Language-Agnostic Text Processing for Information Extraction

Karthika Vijayan[1] and Oshin Anand[2]

[1,2] Data Science Team, Sahaj AI, Bangalore, India
[1]karthikav@sahaj.ai
[2]oshina@sahaj.ai

Abstract

Information extraction from multilingual text for conversational AI generally implements natural language understanding (NLU) using multiple language-specific models, which may not be available for low resource languages or code mixed scenarios. In this paper, we study the implementation of multilingual NLU by development of a language agnostic processing pipeline. We perform this study using the case of a conversational assistant, built using the RASA framework. The automatic assistants for answering text queries are built in different languages and code mixing of languages, while doing so, experimentation with different components in an NLU pipeline is conducted. Sparse and dense feature extraction accomplishes the language agnostic composite featurization of text in the pipeline. We perform experiments with intent classification and entity extraction as part of information extraction. The efficacy of the language agnostic NLU pipeline is showcased when (i) dedicated language models are not available for all languages of our interest, and (ii) in case of code mixing. Our experiments delivered accuracies in intent classification of 98.49%, 96.41% and 97.98% for same queries in English, Hindi and Malayalam languages, respectively, without any dedicated language models.

Keywords

Information Extraction, Multilingual Text, Natural Language Understanding, Language Agnostic Processing, Composite features

1. Introduction

Natural language understanding (NLU) deals with comprehension of human languages by computers [1]. The NLU enables an AI to interpret the meaning from text by performing several tasks such as text categorisation, entity extraction, intent recognition, etc. These tasks act as basic building blocks of conversational assistants.

NLU is usually accomplished with specific language models catering to individual languages. The language modelling follows procedures like n-grams computation [1], or adopts recursive neural architectures [2, 3] or the current state-of-the-art transformers [4]. Constructing a conversational assistant for a popular language like English is fairly straightforward, as multiple resources and pre-trained models are readily available [5, 6, 7, 8, 9]. This is not the case with multilingual settings, where several languages are from the low-resources category.

For a multilingual application, building language models for all languages under consideration is arduous [10, 11]. Also, the task of language modelling is challenging for low resource languages. One of the earlier and widely used strategy for accomplishing multilingual capability was through translation, where language model is available only in one language that caters to all other languages of interest by using translation in the front-end and back-end. This method requires accurate translation models and results in error accumulation in the process pipeline.

David C. Wyld et al. (Eds): AIAA, DPPR, DSA, ICSS, IOTE, NLPTA, WEST, CACIT, BMLI- 2022
pp. 129-140, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.122310

Another widely accepted solution for performing NLU without explicit language modelling is by using pre-trained resources and fine-tuning them for specific use-cases [5, 6, 11, 12, 13, 14]. However, this approach will reap immediate advantages only if, there exist a pre-trained model for the language of our interest. An alternate way of using an existing pre-trained model to cater to the language of our interest is via transfer learning [15, 16, 17, 18]. Yet, it requires the knowledge of structure of languages and relationship between source and target languages to use pre-trained model from a source language to perform NLP for a target language [19]. Also, the effectiveness of pre-trained models for NLU in code mixed scenarios is limited.

Recently, several advances have been made in the direction of language agnostic NLP by means of joint multilingual language models [19, 20, 21, 22, 23]. These models attempt to map the text from multiple languages to a shared representational space in terms of word/sentence embeddings, constructed on a shared vocabulary from multiple languages [19]. The embeddings of words from different languages, those have similar meanings, are represented by closer vectors. These shared representations can then undergo some language specific post-processing to address different NLP tasks in different languages. NLU from code mixed text data is identified as a far more challenging task [24]. Fine tuning, data augmentation, generation of code mixed data, etc. were proposed to use pre-trained models for NLU from code mixed text [25, 26, 27, 28, 29, 30]. Code mixing was also attempted by translation and/or transliteration to a target script and then employing pre-trained models for NLU [25, 31]. These methods are known to propagate errors from translation modules to the NLU process.

In this paper, we study the usefulness of language agnostic NLP in information extraction from different Indian languages and code mixing of languages, without any data augmentation or translation/transliteration. We develop an NLU process pipeline for the application of conversational assistants with all individual process components to be language agnostic. We choose a case study with conversational sales assistants built using the RASA framework [32], serving customer queries in English, Hindi, Malayalam and code mixing of English and Hindi. For the development of the language agnostic NLU pipeline, we carefully study and choose the process components. Later, we attempt information extraction (intent classification and entity recognition) from customer queries using the developed NLU pipeline. The studies presented in this paper show that the language agnostic NLU pipeline delivers competent performance in entity extraction and intent classification for all language scenarios under consideration. Through this study, we showcase the enabling of language understanding capability in conversational assistants using language agnostic processing. Also, we demonstrate the effectiveness of language agnostic processing in code mixing of languages written in same and different scripts.

The rest of the paper is organised as follows: Section 2 explains the conversational assistant, which is the application that we have selected to aid the study. Section 3 describes the development of the language agnostic NLU pipeline. In Section 4, we present the evaluation of NLU pipelines for different language cases and showcase their efficiency in information extraction. We summarise the inferences from this study in Section 5.

2. Case Study: A Conversational Sales Assistant

For the study of language agnostic NLU, we choose the case of a conversational assistant providing automatic sales support to customers by responding to their queries. We build conversational assistants to address queries in English, Hindi, Malayalam and code mixing of Hindi and English languages. These assistants are built using the RASA framework [32], and we experiment on the language agnostic NLU process pipeline for multiple assistants.

The RASA is an open source platform designed to build chat and voice-based automatic assistants. It helps in creating process pipelines for NLU and RASA core, using components available either from within the framework or from external libraries [32, 34]. It also provides a joint intent-entity classifier called DIET for NLU, and helps with dialogue management using stories and rules dictated by RASA policies [32, 37]. As its open source and rich in features, we chose RASA to build our case study.

The conversational sales assistant in our study is expected to receive customer queries related to a product specifications, price and other details regarding an intended purchase. Example queries in English and transliterated Hindi, received by the sales support assistant, are given in Table 1. To understand customer queries, the assistant requires NLU capability to perform information extraction from short text messages. The NLU for such conversational assistants include intent classification to identify the intent of a customer's query (eg: whether the customer aims to find the price of a product or its' specifications) and entity extraction to identify relevant details in the customer's messages (eg: which product is the customer talking about, location from which the person wants to make a purchase from, etc.).

Table 1: Sample queries for a conversational sales assistant with respect to intents. The entities are highlighted in bold fonts.

Intent	**Example query**
Greet	Hello, my name is **nameA**. Namasthe, mein **Kolkata** se hoon. *(Hindi in latin script)*
Goodbye	Bye. Thanks.
Product details query	I would like to know different options in **washing machines**. Can you tell me more about **BrandA** ? Mein **carBrandA** khareedna chaahta hoon. Uske detailed info chahiye. *(Code mixing)*
Product price query	I want to know the price of washing machine **BrandA**. **CarB** ki on road price kitna hain ?

The intent classification in our case study aims to classify customer queries into one among 4 intents, namely, *greet, goodbye, product details query* and *product price query*. In this limited query set up, we chose two simpler intents as *greet* and *goodbye*, where the user messages are of simpler constructs. On the other hand, we have chosen two more complex intents as '*product details query*' and '*product price query*', where the user messages can be long complex sentences, mentioning multiple entities. See the Table. 1 for examples.

The conversational assistant's NLU also needs to identify entities of relevance to the use case, specifically, *name* & *location* of the customer and *class* & *name of the product* being queried about. We have chosen two entities that can be considered as restricted entities and the other two can be viewed as unrestricted entities. The '*name*' and '*location*' of the customer can take a wide range of values (unrestricted). On the other hand, the entities '*product class*' and '*product name*' may take a rather restricted set of values depending upon the use-case, as a particular vendor or company generally sells a closed set of classes and variants of their products.

The intent classification and entity extraction together represent information extraction required for the conversational sales assistant in our study. Using this case study of a conversational assistant, we analyse the language agnostic NLU process pipeline for information extraction from text messages in multiple languages and code mixing of languages.

3. Language Agnostic NLU Process Pipeline

The implementation of NLU process for the conversational assistant is carried out using the RASA framework [32]. The most common components in an NLU process pipeline are text preprocessing, extracting a feature representation of text, and classification. Generally, these components are realised using strategies/models learned from text data belonging to a specific language, essentially building capability of understanding that particular language. In this study, we attempt to make all components in the NLU pipeline to be language agnostic, by making use of strategies like n-gram modelling and massively multilingual models like multilingual BERT [33]. Thus, no component in the NLU process pipeline has to be changed with a change in language of text messages.

3.1. Preprocessing and data preparation

The text data in our case study consisted of short text messages as shown in Table.1. We performed basic data cleaning and sanity checks, later, removed all punctuations. Then we labelled sentences tagging entities of our interest and categorised the sentences based on intents.

The first component in the NLU pipeline is tokenisation, which is the process of segmenting text data into tokens [1]. As most of the Indian languages and English are written with a white space separating words, we use the simplest of tokenisers in the NLU pipeline, that is the white space tokeniser. RASA supports the implementation of the tokeniser, namely, ‘WhitespaceTokenizer’, or the implementation from spaCy can be used as well [34, 7].

3.2. Featurization of text

Feature representation of tokens is the next critical component in the NLU pipeline. The most simple features that one can choose for text stem from syntax and semantics of the language. We choose to extract lexical and syntactic features for tokens in our text data. RASA supports the usage of ‘LexicalSyntacticFeaturizer’, which tags tokens with information like, whether a token appear at the beginning/ending of a sentence, whether a token is just digits, whether a token is a title, the part-of-speech of the token, etc. Tagging these information corresponding to tokens generate a sparse feature vector per token [34].

One of the earliest and commonly used language modelling strategy is the n-gram modelling. This probabilistic model essentially represents the prominence of a token in a language, by counting its occurrences in training dataset. For the NLU pipeline, we chose to represent tokens as a bag of words representation using n-grams; unigrams for words and uni-, bi-, tri- and tetra-grams for characters. The n-gram based bag of words representation also result in sparse feature vectors. The RASA an n-gram featurizer called ‘CountVectorsFeaturizer’ [34].

Additionally, one can use RASA’s ‘RegexFeaturizer’ for capturing particular patterns of words, that occur in context of specific use-cases. For example, entities following a particular naming strategy, like models of an electrical appliance or a vehicle, can be expressed as regular expressions (regex) for the featurizer to identify. The features including lexical, syntactic, bag of words and regex constitute the set of sparse features included in the NLU pipeline.

3.2.1. Dense featurization

The usage of sparse features may not result in efficient representations of tokens, as they capture only peripheral and probabilistic information from text. The dense featurization attempts to capture meaning of words, and is generally achieved through language specific word embeddings [5, 12, 13, 35].

We intent to develop a language agnostic NLU pipeline without any language-dependent component, and we utilised the pre-trained multilingual BERT (mBERT) model [21, 33]. The mBERT is a transformer model trained using an extremely large multilingual text dataset. The word embeddings from mBERT are faithful representation of meaning and context of words in the respective language. The word embeddings from multiple languages should lie close to each other if the words themselves are similar in meaning [33]. The mBERT is used for dense feature extraction from tokens, in general [38]. However, for a specific use-case, the mBERT may not have the domain knowledge or word representation in its vocabulary. The mBERT should be fine-tuned to present it with required information for successful domain transfer learning.

We have used 'bert-base-multilingual-cased' model provided by Hugging Face, supporting 104 languages, to extract dense features for tokens of our text data [36]. This model is fine-tuned while training the NLU pipeline using the customer query dataset in multiple languages. The fine-tuning is intended to provide the mBERT with context and vocabulary specific to the use-case of conversational sales assistant, even though the model has language representation of all languages under our consideration. The feature extraction is realised using 'LanguageModelFeaturizer' calling the mBERT model using the RASA framework [34].

We have used sparse, dense and a combination of sparse and dense features incrementally to represent text data. The featurization involving both sparse and dense features in the NLU pipeline is attempted to draw advantages from any complementary/supplementary information that they may have captured from text data. This is termed as composite featurization in our study, which is the significant contribution to the language agnostic NLU process pipeline.

3.3. Classification

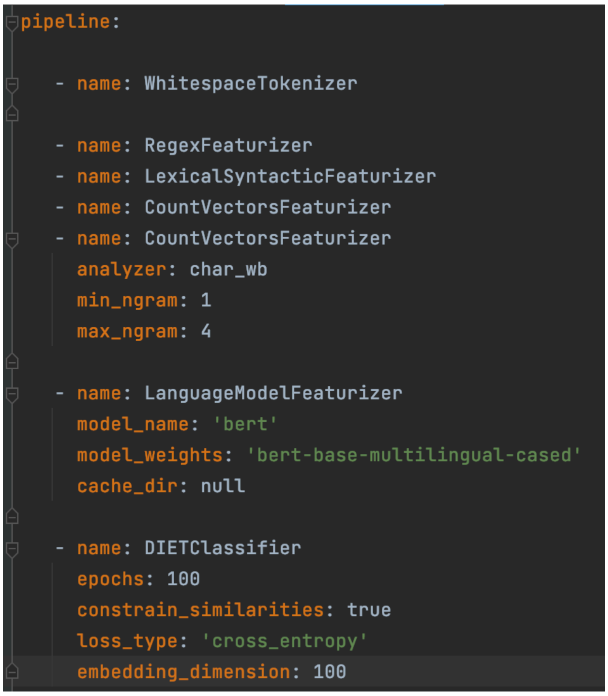
The next component in the NLU pipeline is a classifier that is intended to perform entity extraction and intent classification, thereby "understanding" the meaning of text data. We have employed RASA's DIET classifier for identifying intents and entities mentioned in Section 2. The Dual Intent Entity Transformer (DIET) is a multitask transformer model that performs intent-entity identification jointly, using a sequence model to learn from the word order and context [37]. It employs the lexical syntactic features for entity extraction, count vector features for intent classification and regex & dense features for both entity and intent identification [34].

The language agnostic NLU process pipeline developed for multilingual and code mixing languages is shown in Figure 1. The pipeline is implemented with RASA framework. The highlight of this pipeline is its truly language agnostic components and the existence of composite featurization.

4. Experimental Evaluation

We study the performance of the language agnostic processing pipeline for NLU discussed in Section 3, in understanding customer queries for a conversational sales assistant described in Section 2. We implemented the assistants for multiple languages using RASA framework [32].

The NLU process pipeline is trained using a dataset consisting of sample queries corresponding to each intent under consideration, and all entities are labelled in the query sentences. The training dataset contains 120 example queries corresponding to complex intents related to product-enquiry and at least 50 sentences for simple intents. The same query sentences are translated into multiple languages and code mixing is introduced appropriately so that the performance of NLU across all scenarios remain comparable. We experimented with sparse, dense and composite featurization in the NLU pipeline. The tokeniser and DIET classifier stay consistent in our experiments. The performance of NLU is evaluated with a test dataset of size in a 1:1 proportion with the training dataset without any overlapping sentences between training and testing. We report the accuracies of entity extraction and intent classification as measures of effectiveness of information extraction from text messages using the NLU pipeline.



```
pipeline:

  - name: WhitespaceTokenizer

  - name: RegexFeaturizer
  - name: LexicalSyntacticFeaturizer
  - name: CountVectorsFeaturizer
  - name: CountVectorsFeaturizer
    analyzer: char_wb
    min_ngram: 1
    max_ngram: 4

  - name: LanguageModelFeaturizer
    model_name: 'bert'
    model_weights: 'bert-base-multilingual-cased'
    cache_dir: null

  - name: DIETClassifier
    epochs: 100
    constrain_similarities: true
    loss_type: 'cross_entropy'
    embedding_dimension: 100
```

Figure 1. The language agnostic NLU process pipeline in RASA framework for conversational assistants.

4.1. Evaluation of entity extraction

The entity extraction performance of language agnostic NLU for multiple languages are reported in Table 2 and average performance is reported in Table 3. Generally the accuracy of extracting restricted entities, like product class and name, is considerably higher than that for unrestricted entities, like names of customers and places of their residence, as can be observed from Table 2. For the unrestricted entities, the sparse features grossly delivered better accuracies than dense features for all languages. We observed that the context in which these entities appear in customer messages is fairly consistent for the sparse featurizers to learn efficiently from the training dataset. Also, the unrestricted set of values of these entities may not appear in the normal vocabulary of a language, making it difficult for multilingual joint language models to efficiently extract word embeddings.

On the other hand, product class and product name are restricted entities, where the dense features had grossly delivered better accuracy of entity extraction, as shown in Table 2. Also when code mixing of languages occurs, the dense features outperform the sparse features by a considerable margin. The dense features are extracted using a pre-trained multilingual joint language model, which has knowledge of multiple languages apriori. The training of NLU pipeline accomplishes the fine-tuning of the multilingual model for it to obtain knowledge specific to our use-case, thus enabling them to generate efficient embeddings representing the restricted entities and code mixed text. On the contrary, the sparse features learn the context of words from the limited code mixed training dataset, hence failing to extract efficient representations of tokens resulting in inferior entity extraction in code mixing. Finally, we perform composite featurization by including both sparse and dense featurizers in the pipeline to draw advantages from both, as can be seen from Table 3.

Table 2. Accuracy (%) of entity extraction using language agnostic NLU for the languages (LN): English (EN), Hindi (HI), Malayalam (ML), code mixing of English with transliterated Hindi (CM 1) and code mixing of English with Hindi written in native script (CM 2). Notations ‘Sp’, ‘Dn’ and ‘Comp’ denote sparse, dense and composite features, respectively.

LN	Unrestricted entities						Restricted entities					
	Name			Location			Product class			Product name		
	Sp	Dn	Comp	Sp	Dn	Comp	Sp	Dn	Comp	Sp	Dn	Comp
EN	75.00	60.00	78.57	86.36	80.00	86.96	91.61	95.42	95.42	93.75	96.55	96.33
HI	66.66	50.00	75.00	88.24	76.19	90.00	88.39	90.90	95.32	86.67	90.90	94.38
ML	98.80	71.43	98.80	50.00	41.66	58.88	92.68	96.20	96.83	89.74	90.90	91.18
CM1	44.44	50.00	50.00	53.84	81.82	81.82	79.41	85.39	82.35	93.75	90.90	92.85
CM2	50.00	83.33	85.00	76.47	78.94	82.86	94.31	97.34	98.79	75.00	88.88	89.33

Table 3. Average accuracy (%) of entity extraction using language agnostic NLU for multiple language scenarios

LN \ Features	Sp	Dn	Comp
EN	86.68	82.99	89.32
HI	82.49	76.99	87.93
ML	82.81	75.05	86.42
CM1	67.86	77.03	76.75
CM2	73.95	87.12	88.99

We also performed a comparative study of language specific (LS) and language agnostic (LA) NLU pipelines for entity extraction. We utilised the NLP, tokenisation and featurization components from the pre-trained spaCy English pipeline named as ‘en_core_web_lg’, to realise the LS NLU for English language [7]. The LA NLU pipeline includes tokenisation and composite featurization. DIET classifier is used consistently in both pipelines.

We opted to perform the comparative study with pure English and English code mixed with transliterated Hindi (CM 1), as dedicated language model pipeline for English was available. The Table 4 presents the accuracy of entity extraction for pure and code mixed language scenarios with and without dedicated language modelling. It can be seen that the LA NLU process performed better than LS NLU in extracting both restricted and unrestricted entities from pure English text and code mixed English-Hindi text. The language agnostic processing pipeline is not affected by a predefined vocabulary and showcases its ability to generalise faithfully well to content that are previously unseen. Thus the LA NLU performed superior to LS NLU, as the latter has a language specific model at its core that failed to capture information from out of vocabulary words occurring in specific context of the conversational sales assistant and transliterated Hindi text.

Table 4. Accuracy (%) of entity extraction using language specific (LS) and language agnostic (LA) NLU processes.

Entity	Language	LS	LA
Name	English	60.00	78.57
	CM 1	54.54	60.00
Location	English	79.41	86.96
	CM 1	68.42	81.82
Product class	English	88.16	95.42
	CM 1	72.16	82.35
Product name	English	77.14	96.33
	CM 1	72.31	92.85

4.2. Evaluation of intent classification

We performed intent classification required for information extraction from customer queries for conversational sales assistant. We experimented with multiple featurizers in the language agnostic NLU pipeline, and the performance of intent classification is shown in Table 5. The sparse and dense features performed equivalently well in recognising intents. We observed that the sparse features performed slightly better than the dense features in identifying simpler intents, namely, greet and goodbye. The sentence structure corresponding to simpler intents and relatively simple with multiple words occurring repeatedly. This scenario is much easier for sparse featurizers to learn from.

On the other hand, the dense features had delivered slightly better accuracy in identifying the complex intents, namely, queries related to product details and price. In code mixing scenario for CM 1 and English-Hindi written in Devanagari script (CM 2), the dense features show their superiority upon sparse features. This behaviour is expected from dense featurizers, as they extend and generalise their learning to complex scenarios. The composite featurization has proven to be advantageous than using either sparse or dense features. The intent classification performance delivered by sparse, dense and composite features are given in Table 5.

Table 5. Average accuracy (%) of intent classification, using language specific (LS) and language agnostic (LA) NLU processes.

Language	LS	LA		
		Sp	Dn	Comp
English	92.62	98.49	97.74	98.49
Hindi	-	95.20	93.57	96.41
Malayalam	-	97.69	97.21	97.98
CM 1	92.23	94.88	97.71	98.25
CM 2	-	95.58	98.13	98.27

We also performed the intent classification using LS and LA NLU pipelines. We have implemented the LS NLU for English and English-transliterated Hindi code mixing (CM 1) only, as language specific model for English is available from spaCy. In line with our observation regarding entity extraction, the intent classification performance of LA NLU is better than LS NLU. Upon fine-tuning, the LA NLU is able to extend its learning to words and context specific to the use case of conversational sales assistants, whereas, the LS NLU is constricted by the vocabulary and context preset to the language for which it is trained on.

5. Conclusions

We studied the usefulness of language agnostic text processing in realising NLU for conversational sales assistants addressing Indian languages and English. We experimented with featurization components in the NLU pipeline using RASA framework for building the conversational assistants. We utilised sparse features to complement/supplement the dense features in characterising the text data efficiently, thereby achieving a composite featurization in a completely language agnostic manner. Including this composite featurization as components, we developed a language agnostic NLU pipeline for information extraction from customer queries to the conversational sales assistant.

The results of intent/entity recognition evaluation, conducted as a measure of information extraction, confirmed the effectiveness of composite featurization over either sparse or dense features alone. We conducted detailed analysis of performance of different featurizers in intent and entity recognitions, and presented valuable inferences regarding the benefits and limitations of each featurizer with respect to nature of entities and intents extracted. The language agnostic pipeline with the composite featurization delivered competent performance in NLU, clearly marking its usage for multilingual applications where dedicated language models are not available. In fact, the language agnostic processing was superior to language specific processing in NLU from both English and English-Hindi code mixing, indicating its efficiency in addressing code mixed text and content with spelling errors or out-of-vocabulary words.

Acknowledgements

We would like to thank Dr. Ravindra Babu T., global head, Data science, Sahaj AI and Prof. Gaurav Raina, Department of Electrical Engineering, Indian Institute of Technology Madras, India, for their valuable guidance. Also, we thank Mr. Sanjoy Bose, ex-Sahaj employee, for initiating the discussions leading up to this research.

References

[1] Daniel Jurafsky & James H. Martin (2000) *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* (1st ed.), Prentice Hall PTR, USA.

[2] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černocký, & Sanjeev Khudanpur (2011) "Extensions of recurrent neural network language model" IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp 5528–5531. doi https://doi.org/10.1109/ICASSP.2011.5947611

[3] Martin Sundermeyer, R. Schlüter, & H. Ney (2012) "LSTM Neural Networks for Language Modeling", INTERSPEECH 2012

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser & Illia Polosukhin (2017) "Attention is All you Need", Advances in Neural Information Processing Systems, Vol.30, CurranAssociates,Inc.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee & Kristina Toutanova (2019) "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[6] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov & Quoc V. Le (2019) "XLNet: Generalized Autoregressive Pretraining for Language Understanding" Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14.

[7] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd (2020) "spaCy: Industrial - strength Natural Language Processing in Python" https: //doi.org/10.5281/ zenodo.1212303

[8] Davis E. King. (2009) "Dlibml: A Machine Learning Toolkit" Journal of Machine Learning Research 10 (2009), 1755–1758.

[9] Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning (2018) "Universal Dependency Parsing from Scratch" Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. Association for Computational Linguistics, Brussels, Belgium, 160–170. https: //nlp.stanford.edu/pubs/qi2018universal.pdf

[10] Rafal Józefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu (2016) "Exploring the Limits of Language Modeling" CoRR abs/1602.02410 (2016). arXiv:1602.02410 http://arxiv.org/abs/1602.02410

[11] Jeremy Howard and Sebastian Ruder (2018) "Universal Language Model Fine - tuning for Text Classification." In ACL.

[12] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut (2020) "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations" International Conference on Learning Representations. https://openreview.net/ forum?id=H1eA7AEtvS

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov (2019) "RoBERTa: A Robustly Optimized BERT Pretraining Approach" CoRR abs/1907.11692 (2019). arXiv:1907.11692 http://arxiv.org/ abs/1907.11692

[14] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever (2019) "Language Models are Unsupervised Multitask Learners". https://openai.com/blog/better-language- models/

[15] Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and EricFosler-Lussier (2017) "Cross-Lingual Transfer Learning for POS Tagging without Cross-Lingual Resources" Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Copenhagen, Denmark, 2832–2838. https://doi.org/10.18653/v1/ D17-1302

[16] Toan Q. Nguyen and David Chiang (2017) "Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation" Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers). Asian Federation of Natural Language Processing, Taipei, Taiwan, 296–301. https://aclanthology.org/I17-2050

[17] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu (2020) "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer" Journal of Machine Learning Research 21, 140 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[18] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight (2016) "Transfer Learning for Low-Resource Neural Machine Translation" Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Austin, Texas, 1568–1575. 18653/v1/D16- 1163

[19] Mikel Artetxe and Holger Schwenk (2019) "Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond" Trans. Assoc. Comput. Linguistics7(2019), pp 597–610. https://transacl.org/ojs/index.php/tacl/article/view/1742

[20] Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang (2015) "Multi- Task Learning for Multiple Language Translation" 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). Association for Computational Linguistics, Beijing, China, 1723–1732. https://doi.org/10.3115/v1/P15- 1166

[21] Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang (2020) "Language-agnostic BERT Sentence Embedding" CoRR abs/2007.01852 (2020). arXiv:2007.01852 https://arxiv.org/abs/2007.01852

[22] Graham Neubig and Junjie Hu (2018) "Rapid Adaptation of Neural Machine Translation to New Languages" Proceedings of the 2018 Conference on Em- pirical Methods in Natural Language Processing. Association for Computational Linguistics, Brussels, Belgium, 875–880. https:// doi.org/10.18653/v1/D18-1103

[23] Yinfei Yang and Amin Ahmad (2019) " Multilingual Universal Sentence Encoder for Semantic Retrieval". https://ai.googleblog.com/2019/07/multilingual-universal- sentence-encoder.html

[24] Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung (2021) "Are Multilingual Models Effective in Code - Switching?" CoRR abs/ 2103.13309 (2021). arXiv:2103.13309 https://arxiv.org/abs/ 2103.13309

[25] Sharanya Chakravarthy, Anjana Umapathy, and Alan W Black (2020) "Detecting Entailment in Code-Mixed Hindi-English Conversations" Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020). Association for Computational Linguistics,Online,165–170. https://doi.org/10.18653/v1/2020.wnut- 1.22

[26] Buddhika Kasthuriarachchy, Madhu Chetty, Gour Karmakar, and Darren Walls (2020) "Pre-trained Language Models with Limited Data for Intent Classification." International Joint Conference on Neural Networks(IJCNN) pp 1–9. https: //doi.org/10.1109/ IJCNN48605.2020.9207121

[27] Jitin Krishnan, Antonios Anastasopoulos, Hemant Purohit, and Huzefa Rangwala (2021) "Multilingual Code-Switching for Zero-Shot Cross-Lingual Intent Prediction and Slot Filling" CoRR abs/2103.07792 (2021). arXiv:2103.07792 https://arxiv.org/ abs/2103.07792

[28] Viswanathan S., Anand Kumar M., and Soman K.P. (2019) "A Comparative Analysis of Machine Comprehension Using Deep Learning Models in Code- Mixed Hindi Language" Recent Advances in Computational Intelligence. Studies in Computational Intelligence, Kumar R. and Wiil U (Eds.). Springer, Cham. https://doi.org/10.1007/978- 3- 030- 12500- 4_19

[29] Rajeshwari S.B. and Kallimani J.S. (2021) "Regional Language Code-Switching for Natural Language Understanding and Intelligent Digital Assistants" Innovations in Electrical and Electronic Engineering. Lecture Notes in Electrical Engineering, Mekhilef S., Favorskaya M., Pandey R.K., and Shaw R.N. (Eds.). Springer, Singa- pore. https://doi.org/ 10.1007/978-981-16-0749-3_71

[30] Wei Zhao, Steffen Eger, Johannes Bjerva, and Isabelle Augenstein (2020) "Inducing Language-Agnostic Multilingual Representations" CoRR abs/2008.09112 (2020). arXiv:2008.09112 https://arxiv.org/abs/2008.09112

[31] Arindrima Datta, Bhuvana Ramabhadran, Jesse Emond, Anjuli Kannan, and Brian Roark (2020) "Language-Agnostic Multilingual Modeling" ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 8239–8243. https://doi.org/10.1109/ICASSP40776.2020.9053443

[32] Tom Bocklisch, Joey Faulkner, Nick Pawlowski, and Alan Nichol (2017) " Rasa: Open Source Language Understanding and Dialogue Management" CoRR abs/1712.05181 (2017). arXiv:1712.05181 http://arxiv.org/abs/1712.05181

[33] Jacob Devlin (2019) "Google-Research/bert" https://github.com/google-research/ bert/blob/ master/multilingual.md.

[34] (2021) Rasa Open Source Documentation. https://rasa.com/docs/rasa/.

[35] Wei Wang, Bin Bi, Ming Yan, Chen Wu, Zuyi Bao, Liwei Peng, and Luo Si. (2019) "StructBERT: Incorporating Language Structures into Pre-training for Deep Language Understanding." CoRR abs/1908.04577 (2019). arXiv:1908.04577 http://arxiv.org/abs/ 1908.04577

[36] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew (2019) "HuggingFace's Transformers: State-of-the-art Natural Language Processing" CoRR abs/ 1910.03771 (2019). arXiv:1910.03771 http://arxiv.org/abs/ 1910.03771

[37] Mady Mantha (2020) "Introducing DIET: state-of-the-art architecture that outperforms fine-tuning BERT and is 6X faster to train", https://rasa.com/blog/introducing-dual-intent-and-entity-transformer-diet-state-of-the-art-performance-on-a-lightweight-architecture/

[38] M. Bounabi, K. E. Moutaouakil and K. Satori, "Neural Embedding & Hybrid ML Models for Text Classification," 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2020, pp. 1-6, doi: 10.1109/ IRASET48871.2020.9092230.

Authors

Karthika Vijayan is a Solution Consultant (Data Science) at Sahaj AI, focussing on voice & text based AI. Karthika holds her Bachelor's and Master's degrees in Electronics and Communication Engineering, and a PhD in Speech Processing from Indian Institute of Technology Hyderabad. Later she worked as a post doctoral research associate at Indian Institute of Science Bangalore (1 year) and as a research fellow at the National University of Singapore (4 years). Her research interests include speech processing and natural language processing for AI, pattern recognition, machine learning and deep learning.

Oshin Anand is a Solution Consultant (Data Science) at Sahaj AI. Oshin has a PhD in MIS (Text Analysis, NLP) from Indian Institute of Management Rohtak, India. She has around 9 years of experience working in NLP and econometric modelling. Previously, she worked in Indian public sector, government think tanks and corporates, and have been leading scientist teams in client engagements and research projects. Her research interest includes natural language processing, information extraction, and conversational AI.