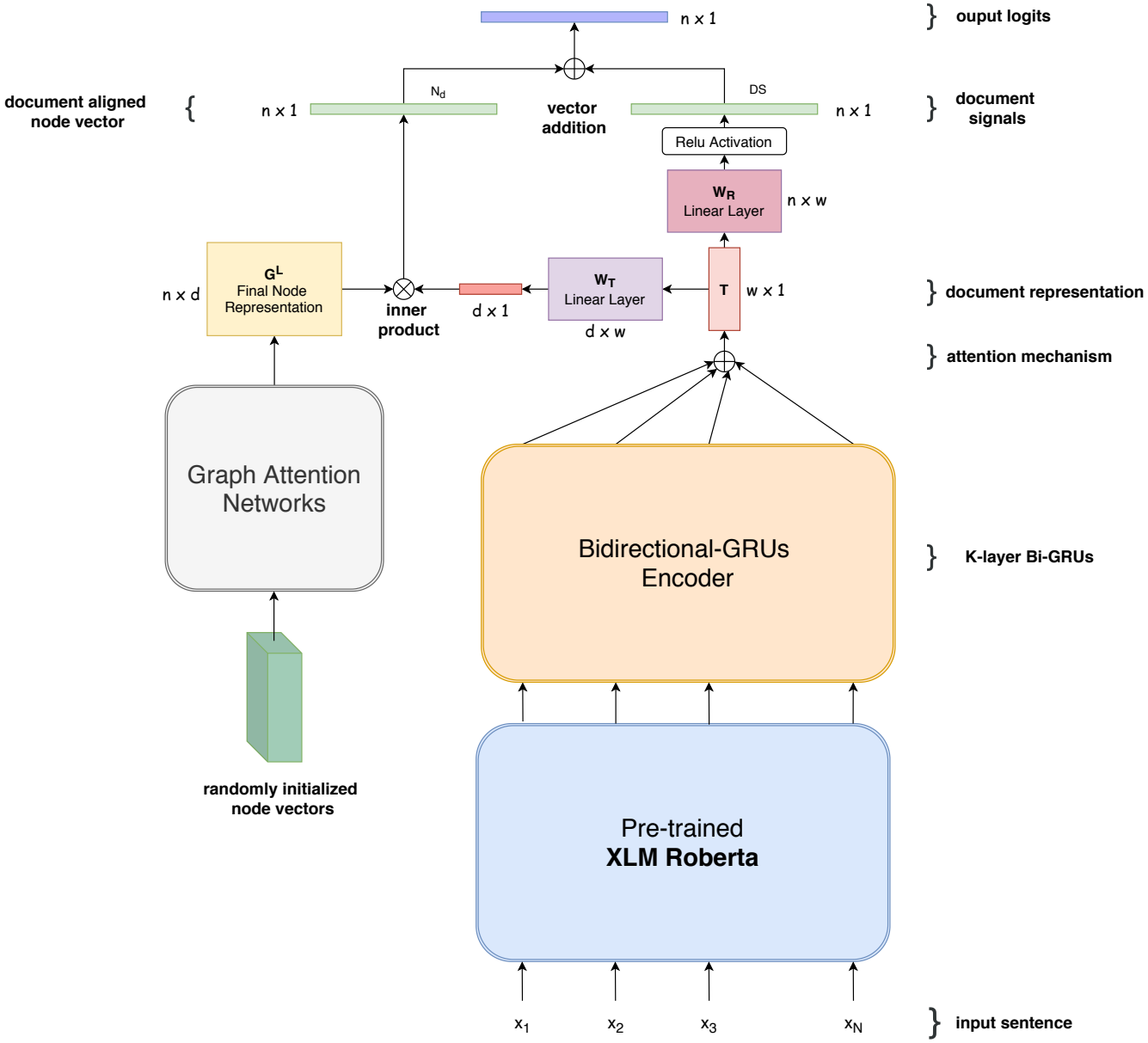

Figure 1: Architecture of our proposed system, RNN_GNN_XLM-R

3.4 NER Based Model

During the task, we also experimented with a model based on named entities present in the text of a Wikipedia Page. We extract the named entities and the entity type of each one of them from the 'text' field of the Wikipedia dump. We then conducted an experimentation study with a CNN-RNN architecture to combine the text, named entity, and entity type for each Wikipedia article. We trained an embedding layer for the named entities and another for the named entity types. We fine-tune the 300D fastText [9] word embeddings for our input text. For every input text token, we thus have a 900D vector (300D fastText, 300D entity embedding and 300D entity type embedding). This 900D vector was transformed using 3 CNN layers to obtain a 100D representation for each token. These were then passed as input to an LSTM whose last layer was connected to an output softmax layer for final prediction.

3.5 Multi-lingual Models

We also experimented with popular pre-trained multi-lingual Transformer architectures. We fine-tuned mBERT-base [7] (12 layers) and XLM-R-base [6] (12 layers). The vector representation of the [CLS] token from last layer is attached to a linear classification head and

the model is trained over the multi-lingual dataset mentioned in Section 2.4.

3.6 Proposed Method: RNN_GNN_XLM-R

Each Wikipedia article is represented as sequences of words of length m, $D' = [b_1, b_2, \ldots, b_m]$ where b_i represented the i^{th} word. Input sub-word token is obtained after passing document D' to sentence-piece tokenizer, to get $D = [x_1, x_2, \ldots, x_n]$ where x_i represents the i^{th} subword. The document D is passed to XLM-Roberta and the output from its last layer is used as input feature representation of each sub-word vector $U = [u_1, u_2, \ldots, u_n]$. Input representation U is then fed to a K-layer attentional bidirectional-GRU to get enhanced contextual representation, and the output vectors from both directions are concatenated. Similar to [13], our model learns the relatedness among the labels through Graph Attention Networks (GAT) [16]. To preserve document level signals for classification we pass the document vector T through a non-linear transformation with ReLU activation. Final score for each label is obtained by adding it with node representation, and then connecting to an output softmax layer. Figure 1 shows the overall architecture of the proposed model.

4 EXPERIMENTS AND RESULTS

4.1 Evaluations

4.1.1 Metric. The model is expected to classify each page into one or more of the 219 taxonomy categories correctly. If the estimated category is not an exact match, the model does not get a score for that. The model is evaluated for the multi-label classification using the micro-averaged F1 measure, i.e., the harmonic mean of micro-averaged precision and micro-averaged recall.

4.1.2 Leaderboard. We test the performance of our baseline models as described in Sections 3.1-3.5 in addition to the proposed model,RNN_GNN_XLM-R, on the leaderboard [2] of the Shinra2020 ML Task. The leaderboard dataset consists of 2000 data samples whose distribution was independent of the training dataset. The organizers provided the performance of the model using the micro-averaged F1 metric. Results from the leaderboard are described in Tables 3 and 4.

4.1.3 Final evaluation by Shinra organisers. For the final evaluations, we were required to submit the model predictions on the entire test set for any number of languages. The detailed statistics of the entire test set are unknown to the participants. The model performance was evaluated by the organizers using micro-averaged F1 metric. We present the results obtained on the test set for our proposed approach in Section 4.3. Results using the entire test set for final evaluation are described in Table 5.

Methods	Training Data	**Micro-Averaged F1-score**
MPAD [12]	Mono-lingual	59.5
DTMT [11]	Mono-lingual	61.9
MAGNET [13]	Mono-lingual	61.3
NER based model (Ours)	Mono-lingual	61.1
mBERT-base [7]	Multi-lingual	66.5
XLM Roberta [6]	Multi-lingual	68.1
RNN_GNN_XLM-R (Ours)	Multi-lingual	**72.1**

Table 3: The F1-score are reflected from Shinra leaderboard submission for Hindi evaluation dataset.

4.2 Hindi dataset results (leaderboard)

On the Hindi dataset we tried all the approaches (both monolingual and multi-lingual). The models mentioned in Sections 3.1-3.4 are trained on the Hindi language dataset only. For implementing the MPAD we extracted all the datapoints with single label and tested it on them. Whereas for MAGNET, NER based Model and DTMT encoder, all the datapoints were considered while testing. Table-3 shows results only for Hindi language obtained from Shinra leaderboard submission. mBERT-base and XLM Roberta have been trained on the multi-lingual training dataset but evaluated on the Hindi leaderboard dataset.

The table clearly validates that RNN involving approaches don't perform that well as compared to Transformer-based approaches. Also we observe that MAGNET outperforms MPAD. The enhanced RNN encoder in DTMT outperforms the traditional RNN encoder models (MPAD and MAGNET). Surprisingly, our entity-aware NER based model does not perform that well. As expected, multi-lingual training outperforms mono-lingual training. Our proposed system, RNN_GNN_XLM-R, is better in comparison to other Transformer-based models. This is because our model not just uses text semantics but also leverages correlations between class labels for improved predictions.

4.3 Multi-lingual results (leaderboard)

We populated the results from the Shinra leaderboard [2] for our proposed system, RNN_GNN_XLM-R, and multi-lingual baselines along with some of the other leading participant models. We trained RNN_GNN_XLM-R on 18 languages and did zero shot inference on the remaining 11 languages – Arabic, Bulgarian, Catalan, Czech, Danish, Hebrew, Indonesian, Norwegian, Romanian, Ukrainian and Thai. Table 4 is the official leaderboard result dated 30^{th} September 2020.

4.4 Official evaluations results

We have populated the official results for six languages. We submitted the predictions to the Shinra[3] evaluation system only for those six languages. Table-5 contains the best micro-averaged F1-score of our model as well as some of the other competitive models in the respective languages.

5 CONCLUSION

In this paper, we studied the problem of multi-lingual fine-grained entity type classification. We investigated the effectiveness of various state-of-the-art monolingual RNN encoders as well as multilingual Transformer-based models. We proposed a novel adaptation of the XLM-R model where we augment the XLM-R base with RNN and GNN modules. We experimented with these models on the datasets provided by the NTCIR-15 Shinra2020 ML task organizers. We evaluated the effectiveness of our models on Hindi

Languages	Models					
	HUKB	PribL	uomfj	mBERT-base	XLM-R-base	RNN_GNN_XLM-R
English (en)	51.2	73.9	74.5	72.1	74.4	**76.0**
Spanish (es)	56.7	75.1	73.2	70.7	73.1	**75.4**
French (fr)	48.9	73.5	70.4	73.3	73.1	**74.3**
German (de)	61.1	**75.8**	71.3	72.5	70.8	75.3
Chinese (zh)	60.1	75.4	72.5	70.5	73.0	**75.8**
Russian (ru)	55.0	**74.5**	69.8	70.9	70.3	73.4
Portugese (pt)	51.9	71.0	69.5	69.5	68.9	**72.4**
Italian (it)	49.0	73.4	72.4	70.0	71.9	**74.3**
Arabic (ar)	51.0	70.7	70.2	67.1	69.6	**72.3**
Indonesian (id)	54.0	-	71.2	72.9	74.4	**74.5**
Turkish (tr)	58.2	73.2	69.5	72.6	**73.9**	**73.9**
Dutch (nl)	60.3	**73.8**	70.8	72.3	72.9	72.9
Polish (pl)	61.6	**76.6**	73.5	71.8	74.2	75.1
Persian (fa)	60.9	-	73.8	70.0	72.9	**73.9**
Swedish (sv)	59.6	-	69.7	70.9	69.9	**73.4**
Vietnamese (vi)	61.7	72.2	72.1	74.4	**75.4**	74.7
Korean (ko)	53.8	**74.6**	72.3	71.2	71.2	70.8
Hebrew (he)	52.2	-	68.6	68.9	68.0	**72.1**
Romanian (ro)	53.2	-	69.8	72.3	71.7	**75.0**
Norwegian (no)	50.2	71.7	70.8	68.5	70.5	**72.2**
Czech (cs)	53.3	69.2	68.1	68.0	69.5	**72.2**
Ukrainian (uk)	58.2	69.6	70.5	70.5	70.0	**71.5**
Hindi (hi)	44.1	60.5	65.9	66.5	68.1	**72.1**
Finnish (fi)	51.9	-	73.1	**73.2**	73.0	72.9
Hungarian (hu)	54.9	-	71.4	71.7	72.8	**74.6**
Danish (da)	52.7	72.2	73.2	70.4	71.4	**73.8**
Thai (th)	60.3	-	50.3	66.8	70.9	**75.2**
Catalan (ca)	43.9	-	71.6	**72.1**	70.9	71.9
Bulgarian (bg)	56.7	-	74.7	75.2	73.9	**76.7**
Average	54.7	72.5	70.5	70.9	71.7	**73.7**

Table 4: Micro-averaged F1 score comparison across various models on the Leaderboard test set. HUKB, PribL and uomfj are the other top performing teams on the Shinra Challenge leaderboard. mBERT-base and XLM-R-base are our other baselines. RNN_GNN_XLM-R is our best proposed model. Best results for each language are highlighted in bold.

Languages	Teams				
	ousia	uomfj	LIAT	PribL	RH312 (ours)
Bulgarian	-	**83.07**	75.2	-	82.13
French	**81.01**	78.21	76.88	78.52	80.31
Hindi	69.75	66.67	16.49	-	**71.7**
Indonesian	-	**78.51**	72.44	-	77.55
Thai	76.36	65.02	49.58	-	**76.77**
Turkish	-	**84.85**	77.19	84.36	83.28

Table 5: Micro-averaged F1 score comparison across various models on the official evaluation dataset for 6 languages. ousia, uomfj, LIAT and PribL are the other top performing teams on the Shinra Challenge leaderboard. RH312 (i.e., RNN_GNN_XLM-R) is our best proposed model. Best results for each language are highlighted in bold.

dataset as well as on datasets for six languages. Our proposed model, RNN_GNN_XLM-R, outperforms other methods we experimented with. We believe this is because XLM Roberta extracts rich semantics from the document, attentional-RNN module enhances the contextual information and the GNN module helps learn classification label correlations leading to improved generalization. The leaderboard and final evaluation results demonstrate the effectiveness of our models.

REFERENCES

[1] Extended Named Entity. *Homepage.* https://ene-project.info/

[2] Shinra leaderboard. *Homepage.* https://www.nlp.ecei.tohoku.ac.jp/projects/AIP-LB/task/shinra2020-ml

[3] Shinra project. *Homepage.* http://shinra-project.info/?lang=en

[4] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. InfoXLM: An Information-Theoretic Framework for Cross-Lingual Language Model Pre-Training. *arXiv preprint arXiv:2007.07834* (2020).

[5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase

representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[6] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *ACL*. Association for Computational Linguistics, 8440–8451.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT (1)*. Association for Computational Linguistics, 4171–4186.

[8] Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. *arXiv preprint arXiv:1909.00964* (2019).

[9] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651* (2016).

[10] Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291* (2019).

[11] Fandong Meng and Jinchao Zhang. 2019. DTMT: A novel deep transition architecture for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 224–231.

[12] Giannis Nikolentzos, Antoine J.-P. Tixier, and Michalis Vazirgiannis. 2020. Message Passing Attention Networks for Document Understanding. In *AAAI*. AAAI Press, 8544–8551.

[13] Ankit Pal, Muru Selvakumar, and Malaikannan Sankarasubbu. 2020. MAGNET: Multi-Label Text Classification using Attention-based Graph Neural Network. In *ICAART (2)*. SCITEPRESS, 494–505.

[14] Satoshi Sekine, Masako Nomoto, Kouta Nakayama, Asuka Sumida, Koji Matsuda, and Maya Ando. 2020. Overview of SHINRA2020-ML Task. In *Proceedings of the NTCIR-15 Conference*.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[16] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph Attention Networks. *CoRR* abs/1710.10903 (2017).