Named Entity Recognition on the CoNLL2002 Spanish Subset

Dipunj Gupta (PittID: DIG44)

March 29, 2023

Abstract

Named entity recognition (NER) is an important task in natural language processing that involves identifying and classifying entities in text, such as people, organizations, and locations. In this report, we present our approach to NER on the Spanish subset of the CoNLL2002 dataset, with the goal of achieving a high F1-score. We experimented with several models and features, including a basic perceptron model with window-based features, a multi-layer perceptron model, and a BERT model. Our best performing model was the BERT model, which achieved an F1-score of 86.90% after fine-tuning it on the CoNLL2002 dataset. We also analyzed the errors made by our models and identified some common patterns, such as confusion between similar named entity types and difficulty in handling nested entities. Overall, our results demonstrate the effectiveness of BERT for NER in Spanish, and highlight the importance of careful feature selection and error analysis in developing accurate NER systems.

Introduction

Named Entity Recognition (NER) is a critical Natural Language Processing (NLP) task that involves identifying and categorizing named entities in text. In this report, we focus on the Spanish subset of the CoNLL2002 dataset and present our experiments to improve the performance of a baseline model on this task. We explore different features and models, including a BERT-based model, and provide a detailed analysis of their impact on the model's performance. Finally, we perform error analysis to identify the strengths and weaknesses of our approach. Our results demonstrate that BERT models are very effective for NER in Spanish, and they perform close to the state of art NER models on this dataset. We used esp.testa subset as test dataset, and esp.train and esp.testb as training and validation datasets respectively.

Baseline

The baseline model used in this experiment is a perceptron model that employs a window of size 3 (i.e., the previous word, current word, and the next word) as the features. The baseline model achieved an F1-score of 49%.

Experiments

We did several experiments to improve the baseline model's performance. We experimented with different features and models, including a multi-layer perceptron model and a BERT model. We also experimented with different window sizes and feature combinations. We present the results of our experiments in the following sub-sections.

Experiment 1

We used the same baseline model and added POS tags to the set of features for each word in the window. We thought that the POS tags could provide additional information about the context in which the named entities occur. However, the addition of POS tags degraded the model's F1-score. On further examining the spanish dataset, we found out that there are terms like "La Coruña", where "La" is actually part of a named entity, unlike most other occurrences of "la" (meaning "the"), we concluded that adding POS tags alone introduced noise to our model instead of further improving it.

Class	Precision	Recall	F1-score	Support
LOC	55.69%	68.60%	61.48	1212
MISC	17.15%	29.44%	21.67	764
ORG	43.50%	48.82%	46.01	1908
PER	60.56%	50.90%	55.31	1027
Accuracy: 91.46%, Precision: 45.98%, Recall: 51.90%, FB1: 48.76				

Table 1: Perceptron model with text and POS tag features and window size 3.

Experiment 2

Even though the POS tags did not improve the model's performance, we thought that the POS tags could be useful in identifying named entities of certain types. Therefore we thought to increase the window size from 3 to 5 while keeping the POS tags as a feature. The overall F1-score improved by 4% as a consequence of increasing the window size, while keeping the POS tags as a feature.

Class	Precision	Recall	F1-score	Support
LOC	54.37%	70.12%	61.25	1269
MISC	23.50%	28.09%	25.59	532
ORG	54.26%	53.94%	54.10	1690
PER	68.67%	46.64%	55.56	830
Accuracy: 92.58%, Precision: 53.27%, Recall: 52.91%, FB1: 53.09				

Table 2: Perceptron model with text, POS tag features and window size of 5.

Experiment 3

We used the same model as in the experiment 2 and added another feature called *is_uppercase* – which indicates if the first letter of the word is capitalized. We thought that this additional information could be correlated to named entities of LOC and ORG entities, as such named entities are usually capitalized. The overall F1 score improved by 12% as a consequence of adding this feature.

Class	Precision	Recall	F1-score	Support
LOC	59.97%	73.37%	66.00	1204
MISC	25.35%	40.90%	31.30	718
ORG	55.89%	61.12%	58.39	1859
PER	67.39%	73.24%	70.20	1328
Accuracy: 94.60%, Precision: 55.55%, Recall: 65.23%, FB1: 60.00				

Table 3: Perceptron model with text, POS tag, and uppercase features and window size 5.

Experiment 4

To the feature set of experiment 3, we added another feature called *is_punctuation* – which indicates if either of the special characters ;?!:;()'"[]{} are present in a word, while keeping the window size of 5. We thought that the addition of is_punctuation feature could help the model to identify named entities of PER and MISC categories, as such named entities are usually punctuated (eg uses cases like "Juan's" contains an apostrophe). The overall F1 score improved by 1% as a consequence of adding this feature.

Class	Precision	Recall	F1-score	Support
LOC	56.70%	78.25%	65.76	1358
MISC	25.59%	41.35%	31.62	719
ORG	57.71%	61.88%	59.72	1823
PER	73.40%	72.26%	72.82	1203
Accuracy: 94.87%, Precision: 56.61%, Recall: 66.40%, FB1: 61.12				

Table 4: Perceptron model with text, POS tag, is_uppercase, is_punctuation features and a window size of 5.

Experiment 5

After achieving a 60+% F1 score on perceptron model, we thought to use the same features on a model that could capture non-linear boundaries. We used a multilayer perceptron model with 2 hidden layers of 100 and 20 neurons respectively. We used the same feature set as in experiment 3. The overall F1 score improved by 3% as a consequence of using a multilayer perceptron model.

Class	Precision	Recall	F1-score	Support
LOC	60.65%	77.54%	68.06	1258
MISC	32.34%	41.57%	36.38	572
ORG	58.72%	69.71%	63.74	2018
PER	79.24%	68.41%	73.43	1055
Accuracy: 95.61%, Precision: 60.55%, Recall: 68.24%, FB1: 64.17				

Table 5: Multi Layer Perceptron model 2 hidden layers of sizes (100, 20). The input features were text, POS tag, is_uppercase, is_punctuation features.

Experiment 6

After experimenting with these simple models, we felt that these models are not able to incorporate the semantic meaning of the words and any other feature engineering was not going to yield massive improvements. Therefore, we thought to instead use a BERT model to extract the contextual embeddings of the words. We used the BERT model with the *dccuchile/bert-base-spanish-wwm-cased* pre-trained weights and fine tuned it on the CONLL-2002 dataset using SpaCy. The overall F1 score improved to 86.9% as a consequence of using the BERT model. This is a significant improvement over the previous models, and we believe that this is because the BERT model is able to capture the semantic meaning of the words that it learned from the pre-training phase.

Class	Precision	Recall	F1-score	Support
LOC	86.83%	87.80%	87.32	995
MISC	70.84%	62.25%	66.27	391
ORG	84.49%	87.18%	85.81	1754
PER	96.40%	94.11%	95.24	1193
Accuracy: 97.83%, Precision: 87.08%, Recall: 86.72%, FB1: 86.90				

Table 6: BERT model with *dccuchile/bert-base-spanish-wwm-cased* pre-trained weights.

Error Analysis

To analyze the errors made by the best performing model (the BERT fine-tuned model), we examined the cases where the model made mistakes in predicting the named entity categories. We found the following misclassification statistics:

Gold/Actual	Maximum Predicted	Number of times
LOC	ORG	87
MISC	ORG	135
ORG	MISC	93
PER	ORG	45
O	MISC	147

Table 7: Misclassification data: what was the maximum predicted category and how many times did it occur against each NER tag.

It looks like that the model mostly misclassifies MISC to ORG and vice versa, as they form the bulk of the misclassifications. We also looked at individual sentences where errors occurred. Here are some examples:

- "Florida Ileana" was predicted as B-LOC B-PER, where as the gold label is B-PER I-PER. This is because the model was not able to identify the context of the sentence and assumed from it's pre-training knowledge that the word "Florida" is more likely a location and not a person.
- "la Comunidad Valenciana" is a location but it was predicted as an organization.
- Acronyms such as "GFBITAL", "MASECA", "IASASA" were misclassified as B-ORG, where as the gold label is B-MISC. Our model most likely learned from pretraining data that Acronyms are more likely to be organizations.
- "el estadio Bentegodi" was predicted as O B-LOC I-LOC, where as the gold label is O O B-LOC. It seems like a mistake in the dataset, because "estadio" which means "stadium" should be a LOC entity and not a non-entity.
- Our model confused names, which is an understandable error, because names like "Fresi" which are labelled as B-PER could actually be B-ORG (which our model predicted) if a place or organization is named after a person.

Conclusion

In this assignment, we experimented with several models for named entity recognition using the CoNLL-2002 Spanish dataset. Our best performing model was a BERT-based model fine-tuned on the dataset, which achieved an F1 score of 86.9%.

Through our error analysis, we found that the model made the most mistakes in distinguishing between MISC and ORG categories. This suggests that the dataset could be improved by providing clearer distinctions between these categories. Additionally, we identified some cases where the model's errors were due to dataset issues or ambiguity in interpretation.

Overall, our experiments demonstrated that using a pre-trained language model like BERT can significantly improve the performance of named entity recognition systems over conventional machine learning models such as Perceptron or Multi layer perceptron, while also reducing the effort needed for manual feature engineering but at the cost of a larger model size and increased training time.