A Multi-task Event and Argument Trigger Detection in Hindi using POS Tagging as an Auxiliary Task

Anonymous ACL submission

Abstract

The event, as well as argument trigger detection, are essential sub-tasks of the event extraction system. Lots of effort has been devoted to improving the performance of trigger detection systems. But, the effect of low-level tasks like Parts-of-Speech (POS) tagging as an auxiliary task in multi-task learning of event and argument trigger detection has not been understood well in literature. In our current work, we propose a BERT-based multi-task architecture that learns a shared representation from two sequence labeling tasks, trigger detection (both event and argument), and POS tagging in a multi-task setup using POS tagging as an auxiliary task. We show that our proposed approach achieves a significant performance boost as compared to single-task models. We perform our experiment in the Hindi language, unlike previously proposed works.

1 Introduction

Multi-task learning (MTL), as the name suggests, is to learn multiple semantically related tasks in parallel. The effectiveness of MTL has been demonstrated in various NLP tasks (Ruder, 2017) by the research community. MTL accomplishes a better-generalized knowledge by leveraging the task-specific knowledge of semantically related tasks (Caruana, 1997). However, apart from some theoretical recommendations (Caruana, 1998; Baxter, 2000; Ben-David and Schuller, 2003), we don't have a well-understood understanding in MTL about the preconditions for tasks to be regarded as related to another task. Finally, Xue et al. (2007) says that similar tasks have a close parameter vector. More recently, (Alonso and Plank, 2016) observe that morphosyntactic tasks like POS tagging with low kurtosis and relatively high entropy values work better as an auxiliary task. In our current work, we try to improve the performance of event and argument trigger detection (henceforth trigger detection) through MTL setup. Similar to (Sahoo et al., 2019), we learn both event trigger and argument trigger detection as a single task. We observe that in most cases, event and argument triggers have a particular POS tag sequence. In both the example, single token triggers like 'बाढ़' (baadh) and 'भूस्खलन' (bhooskhalan) have POS tags 'NN'. For multi-token event trigger "बर्फीले तूफान" (barpheele toophaan) has POS bigram tag sequence "JJ NN". Similarly in most of the cases, place arguments ("पूर्वोत्तर केरल" (poorvottar keral) and "चीन के एंहुई प्रांत" (cheen ke enhuee praant)) contains proper nouns (NNP) in their POS n-gram sequence ("JJ NNP" and "NNP PSP NNPC NN" respectively) and casualty arguments ("10 की मौत" (10 kee maut) and "13 लोग" (13 log)) contains quantifiers (QC). It also makes sense to use the POS tag as an auxiliary task since POS information is used as a feature in the trigger detection task. Previous studies (Gildea and Palmer, 2002; Collobert and Weston, 2008; Alonso and Plank, 2016) also suggests using low-level task as an auxiliary task to improve semantic tasks.

- **Example Sentence 1** : **पूर्वोत्तर केरल** में **बाढ़** और **भूस्खलन** में **10 की मौत**
- **Transliteration** : **poorvottar keral** mein **baadh** aur **bhooskhalan** mein **10 kee maut**
- **Translation** : **10 killed** in **floods** and **landslides** in **northeast Kerala**
- **Output 1 (Trigger Detection)** : **B-Arg I-Arg** O **B-Event** O **B-Event** O **B-Arg I-Arg I-Arg**
- **Output 2 (POS Tagging)** : **JJ NNP** PSP **NN** CC **NN** PSP **QC PSP NN**

- **Example Sentence 2** : **चीन के एंहुई प्रांत** में **13 लोग बर्फीले तूफान** में मारे गए हैं
- **Transliteration** : **cheen ke enhuee praant** mein **13 log barpheele toophaan** mein maare gae hain
- **Translation** : **13 people** killed in **snow storm** in **Anhui province of China**

- **Output 1 (Trigger Detection)** : **B-Arg I-Arg I-Arg I-Arg** O **B-Arg I-Arg B-Event I-Event** O O O O
- **Output 2 (POS Tagging)** : **NNP PSP NNPC NN** PSP **QC NN JJ NN** PSP VM VAUX VAUX

2 Related Work

Event and argument trigger detection has been a prevalent task in the research community for a long time. Initial feature-based approaches (Ji and Grishman, 2008; Liao and Grishman, 2010; Hong et al., 2011; Riedel and McCallum, 2011a,b; Li et al., 2013; Venugopal et al., 2014) and neural network based approaches (Nguyen and Grishman, 2015; Chen et al., 2015; Nguyen and Grishman, 2016; Ghaeini et al., 2016; Feng et al., 2018; Nguyen et al., 2016; Yang and Mitchell, 2016; Liu et al., 2018b) are used for event detection. External resources like uses *FreeBase* (Chen et al., 2017), *FrameNet* (Liu et al., 2016), explicit annotated argument information (Liu et al., 2017) and dependency relationships (Sha et al., 2018; Nguyen and Grishman, 2018) are also used for this task. Techniques like attention mechanism is used by (Liu et al., 2018a; Orr et al., 2018) to enhance performance. Multi-task learning has been explored previously by various studies on multiple NLP tasks, including NER, POS Tagging, chunking and SRL to name a few (Caruana, 1997; Collobert and Weston, 2008; Plank et al., 2016). Alonso and Plank (2016) evaluates a set of semantic sequence labeling tasks as main tasks and morphosyntactic sequence labeling tasks as auxiliary tasks. Similarly, we learn semantic task (trigger detection) as the main task and morphosyntactic (POS tagging) as the auxiliary task. To the best of our knowledge, this combination of main-auxiliary task pairs in any language, including Hindi has never been studied.

3 Task Description and Contribution

Formally we define the task as follows : Given a Hindi sentence (*S*) of form w_1, w_2, w_3,..., w_n, the task is to identifying event and argument triggers from the sentence along with the POS tags of each tokens of the same sentence. The primary goal of this experiment is to investigate whether learning POS tags simultaneously along with trigger detection have any positive influence or not. We briefly describe each task below.

Event and Argument Trigger Detection : An event trigger is a word or phrase that indicates a real-world event. Attributes and participants like time, place, agent of an event are referred to as arguments. We use IOB (Ramshaw and Marcus, 1999) tagging format for tagging the event and argument triggers as they can be a multi-word expression.

Parts-of-Speech Tagging : Parts-of-Speech Tagging is the technique of assigning a word in a sentence to a corresponding part-of-speech tag based on its context and definition. We can summarize the main contribution of this paper as follows :

- We propose a multi-task architecture using multilingualBERT, which learns two sequence labeling tasks viz. trigger detection and POS tagging simultaneously.
- Through experiments, we show and prove our hypothesis that learning POS tags as an auxiliary task significantly improves the performance of trigger detection tasks.

4 Methodology

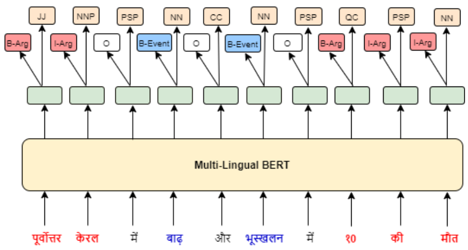

Figure 1: Proposed architectures for multi-task sequence labeling

In all the models, we use pre-trained cased multilingual $BERT_{BASE}$ (henceforth BERT) (Devlin et al., 2018) as encoder. For an input sentence *S* = [w_1, w_2, w_3,..., w_n], BERT encoder takes input sentence of the form [[*CLS*], w_1, w_2, w_3,..., w_n, [*SEP*]. We obtain *H* = [h_1, h_2, h_3, . . . , h_n] where each h_i is the contextualized vector representation of corresponding word w_i of input sentence *S*. We formulate both the problem as a token classification problem where we assign each input word w_i a structured label. In single task baseline-1, we classify each contextualized vector h_i into one of the output labels. For baseline-2, we learn an extra linear layer for each input token w_i. We pass each contextualized vector h_i into a linear layer to obtain another vector representation e_i. Finally, we classify each e_i into one of the output labels. We

Tasks	Multi-Task			Baseline-2 (Single Task)			Baseline-1 (Single Task)			Support
Tags	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	
B-Event	0.69	0.74	**0.71**	0.69	0.70	0.70	0.71	0.68	0.69	1,976
I-Event	0.65	0.59	**0.62**	0.70	0.53	0.60	0.68	0.54	0.60	1,299
B-Arg	0.63	0.57	**0.60**	0.66	0.51	0.58	0.70	0.47	0.56	4,883
I-Arg	0.64	0.56	**0.60**	0.64	0.54	0.58	0.67	0.51	0.58	17,770
O	0.86	0.89	**0.88**	0.85	0.90	**0.88**	0.85	0.92	**0.88**	67,631

Table 1: Results for Trigger Detection. *P*, *R* and *F1* stands for *Precision*, *Recall* and *F1-Score* respectively

present a multi-task architecture (Figure-1) for simultaneously learning triggers and POS tags. Similar to baseline-2, here also we learn an extra linear layer on the top of BERT output. All of the layers except the output layer use hard parameter sharing. Separate output layers for each task are added to the outermost layer. We employ the *Cross-entropy* loss as our loss function and the *Softmax* activation function for token classification for both the task. We calculate the loss function for multi-task model by using the formula : $\lambda_1 \mathcal{L}_{Trig} + \lambda_2 \mathcal{L}_{POS}$, where $\mathcal{L}_{Trig}$, $\mathcal{L}_{POS}$ are the loss functions for trigger detection and POS tagging respectively. λ_1 and λ_2 are learnable weighting parameters.

5 Experimental Setup

	Train	Test	Dev
# sentences	19,845	5,800	4,459
# event triggers (ET)	6,824	1,983	1,577
# total tokens : ET	11,353	3,290	2,657
average length : ET	1.66	1.66	1.68
# argument triggers (AT)	16,159	4,897	3,903
# total tokens : AT	76,869	22,680	18,285
average length : AT	4.76	4.63	4.68

Table 2: Dataset Statistics

5.1 Dataset

We use "A Platform for Event Extraction in Hindi" dataset (Sahoo et al., 2020) in our experiments. The dataset is annotated for the tasks: event and argument trigger detection and classification, and event-argument linking. We only consider event and argument trigger labels and follow the same train-dev-test split of the dataset described in the paper. We also incorporate the POS tags into the dataset. To minimize manual effort, we use an available Hindi POS tagger (shallow parser)[1] to automatically assign synthetic POS labels to each tokens. Table-2 shows the dataset statistics.

[1]The shallow parser examines a sentence for morphological analysis, POS tagging, Chunking, etc. It is managed by the LTRC IIIT-Hyderabad and developed by a group of institutions

5.2 Experiment Overview

We carry out our experiments on five models: two single task trigger detection models (Baseline-1 and Baseline-2 models), two single tasks POS tagging models (Baseline-1 and Baseline-2 models), and one multi-task model. We run each model thrice with three different seed values (42,142,333) and report the average of all three runs for each model. We run each experiment with a maximum sequence length of 250 for 15 epochs. The training batch size is 16. We fine-tune all the models using the AdamW optimizer with learning rate $5 * 10^{-5}$. We select the best model based on its performance in the validation set. We also use global gradient clipping with norm value 1.

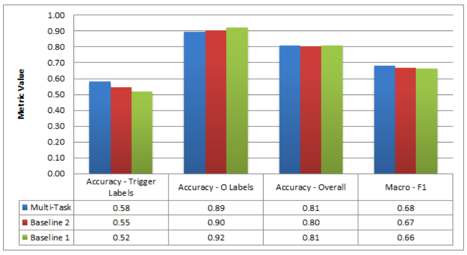

Figure 2: Comparison of results for various metrics for all the models. Each of the values is calculated by taking the average of three runs

6 Experimental Results and Analysis

Table-1 shows the results for the trigger detection results. We compare the results of both the single-task models with the proposed multi-task model and find out that the performance of the trigger detection task of the multi-task model is improved compared to both the baseline model. However, the performance of the POS tagging task is slightly deteriorated as compared to the baselines. For the POS tagging task, baseline-2 gives the best performance. For trigger detection task, we observe performance improvement for trigger labels (*B-Event*, *I-Event*, *B-Arg* and *I-Arg* tags) where the improvement is 1.5% - 3.5% as compared to baseline-2 and 3% -7% as compared to baseline-1 in terms

Tasks	Multi-Task			Baseline 2 (Single Task)			Baseline 1 (Single Task)			Support
Tags	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	
JJ	0.95	0.94	0.94	0.95	0.94	**0.95**	0.95	0.94	0.94	5,336
NNP	0.88	0.88	0.88	0.89	0.88	0.88	0.89	0.89	**0.89**	6,461
PSP	1.00	1.00	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	19,639
NN	0.96	0.96	**0.96**	0.96	0.97	**0.96**	0.96	0.96	**0.96**	23,232
CC	0.99	0.99	**0.99**	0.99	0.99	**0.99**	0.99	0.99	**0.99**	2,941
QC	0.99	0.99	**0.99**	0.99	0.99	**0.99**	0.99	0.99	**0.99**	3,280
VM	0.98	0.98	**0.98**	0.98	0.98	**0.98**	0.98	0.98	**0.98**	10,873
VAUX	0.99	0.98	0.98	0.99	0.98	**0.99**	0.99	0.98	**0.99**	7,746
PRP	0.99	0.98	**0.98**	0.99	0.98	**0.98**	0.99	0.98	**0.98**	2,761
NNPC	0.81	0.80	0.81	0.82	0.81	**0.82**	0.82	0.81	0.81	2,901
QF	0.98	0.98	**0.98**	0.99	0.98	**0.98**	0.98	0.98	**0.98**	1,055
QCC	0.96	0.99	0.97	0.97	0.99	**0.98**	0.97	0.99	**0.98**	198
DEM	0.96	0.98	**0.97**	0.96	0.98	**0.97**	0.96	0.98	**0.97**	894
NST	1.00	1.00	**1.00**	1.00	1.00	**1.00**	1.00	1.00	**1.00**	1,760
RP	0.98	0.98	**0.98**	0.98	0.98	**0.98**	0.98	0.98	**0.98**	1,583
NNC	0.80	0.80	0.80	0.82	0.81	**0.82**	0.83	0.81	**0.82**	1,590
RB	0.95	0.96	0.95	0.96	0.96	**0.96**	0.96	0.96	**0.96**	450
QO	0.94	0.97	0.95	0.95	0.97	**0.96**	0.96	0.96	**0.96**	147
UNK	0.78	0.82	0.80	0.85	0.83	**0.84**	0.84	0.84	**0.84**	57
NEG	0.99	1.00	**1.00**	0.99	1.00	**1.00**	0.99	1.00	**1.00**	452
INTF	0.98	0.92	0.95	0.98	0.94	**0.96**	0.98	0.94	**0.96**	142
JJC	0.62	0.52	0.57	0.69	0.55	**0.61**	0.69	0.55	0.60	14
PRPC	0.79	1.00	0.88	0.79	1.00	0.88	0.83	1.00	**0.91**	5
WQ	0.87	0.95	0.91	0.89	0.95	**0.92**	0.88	0.96	**0.92**	37
SYM	0.67	0.22	0.33	0.72	0.44	**0.52**	0.67	0.22	0.33	3
INJ	1.00	1.00	**1.00**	0.83	1.00	0.89	1.00	1.00	**1.00**	1

Table 3: Results for POS Tagging . *P*, *R* and *F1* stands for *Precision*, *Recall* and *F-Score* respectively

of *F1-Score*. Further investigation reveals that the proposed multi-task model attains superior performance by improving the *recall* value of the model across all the tag labels except the *O*-label. For *O*-label, we observe same *F1-score* for all the models. A deeper investigation reveals that baseline-1 performs the best for *O*-label. We calculate the accuracy for three variants, one considering only the trigger labels and discarding *O*-labels, second considering only the *O*-labels and discarding the triggers labels, and third considering all the tag labels. Figure -2 shows that the proposed multi-task model performs better than both the baselines for the trigger detection accuracy by 5% (compared to baseline-2) - 11% (compared to baseline-1). However, baseline-1 is performing better in detecting *O*-label than the others models. The overall accuracy of the multi-task model is similar to baseline-1 and slightly better baseline-2. This is due to the superior performance of the baseline models in detecting *O*-labels. As the number of *O*-labels is much greater than the others labels (refer Table 1), we also compare the macro-averaged *F1-score*, for all the three models for trigger detection. Form Figure2, we can observe that the macro-averaged *F1-score* is significantly better ($p < 0.05$ in paired t-test) in the multi-task model than the other two baseline models.

7 Conclusions and Future Work

We assess the influence of POS tagging as an auxiliary task on event and argument trigger detection. We observe that the learning POS tagging in parallel with trigger detection improves trigger detection performance, though the overall accuracy is almost similar for all the cases. Due to the significantly higher number of *O*-labels, the dataset has an imbalance tag distribution. So, we consider macro-averaged *F1-score* in which the proposed multi-task model performance is significantly high than the baseline models. In this paper, we investigate the influence of POS tags in detecting the trigger words in the Hindi dataset and show significant improvement. However, we hypothesize that POS tagging, a low-level task, would help predict the event's and argument's class labels. We want to widen our scope of research in various directions: choosing additional main and additional tasks and additional datasets of different languages.

Ethical declaration

The dataset utilised in this paper is publicly available and has only been used for scholarly purposes. There are no further matters to declare.

References

Héctor Martínez Alonso and Barbara Plank. 2016. When is multitask learning effective? semantic sequence prediction under varying data conditions. *arXiv preprint arXiv:1612.02251*.

Jonathan Baxter. 2000. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198.

Shai Ben-David and Reba Schuller. 2003. Exploiting task relatedness for multiple task learning. In *Learning theory and kernel machines*, pages 567–580. Springer.

R Caruana. 1998. Multitask learning. autonomous agents and multi-agent systems.

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28(1):41–75.

Yubo Chen, Shulin Liu, Xiang Zhang, Kang Liu, and Jun Zhao. 2017. Automatically labeled data generation for large scale event extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 409–419.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 167–176.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Xiaocheng Feng, Bing Qin, and Ting Liu. 2018. A language-independent neural network for event detection. *Science China Information Sciences*, 61(9):092106.

Reza Ghaeini, Xiaoli Fern, Liang Huang, and Prasad Tadepalli. 2016. Event nugget detection with forward-backward recurrent neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 369–373.

Daniel Gildea and Martha Palmer. 2002. The necessity of parsing for predicate argument recognition. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 239–246.

Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1127–1136. Association for Computational Linguistics.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. *Proceedings of ACL-08: HLT*, pages 254–262.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 73–82.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 789–797. Association for Computational Linguistics.

Jian Liu, Yubo Chen, Kang Liu, and Jun Zhao. 2018a. Event detection via gated multilingual attention mechanism. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shulin Liu, Yubo Chen, Shizhu He, Kang Liu, and Jun Zhao. 2016. Leveraging framenet to improve automatic event detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2134–2143.

Shulin Liu, Yubo Chen, Kang Liu, Jun Zhao, et al. 2017. Exploiting argument information to improve event detection via supervised attention mechanisms.

Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018b. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 300–309.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 365–371.

Thien Huu Nguyen and Ralph Grishman. 2016. Modeling skip-grams for event detection with convolutional neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 886–891.

Thien Huu Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Walker Orr, Prasad Tadepalli, and Xiaoli Fern. 2018. Event detection with neural networks: A rigorous empirical evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 999–1004.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. *arXiv preprint arXiv:1604.05529*.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Sebastian Riedel and Andrew McCallum. 2011a. Fast and robust joint models for biomedical event extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1–12. Association for Computational Linguistics.

Sebastian Riedel and Andrew McCallum. 2011b. Robust biomedical event extraction with dual decomposition and minimal domain adaptation. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 46–50. Association for Computational Linguistics.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Sovan Kumar Sahoo, Saumajit Saha, Asif Ekbal, and Pushpak Bhattacharyya. 2019. A multi-task model for multilingual trigger detection and classification. In *Proceedings of the 16th International Conference on Natural Language Processing*, pages 160–169.

Sovan Kumar Sahoo, Saumajit Saha, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A platform for event extraction in hindi. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2241–2250.

Lei Sha, Feng Qian, Baobao Chang, and Zhifang Sui. 2018. Jointly extracting event triggers and arguments by dependency-bridge rnn and tensor-based argument interaction. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Deepak Venugopal, Chen Chen, Vibhav Gogate, and Vincent Ng. 2014. Relieving the computational bottleneck: Joint inference for event extraction with high-dimensional features. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 831–843.

Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. 2007. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research*, 8(1).

Bishan Yang and Tom Mitchell. 2016. Joint extraction of events and entities within a document context. *arXiv preprint arXiv:1609.03632*.