Dialogue System with Named Entity Recognition using BERT

Manasi Swaminathan
Luddy School of Informatics, Computing, and Engineering
Indiana University, Bloomington, IN
mswamina@iu.edu

Abstract

Named Entity Recognition is the task in Natural Language Processing (NLP) that has its real world application growing in every field. The purpose of this project is to build a dialogue system that converses with the user agent by asking questions for collecting information on flight ticketing. Given a dialogue from the user agent Named Entity Recognition using BERT is performed. BERT is a language model that relies on a transformer which is bidirectionally trained which allows augments a greater sense of context in language compared to other models. An entity is a word or group of words that belong to a particular category like location, organization, etc. The next step after detecting an entity and categorizing the entity, is to fed it into the dialogue system to form a question for the dialogue agent. This is implemented with the inspiration of hybrid architecture and frame based architecture. The necessary responses from the user is stored in frames or slots. These slots or frames are used to perform a certain goal or task like finding available flights based on the user's input. In this project a part of the dialogue system is implemented. Once the user response is fed into the NER model which acts as Natural Language Understanding component of the dialogue system architecture. Some rules are created based on the output of the NER model to store the information from the user. The agent dialogues is prompted by rules based on the design of the conversation flow. The BERT model for Named Entity Recognition is trained on two different optimizers Adam and Stochastic Gradient Descent (SGD), and the accuracies are compared. The BERT using SGD performs the best with an accuracy of 97.8% and the BERT using Adam results in accuracy of 86.3% .

Keywords: Named Entity Recognition (NER), Natural Language Processing (NLP), BERT, Dialogue System

1 Introduction

In the field of Natural Language Processing and Computational Linguistics, Named Entity Recognition (NER) and dialogue systems continue to be a topic of research. The applications of these tasks are widespread in real world technology. In this project, the two main components are Named Entity Recognition and Dialogue System. These two topics of research is very challenging because it needs to be as accurate and efficient with multiple languages and it competes with human beings. In this project we are only going to deal with English data.

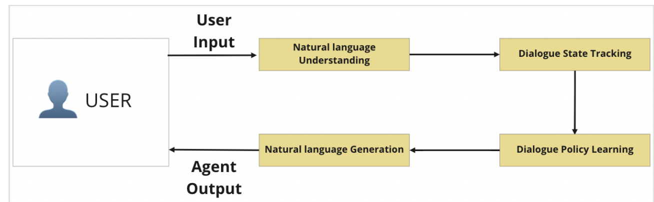

Figure 1: Goal Based Dialogue System

1.1 Named Entity Recognition (NER)

Named Entity Recognition is a methodology for Information Extraction for text data in the field of Natural Language Processing.(Liu, 2021) Named Entity Recognition can be considered as a classification problem. The goal of NER is to extract information from a text and classify it in the category that it belongs for example name, location, organization etc. NER is used in various application such as trend analysis, recommendation system from social media. It can also be used in gaining insights in feedback or summarizing a large text document. It is also used in processing resumes as well so that the most relevant information like candidate name, address, skills etc are extracted.

1.2 Dialogue System

Dialogue systems or conversational agents are designed to conduct a conversation with users. It can be roughly classified into task based dialogue system that is designed to do a particular task or non task based or chatbots designed to have random (Martin.,) conversations. An example of task based dialogue system can be Alexa, Google Assistant, Siri, etc which performs tasks like setting alarm, making calls or even finding nearby restaurants to eat in. Chatbots can be rule based like the history setting ELIZA and PARRY chatbot. Corpus based dialogue system rely heavily on data and it is trained on machine learning models. The data for such dialogue systems contains patterns and intents that are matched. Hybrid based dialogue system architecture is designed with rules as well as corpus based. For example GPT-2 and GPT-3 developed by Open AI works on this concept.

The state of the art current development of dialogue system especially for goal oriented tasks is the sophisticated use of frame based architecture. (Chen et al., 2017) The structure of this architecture contains mainly 4 components. Natural Language Understanding (NLU), Dialogue State Tracker, Dialogue Policy learning and Natural Language Generation (NLG). NLU is the component that breaks the user dialogue into entities and feeds it into predefined slots. The Dialogue state tracker controls the input for every turn handles the history of the dialogue, and outputs the current state. The Dialogue policy learning learns the next action based on the current dialogue and the NLG outputs a response. 1

The rest of the paper is organized as follows. In Section 2, we provide a literature survey related to various models used for NER and on various dialogue systems. In Section 3, details of the dataset along with label description is discussed followed by Section 4, where we present the BERT model, the conversation flow and the rule for dialogue system. 5 where the accuracy of the model is discussed. Finally, Section 6 concludes the study with future implementation.

2 Related Works

2.1 Named Entity Recognition

The first concept of Named Entity Recognition originated from the message understanding conference in 1996 (Grishman and Sundheim, 1996). The conference talked about the necessity to make information extraction system more portable. It demonstrated task-independent information extraction system using templates. The first ever NER dataset was created from this conference. Since then many datasets like CoNLL 2002, 2003 were created from newswire in various languages(Yadav and Bethard, 2018). There has been datasets for various domains like Biomedical, Archaeology etc.

The knowledge based system for NER is used for domain specific datasets. This time of system does not need annotated training data and the system depends on lexicon resources and domain specific knowledge (Yadav and Bethard, 2018). During the early days the NER systems had to make use of the limited training data. (Etzioni et al., 2005) The researchers of this paper proposed an unsupervised learning technique by using 8 generic pattern extractors from the web. The pattern learning extracted domain specific extraction rules that boosted the 4-fold to 8-fold increase in recall at precision of 0.90. In the paper (Zhang and Elhadad, 2013) the researchers suggested a unsupervised learning technique that does not rely on annotated data and handcrafted rules instead used syntactic knowledge and inverse document frequency (IDF) on biomedical data that reached an accuracy of 69.5%.

In the later years with the increase in data supervised learning that is trained of example inputs and their expected labels began. Algorithms like Support Vector Machine (SVM), decision tree and other classifier algorithms were used. (Zhou and Su, 2002) In this paper Hidden Markov Model (HMM) was proposed on MUC-6 and MUC-7 data, which resulted in 96.6% and 94.1% F score, respectively. Features like (1 number, 2 numeral, 4 numeral, all capitals, numerals and alphabets, contains underscore or not, etc.) were incorporated in this. A list of words (10000 for the person entity class) from

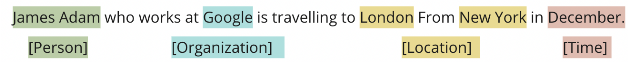

Figure 2: Named Entity Recognition Example

multiple gazetteers, as well as a list of trigger words for the named entities (for example, 36 trigger words and affixes, like river, for the place entity class).(Li,) In this paper SVM model was implemented on English CoNLL 2003 data. In order to balance positive and negative class, they experimented with a variety of window sizes, features (orthographic, prefixes, suffixes, labels, etc.) from nearby words, weighting neighboring word features according to their position, and class weights. One SVM classifier was used to identify named entity starts, and the other was used to identify named entity endings. They received a F score of 88.3%.

Later discovered that NER benefits from non-linear mapping from input to output therefore deep learning techniques that uses non linear activation functions will make NER systems more efficient. (Li et al., 2022). The commonly used models are RNN, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). (Chiu and Nichols, 2015) In this paper the author suggested a novel neural network architecture that detects word and its character level feature. The model is a hybrid Bidirectional LSTM and CNN model. This resulted in a F1 score of 91.62% on the CoNLL-2003 dataset.

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) was introduced in this paper by Google. In the paper (Liu, 2021) BERT was implemented on several datasets of various domains with some fine-tuning. An average 76.49%. accuracy was achieved. Since the introduction of BERT as a multilingual model the efficiency of NER in other languages has increased. (Litake et al., 2022) In this paper the authors proposed systems based on variations of BERT like RoBERTa, AlBERT and base-BERT and performed a comparison study on cross languages. 88.90% F1 score was achieved on Marathi dataset and 83.04% on Hindi dataset.

2.2 Dialogue System

Dialogue systems can be roughly categorized as (1) task oriented and (2) non-task oriented like a chat bot. Task oriented dialogue system are designed to perform a task to achieve a goal. The system can be simple rule based or trained with machine learning and deep learning models. Rule based dialogue system started from this paper (Weizenbaum, 1966) which proposed ELIZA. This is the most historical dialogue system in the field. It worked on pattern transformed rules.

Corpus based dialogue system requires enormous amount of data for training. This is used for question and answering systems. Using the methodology suggested in this paper (Humeau et al., 2020) BERT model is also used for longer context dialogue system. (Wen et al., 2016) This paper the authors have proposed a neural network based text-in, text-out end to end goal oriented dialogue system with collection of data using Wizard-of-Oz.

The groundbreaking (Radford et al., 2019) GPT-2 model

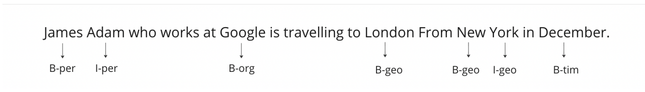

Figure 3: Example of labels in Dataset

from OpenAI uses 1.5 Billion parameters to achieve state of the art dialogue system. Now GPT-3 exists with various other model releases inside with more fine tuned parameters that continues to surpass all dialogue systems.

3 Dataset

The dataset for the Names Entity Recognition comes from two sources. The two datasets has been combined to get 140677 data entries to increase the number of entity types. It contains two columns the text and the labels. The labels have the entity type for each and every word of the text or sentence. The first dataset is the CoNLL-2003 dataset (Liu, 2021) is from Kaggle and has 47958 data entries. This dataset contains 9 entity categories: geographical entity, organization entity, person entity, geopolitical entity, time indicator entity, artifact entity, event entity, natural and phenomenon entity. The second dataset (Tedeschi et al., 2021) contains 92718 data entries. The dataset contains 4 entity categories person entity, organization entity, geographical entity and miscellaneous entity. Both combined has the following label shown in Table 1.

Label	Definition
B-per	Beginning of person entity
I-per	Continuation of person entity
B-org	Beginning of organization entity
I-org	Continuation of organization entity
B-geo	Beginning of a geographic entity
I-geo	Continuation of geographic entity
B-gpe	Beginning of a geopolitical entity
I-gpe	Continuation of a geopolitical entity
B-tim	Beginning of a time indicator entity
I-tim	Continuation of a time indicator entity
B-eve	Beginning of a event entity
I-eve	Continuation of a event entity
B-art	Beginning of a artifact entity
I-art	Continuation of a artifact entity
B-eve	Beginning of a natural and phenomenon entity
I-eve	Continuation of a natural and phenomenon entity
B-misc	Beginning of a miscellaneous entity
I-misc	Continuation of a miscellaneous entity
O	If a word does not belong to any entity

Table 1: Dataset Labels

4 Methodology

4.1 Training Named Entity Recognition System

Bidirectional Encoder Representations from Transformers (BERT) is a language model that is Bidirectional. As opposed to the single-direction language models, we can now perceive the context and flow of language more deeply. The input for the BERT model is a sequence of words or tokens. BERT looks for two specific tokens CLS and SEP for an input. CLS is the first token in a sequence which is the classification token and SEP is the token that lets BERT identify which token belongs to which sequence. For sentence prediction or question and answering tasks this token plays a role. BERT outputs an embedding vector. This embedding vector can be used for various NLP tasks. To use BERT for NER, we need the embedding vector output for all the tokens. The first step of implementation is Data pre-processing where first we need perform tokenization. Tokenization can be done with BERT using BertTokenizerFast. Here we are using the class from the pretrained BERT base model with HuggingFace. The maximum length of a sequence in BERT is 512 therefore, we need to use a special PAD token to perform padding to fill the unused token slots if the sequence is less that 512. If the sequence is higher than 512 then it needs to truncated. The tokenization outputs a dictionary that contains three components. First is the id that represents the token in the sequence. There are 3 different ids. The second component is the token type id which identifies which sequence the token belongs to. In this problem we have only one sequence per text therefore, it is 0. The third component is the attention mask that identifies which token is a real token and which is a padded token.

The second step is to adjust the labels after tokenization. The BERT tokenizer used sub word tokenizer that even splits certain words. This is done so that the model learns semantic meaning. But this leads to additional tokens therefore they need to be matched with the label. Every split word has the same the word-id. So this methodology to adjust the labels provides a label to the first split token and -100 as label for the sub token. Next the dataset is split for training, testing and validation. To build the model in this implementation we are using the pre-trained model from HuggingFace. We are using BertForTokenClassification class as we are classifying text in token level. This class adds linear layers on top of BERT level which therefore, acts like a token level classifier. We are performing a comparitive study of two different optimizers for training Adam and Stochastic Gradient Descent (SGD). The model is trained for 3 epochs with a batch size 2. After training the -100 labels are ignored.

The evaluation of the model is done using accuracy. Then the next step is to make use of the trained model to predict the entity of every word in a sentence.

4.2 Hybrid Based Dialogue System

Creating a dialogue system that can ask the user ticketing information and store them for searching tickets. This dialogue system uses the trained NER model as the Natural Language Understanding component and the results are stored in slots or frame which is created using a dictionary which acts as the Dialogue State Tracker.

First a conversation flow was created Figure 4. Using the conversation flow the rules were made in order to ask the particular question and receive the response to store the data that is required from the user.

After each response the sentence is fed into the NER model. The NER model's output is the entity that is predicted for each

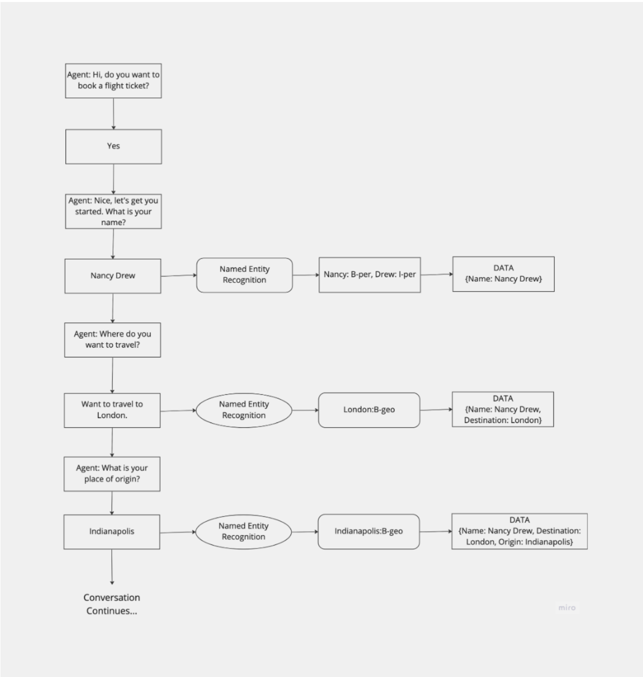

Figure 4: Sample Conversation Flow

```
  warnings.warn(
100%|██████████| 56271/56271 [59:38<00:00, 15.73it/s]
Epochs: 1 | Loss:  0.699 | Accuracy:  0.864 | Val_Loss:  0.697 | Accuracy:  0.863
100%|██████████| 56271/56271 [59:47<00:00, 15.69it/s]
Epochs: 2 | Loss:  0.696 | Accuracy:  0.864 | Val_Loss:  0.703 | Accuracy:  0.863
100%|██████████| 56271/56271 [59:53<00:00, 15.66it/s]
Epochs: 3 | Loss:  0.696 | Accuracy:  0.864 | Val_Loss:  0.701 | Accuracy:  0.863

    val_accuracy tensor(0.8631, device='cuda:0')
    Test Accuracy:  0.863
```

Figure 5: Training of BERT on Adam Optimizer

word in the sentence. Based on the conversation flow the entity predicted is fed into the rules for the next question from the agent and the slot or frame is updated.

5 Results and Discussions

The Named Entity Recognition model is evaluated based on accuracy. The accuracies of the NER models trained on Adam Optimizer and Stochastic Gradient Descent (SGD) is compared as seen in Table 2. It is seen that SGD optimizer has a better accuracy than Adam. The output of the conversation flow looks like Figure 7. The data that is made of the slot or frame looks like Figure 8.

```
You should probably TRAIN this model on a down-stream task to be able to use it for
100%|██████████| 56271/56271 [53:12<00:00, 17.63it/s]
Epochs: 1 | Loss:  0.147 | Accuracy:  0.957 | Val_Loss:  0.093 | Accuracy:  0.972
100%|██████████| 56271/56271 [53:08<00:00, 17.65it/s]
Epochs: 2 | Loss:  0.087 | Accuracy:  0.973 | Val_Loss:  0.078 | Accuracy:  0.977
100%|██████████| 56271/56271 [53:15<00:00, 17.61it/s]
Epochs: 3 | Loss:  0.072 | Accuracy:  0.978 | Val_Loss:  0.075 | Accuracy:  0.978

  val_accuracy tensor(0.9779, device='cuda:0')
  Test Accuracy:  0.978
```

Figure 6: Training of BERT on SGD Optimizer

Optimizer	Accuracy
Stochastic Gradient Descent	**97.8%**
Adam	86.3%

Table 2: Comparison of Adam and SGD Optimizers on BERT

```
Agent: Hi, do you want to book a flight ticket? [Reply Yes or No]
yes
Agent: Nice, lets get you started, What is your name?
My name is Nancy Drew
Agent: Where do you want to travel?
I want to travel to London.
Agent: What is your place of origin?
I am from Indianapolis.
Agent: What day do you want to travel? [example: 12th December 2022]
12th June 2023
Agent: What time of the day do you want to travel? [example: morning, afternoon, evening]
morning
Agent: Do you want a return ticket?[Reply Yes or No]
Yes
Agent: When do you want to return?[example: 12th December 2022]
23rd June 2023
Agent: What time of the day do you want your return flight? [example: morning, afternoon, evening]
afternoon
Thank you
```

Figure 7: Dialogue System Output

6 Conclusions

Named Entity Recognition (NER) models can be used for dialogue systems to perform a particular task or achieve a particular goal. In this project we looked at how NER can be used to collect information for flight ticketing from the user. The Dialogue system can be further implemented with Dialogue Policy Learning which includes the Dialogue manager, and Natural Language Generator that completes the goal of ticketing the user by recommending available flights and its price.

The NER model can be further improved by customizing the data based on the task or goal of the dialogue system. There needs to be balance in data. This dataset has a problem in data imbalance is there is more labels with 'O' than any other entity. This affects the accuracy of the model. The BERT model can be further fine tuned with the weights in the optimizer.

References

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *ArXiv*, abs/1711.01731.

Jason P. C. Chiu and Eric Nichols. 2015. Named en-


```
[{'Name': 'Nancy Drew',
'Origin': 'Indianapolis.',
'Destination': 'London.',
'Date of Travel': '12th June 2023',
'Time of Travel': 'morning',
'Return Date': '23rd June 2023',
'Return Time': 'afternoon'}]
```

Figure 8: Frame Data Output

tity recognition with bidirectional lstm-cnns. *CoRR*, abs/1511.08308.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. pages 4171–4186, June.

Oren Etzioni, Michael J. Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artif. Intell.*, 165:91–134.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Onkar Litake, Maithili Sabane, Parth Patil, Aparna Ranade, and Raviraj Joshi. 2022. Mono vs multilingual bert: A case study in hindi and marathi named entity recognition. *ArXiv*, abs/2203.12907.

Jiang F. Hu Y. Shi C. amp; Fung P. Liu, Z. 2021. December.

Daniel Jurafsky James H. Martin. *Speech and Language Processing*. Stanford.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners.

Simone Tedeschi, Valentino Maiorca, Niccolò Campolungo, Francesco Cecconi, and Roberto Navigli. 2021. Nov.

Joseph Weizenbaum. 1966. Eliza—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45, jan.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, David Vandyke, and Steve J. Young. 2016. A network-based end-to-end trainable task-oriented dialogue system. *CoRR*, abs/1604.04562.

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. pages 2145–2158, August.

Shaodian Zhang and Noémie Elhadad. 2013. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of biomedical informatics*, 46 6:1088–98.

GuoDong Zhou and Jian Su. 2002. Named entity recognition using an HMM-based chunk tagger. pages 473–480, July.