Template and Slot filling using Generative Transformers and Deep Learning networks

Anupam Dwivedi
Enrollment no. IIT2020198
B-TECH Information Technology
Indian Institute of Information Technology
Prayagraj, India

***Abstract*—Template filling is generally tackled by a pipeline of two separate supervised systems – one for role-filler extraction and another for template/event recognition. Since pipelines consider events in isolation, they can suffer from error propagation. A framework is introduced, based on end-to-end generative transformers for this task (i.e., GTT). It naturally models the dependence between entities both within a single event and across the multiple events described in a document. It is further shown that the framework specifically improves performance on documents containing multiple events. Also different combinations of encoders and decoders are used for the task specifically RNNs and CharCNNs. Different pre-trained transformers such as RoBERTa and DistilBERT are used and evaluated and compared against models in previous literature. Github Project link: https://github.com/a-n-u-p-a-m/NLPCOURSEPROJECT**

***Index Terms*—Transformers, Template, Tokenization, Encoder-decoder, Role-filler Entity extraction(REE)**

I. Introduction

To naturally model between-event dependencies across a document for template filling, a framework is proposed called "GTT" based on generative transformers. The framework is built upon GRIT, which tackles role-filler entity extraction (REE), but not template/event recognition. GRIT performs REE by "generating" a sequence of role-filler entities, one role at a time in a prescribed manner. For the template-filling setting, the GRIT approach is extended to include tokens representing event types as part of the input sequence. The decoder is further modified to attend to the event type tokens, allowing it to distinguish among events and associate event types to each role-filler entity that it generates. The models also use different types of neural networks such as RNNs and CharCNNs as well as different transformers as the base model for the encoder-decoder architecture. The model is evaluated on the MUC-4 (1992) template filling task.

II. Dataset Description

- MUC-4 consists of 1,700 documents with associated templates. We split the dataset as: 1,300 documents for training, 200 documents (TST1+TST2) as the development set and 200 documents (TST3+TST4) as the test set.
- There are 6 event types (i.e., kidnapping, attack, bombing, robbery, arson, forced work stoppage) in MUC-4..

III. Literature Review

There is previous work available that used both transformers as well as different Deep Learning models for the task of template filling. The task of this project is to evaluate models defined in this project and compare them with the existing models defined in the following papers.

A. *Paper-1: Template Filling with Generative Transformers*

- **Author Names: Xinya Du, Alexander M. Rush and Claire Cardie**
- **Year of publication: 2021**
- The paper presents a generative model for template filling, a task that involves filling structured templates with information extracted from a given document.
- The model transforms template filling into a sequence generation problem and uses a generative transformer-based approach to solve it.
- The source sequence is prepared by adding event/template type tokens to the words of the document, along with separator tokens.
- The target sequence consists of the concatenation of template extractions, separated by the separator token, with each template's event type and role-filler entities represented as sub-sequences.
- The model uses a BERT model as an encoder and decoder, and pointer decoding is used to select the next token in the generated sequence.
- The paper introduces several decoding constraints to ensure that the generated output is structured and valid, including downweighting factor, decoding cutoff stop, and a constraint to ensure that the pointers for the start and end token for one entity are in order.
- The authors evaluate their model on the MUC-4 dataset and show that it outperforms several strong baselines, achieving state-of-the-art results.
- They also perform ablation studies to analyze the effect of different components of their model and demonstrate the importance of each component.
- The defined framework substantially outperforms the baseline extraction models in precision, recall and F1, with approximately a 4 percent F1 increase over the end-to-end baselines. It outperforms the GRIT-PIPELINE system by around 3 percent F1.

- The results demonstrate that the defined framework more often predicts the correct event type, performs better on PERPIND and PERPORG, and achieves slightly worse performance with GRIT-PIPELINE on roles that appear later in the template (i.e., TARGET and VICTIM). It was also found that DYGIE++ performs better on TARGET, mainly due to its high precision in role assignment for spans.
- When the number of gold events in the document is smaller (E = 1, 2), our approach performs on par with the pipeline-based and DYGIE++ baselines. However, as E grows larger, the baselines' F1 drop significantly (e.g., over -10 percent as E grows from 2 to 3).

B. Paper-2: Document-Level Event Role Filler Extraction using Multi-Granularity Contextualized Encoding

- **Author Names: Xinya Du and Claire Cardie**
- **Year of publication: 2020**
- The paper is about a new approach for extracting event role fillers from a document.
- The approach uses a hierarchical model that combines different levels of representation to generate the final output.
- The different levels of representation include word-level, sentence-level, and document-level representations.
- The word-level representations are obtained using pre-trained contextualized embedding models.
- The sentence-level representations are obtained using a bi-directional LSTM.
- The document-level representations are obtained by encoding the document as a sequence of sentences using another bi-directional LSTM.
- The final output is generated using a joint inference model that takes into account all the representations at different levels.
- The approach was evaluated on the MUC-4 dataset.
- The approach achieved state-of-the-art results on both datasets.
- The authors conclude that their approach is effective for document-level event role filler extraction and can be easily adapted to other related tasks.
- The end-to-end neural readers can achieve nearly the same level or significantly better results than the pipeline systems. Although the described models rely on no hand-designed features, the contextualized double-sentence reader and paragraph reader achieves nearly the same level of F-1 compared to Cohesion Extraction (CE), judging by the head noun matching metric. The multi-granularity reader performs significantly better (approx 60) than the prior state-of-the-art.
- Contextualized embeddings for the sequence consistently improve the neural readers' performance. The results show that the contextualized k-sentence readers all outperform their non-contextualized counterparts, especially when k ¿ 1. The trends also exhibit in the per event role analysis. To notice, the transformers' parameters are frozen during training (fine-tuning yields worse results).
- When increasing the input context from a single sentence to two sentences, the reader has a better precision and lower recall, resulting in no better F-1; When increase the input context length further to the entire paragraph, the precision increases and recall remains the same level, resulting in higher F-1; When we keep increasing the length of input context, the reader becomes more conservative and F-1 drops significantly. All these indicate that focusing on the local (intra-sentence) and broader (paragraph-level) context are both important for the task.

C. Paper-3: Entity, Relation, and Event Extraction with Contextualized Span Representations

- **Author Names: David Wadden, Ulme Wennberg, Yi Luan and Hannaneh Hajishirzi**
- **Year of publication: 2019**

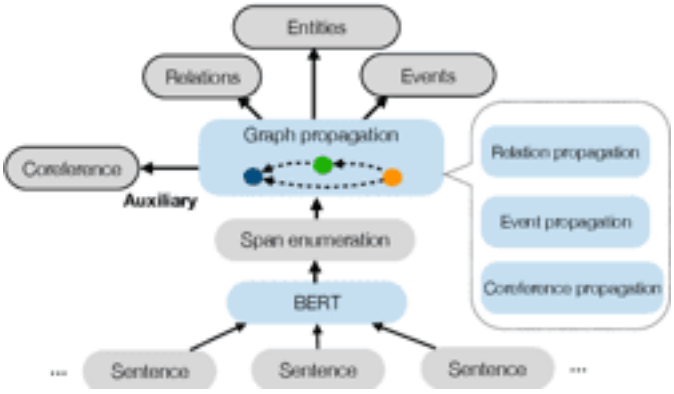

Fig. 1. Architecture of the model

- We examine the capabilities of a unified, multitask framework for three information extraction tasks: named entity recognition, relation extraction, and event extraction.
- Token encoding: DYGIE++ uses BERT for token representations using a "sliding window" approach, feeding each sentence to BERT together with a size-L neighborhood of surrounding sentences.
- Span enumeration: Spans of text are enumerated and constructed by concatenating the tokens representing their left and right endpoints, together with a learned span width embedding
- Span graph propagation: A graph structure is generated dynamically based on the model's current best guess at the relations present among the spans in the document. Each span representation

$$g_j^t$$

is updated by integrating span representations from its neighbors in the graph according to three variants of graph propagation.

- Multi-task classification: The re-contextualized representations are input to scoring functions which make predictions for each of the end tasks.A two-layer feedforward neural net is used (FFNN) as the scoring function.
- Four different datasets are used for experimentation: ACE05, SciERC, GENIA and WLPC.
- The framework establishes a new state-of-the-art on all three high-level tasks, and on all subtasks except event argument identification. Relative error reductions range from 0.2 - 27.9state of the art models.
- Coreference propagation (CorefProp) improves named entity recognition performance across all three domains. The largest gains are on the computer science research abstracts of SciERC, which make frequent use of long-range coreferences, acronyms and abbreviations. CorefProp also improves relation extraction on SciERC.
- Relation propagation (RelProp) improves relation extraction performance over pretrained BERT, but does not improve fine-tuned BERT.
- Event propagation is not helpful due to the asymmetry of the relationship between triggers and arguments.

*D. **Paper-4: Joint Extraction of Events and Entities within a Document Context***

- **Author Names: Bishan Yang and Tom Mitchell**
- **Year of publication: 2016**
- In this paper,a novel approach is proposed that models the dependencies among variables of events, entities, and their relations, and performs joint inference of these variables across a document. The goal is to enable access to document-level contextual information and facilitate contextaware predictions.
- The learning problem is firstly decomposed into three tractable subproblems: (1) learning the dependencies between a single event and all of its potential arguments, (2) learning the cooccurrence relations between events across the document, and (3) learning for entity extraction.
- A joint inference framework is introduced that combines probabilistic models of within-event structures, event-event relations, and entity extraction for joint extraction of the set of entities and events over the whole document.
- Extensive experiments are conducted on the Automatic Content Extraction (ACE) corpus, and show that the described approach significantly outperforms the state-of-the-art methods for event extraction and a strong baseline for entity extraction.
- The WITHINEVENT model, which explicitly models the trigger-argument dependencies and argument-role-entity-type dependencies, outperforms the MaxEnt pipeline, especially in event argument extraction. This shows that modeling the trigger-argument dependencies is effective in reducing error propagation. Comparing to the state-of-the-art event extractor JOINTBEAM, the improvements introduced by WITHINEVENT are substantial in both event triggers and event arguments.
- JOINTEVENTENTITY provides the best performance among all the models on all evaluation categories. It boosts both precision and recall compared to WITHINEVENT. This demonstrates the advantages of JOINTEVENTENTITY in allowing information propagation across event mentions and entity mentions and making more context-aware and semantically coherent predictions.
- JOINTEVENTENTITY also extracts entity mentions. Its output is compared with the output of a strong entity extraction baseline CRFENTITY. JOINTEVENTENTITY introduces a significant improvement in recall and F1.

IV. Methodology

- The model architecture of BERT is a multi-layer bidirectional Transformer encoder based on the original Transformer model. The input representation is a concatenation of WordPiece embeddings, positional embeddings, and the segment embedding. Specially, for single sentence classification and tagging tasks, the segment embedding has no discrimination. A special classification embedding ([CLS]) is inserted as the first token and a special token ([SEP]) is added as the final token. The BERT model is pre-trained with two strategies on large-scale unlabeled text, i.e., masked language model and next sentence prediction. The pre-trained BERT model provides a powerful context-dependent sentence representation and can be used for various target tasks, i.e., intent classification and slot filling, through the finetuning procedure, similar to how it is used for other NLP tasks.
- BERT can be easily extended to a joint intent classification and slot filling model. Based on the hidden state of the first special token ([CLS]), denoted h1, the intent is predicted as: y i = softmax(Wih1 + b i), (1) For slot filling, we feed the final hidden states of other tokens h2, . . . , hT into a softmax layer to classify over the slot filling labels.
- For joint intent detection and slot filling is illustrated,no encoder side, we use a bidirectional RNN. Bidirectional RNN has been successfully applied in speech recognition and spoken language understanding. We use LSTM as the basic recurrent network unit for its ability to better model long-term dependencies comparing to simple RNN. The decoder is a unidirectional RNN. Again, we use an LSTM cell as the basic RNN unit.
- For joint modeling of intent detection and slot filling, we add an additional decoder for intent detection (or intent classification) task that shares the same encoder with slot filling decoder. During model training, costs from both decoders are back-propagated to the encoder. The intent decoder generates only one single output which is the intent class distribution of the sentence, and thus alignment is not required.
- In the proposed model, a bidirectional RNN (BiRNN) reads the source sequence in both forward and backward directions. We use LSTM cell for the basic RNN unit.

Slot label dependencies are modeled in the forward RNN. Similar to the encoder module in the above described encoder-decoder architecture, the hidden state hi at each step is a concatenation of the forward state fhi and backward state bhi, hi = [fhi, bhi]. Each hidden state hi contains information of the whole input word sequence, with strong focus on the parts surrounding the word at step i. This hidden state hi is then combined with the context vector ci to produce the label distribution, where the context vector ci is calculated as a weighted average of the RNN hidden states h = (h1, ..., hT).

- The model transforms template filling into a sequence generation problem and uses a generative transformer-based approach to solve it. The source sequence is prepared by adding event/template type tokens to the words of the document, along with separator tokens. The target sequence consists of the concatenation of template extractions, separated by the separator token, with each template's event type and role-filler entities represented as sub-sequences. The model uses a BERT model as an encoder and decoder, and pointer decoding is used to select the next token in the generated sequence.
- Proposition of a template-based method for NER, treating NER as a language model ranking problem in a sequence-to-sequence framework, where original sentences and statement templates filled by candidate named entity span are regarded as the source sequence and the target sequence, respectively. For inference, the model is required to classify each candidate span based on the corresponding template scores.

V. Results

- On the ATIS dataset the BERT model has 0.9787 intent accuracy, 0.9559 slot F1 and 0.8824 sentence accuracy. On the ATIS dataset the ALBERT model has 0.9764 intent accuracy, 0.9578 slot F1 and 0.8813 sentence accuracy. On the ATIS dataset the DistilBERT model has 0.9776 intent accuracy, 0.9550 slot F1 and 0.8768 sentence accuracy.
- On the MUC-4 dataset it is concluded that for 'Victim' the precision is the best and for 'target' it is the worst. Moreover the generative transformer provides a total precision of 0.6419, a recall of 0.4736 and a F1 score of 0.5450. This accuracy is increased for the testing cases of the dataset.

VI. Future Scope

In this project I have studied and implemented different techniques for different Information Extraction tasks. For future work there is a possibility of checking the models in some Out-Of-Distribution(OOD) cases. Also an ensemble can be used for the task instead of a single encoder-decoder. There is also the possibility of checking different transformer networks for the tasks and fine-tuning them.

VII. Conclusion

In this project I studied, analyzed and designed some techniques and models for different information extraction tasks. These tasks include Named Entity Recognition, Template Filling, Event extraction, etc. I studied different transformers and deep learning models and used them for the mentioned tasks. In the end I concluded that using generative transformers is better for the task of template filling over traditional deep learning models.

References

[1] Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana - Practical Natural Language Processing-A Comprehensive Guide to Building Real-World NLP Systems-O'Reilly Media, Inc. (2020)

[2] Template-Based Information Extraction without the Templates by Nathanael Chambers and Dan Jurafsky

[3] Newton Spolaor, Everton Alvares Cherman, Maria Carolina Monard, and Huei Diana Lee. 2013. A comparison of multi-label feature selection methods using the problem transformation approach. Electronic Notes in Theoretical Computer Science, 292:135– 151.

[4] GRIT: Generative Role-filler Transformers for Document-level Event Entity Extraction by Xinya Du Alexander M. Rush Claire Cardie

[5] Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling by Bing Liu , Ian Lane