CLEF2023' JOKER

Notebook for the JOKER Lab at CLEF 2023

Julia Komorowska[1], Iva Čatipović[2] and Darko Vujica[3]

[1]Univeristy of Gdansk,Jana Bażyńskiego 8, 80-309 Gdańsk
[2]Univeristy of Split, Ruđera Boškovića 31, 21000, Split, Chorwacja
[3]Univeristy of Split, Ruđera Boškovića 31, 21000, Split, Chorwacja

Abstract
These working notes explore the application of OpenAI's GPT-3 in tasks related to puns, including pun detection, translation, interpretation, and location. GPT-3 is a powerful language model that can be leveraged to assist in various language-related tasks. However, it is important to note that its capabilities in specific areas, such as puns, may have certain limitations.

Keywords
Pun detection, pun translation, paper formatting, Joker, Naive Bayes, Random Forest,

1. Introduction

The problem of pun detection, location, translation, and interpretation involves the understanding and analysis of puns in language. A pun is a form of wordplay that exploits different possible meanings of a word or words, or the similarity of sounds between different words, to create a humorous or witty effect. It often involves the clever use of double entendre, homophones, or similar-sounding words to create a play on words. Puns are typically used to create a humorous or lighthearted effect in writing, jokes, riddles, or verbal conversation. They rely on the listener or reader recognizing the multiple meanings or sounds involved and appreciating the cleverness or wordplay involved in the pun.

Pun detection involves identifying whether a given statement contains a pun or not, while pun location involves determining the specific word or phrase in the statement that is being used in the pun.

Pun translation involves finding equivalent puns in different languages, which can be challenging due to differences in word structure, sound patterns, and cultural references. Finally, pun interpretation involves understanding the intended meaning of a pun, which can be difficult because puns often rely on context and can have multiple interpretations.

Pun detection, location, translation, and interpretation are important tasks in natural

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

language processing, as they can improve the accuracy of machine translation, sentiment analysis, and other language-based applications.

2. Approach

GPT-3 is a powerful language model that can be used for a variety of natural language processing tasks, including pun detection, location, translation, and interpretation. Here are some approaches to solving these tasks using GPT-3:

Pun detection: GPT-3 can be used to detect puns by analyzing the language patterns and the context of the text. To do this, you can feed the text containing the pun to GPT-3 and then analyze the output to see if it recognizes the pun. This approach can be enhanced by fine-tuning GPT-3 on a pun detection dataset to improve its accuracy.
GPT can be trained to recognize patterns and wordplay commonly associated with puns. By providing examples of puns and non-puns during the training process, you can potentially develop a model that can detect puns in text. However, building a robust pun detection model can be challenging, as puns often rely on contextual understanding and cultural references.

Pun location: GPT-3 can also be used to locate puns by identifying the specific word or phrase that is being used in the pun. This can be achieved by training GPT-3 on a dataset of puns and non-puns, then using it to analyze new text samples to locate puns. This approach can be further improved by incorporating additional features, such as part-of-speech tags or syntactic structures.
GPT can help with pun-related queries by providing information about puns associated with specific locations, such as puns related to place names or local references. By specifying the location or context, you can ask GPT for puns related to that particular place. However, the availability and accuracy of puns related to specific locations may vary.

Pun translation: GPT-3 can be used to translate puns between languages by leveraging its multilingual capabilities. You can input the pun in one language, and then use GPT-3 to translate it to another language while preserving the pun's intended meaning. This approach can be further improved by fine-tuning GPT-3 on a pun translation dataset to enhance its translation accuracy.
GPT can assist in translating puns between languages to some extent. By providing the original pun in one language, you can ask GPT to generate an equivalent pun in another language. However, due to the complexity and nuance of puns, translation may not always preserve the humor or intended meaning.

Pun interpretation: GPT-3 can also be used to interpret puns by analyzing the context and the meaning of the text. To do this, you can input the pun to GPT-3 and then use its language understanding capabilities to determine the intended meaning of the pun. This approach can be enhanced by fine-tuning GPT-3 on a dataset of puns and their interpretations to improve its accuracy.

GPT can assist in interpreting puns by providing explanations or alternative interpretations. By providing a pun and asking for its meaning or different ways to understand it, GPT can generate explanations based on its training data. However, due to the subjective nature of humor and the complexity of puns, interpretation may vary, and not all puns will have a definitive explanation.

Overall, GPT-3 can be used to solve pun detection, location, translation, and interpretation tasks by leveraging its natural language processing capabilities and by fine-tuning it on pun-specific datasets to enhance its accuracy.

Classifiers are also very common for pun detection. **Naive Bayes[3]** is a probabilistic algorithm that can be used for classification tasks, such as identifying whether a given statement contains a pun or not. It works by calculating the probability of each word in the statement being associated with either pun or non-pun categories, and then making a classification decision based on these probabilities.

Random Forest[6] is a decision tree-based algorithm that can be used for both classification and regression tasks. In pun detection and location, it can be used to create a set of decision rules based on the features of the text, such as the presence of homophones or multiple meanings of words.

Neural networks[4] are a powerful class of machine learning algorithms that can be used for a variety of tasks, including classification, regression, and language modeling. In pun detection, location, translation, and interpretation, neural networks can be used to process and analyze the text, learn patterns, and make predictions based on these patterns.

K-Nearest Neighbors (KNN)[5] is a non-parametric algorithm that can be used for classification tasks. It works by finding the k closest data points in a training set to a new data point and using the labels of those neighbors to classify the new data point. In pun detection and location, KNN can be used to find similar text samples that contain puns and use their labels to classify new text samples.

Overall, these algorithms can be used in combination with various features of text, such as lexical, syntactic, and semantic features, to build models that accurately detect, locate, translate, and interpret puns in language. The choice of algorithm and feature set will depend on the specific requirements and constraints of the task at hand.

3. Results

3.1. GPT3

GPT-3 indicated perfectly all the pun locations, translations, interpretentions as can be seen on photos.

location	text
drift	When they bought a water bed, the couple started to drift apart.
lose	OLD BREAD MEN never die, they just lose their dough.
melons	She was only a Fruit vendor's daughter, but, my, she had big melons.
felt	He crashed through several windows, but felt no pane.
a-maize-ing	Corn is so versatile that it is an a-maize-ing grain.
dawn	Those who get up at sunrise have many ideas dawn on them.

It was done just by changing a prompt. For example this is prompt for pun interpretation. GPT-3 can show where probably is location of a pun. Here's an example to illustrate how GPT-3 can find location:

User: "Why did the sand go to therapy? It had some 'shore' issues!"

In this example, GPT-3 recognizes the keyword "beach" and generates a pun involving the word "shore." While the generated pun is related to the beach, GPT-3 does not possess knowledge of the exact location where the pun might be relevant.

GPT-3 relies on its language understanding and knowledge acquired from training on a vast amount of text data to find possible locations of the puns.

```
    def simpleMyPrompt(prompt, input):

response = openai.Completion.create(
model="text-davinci-003",
prompt=prompt,
temperature=0.7,
max_tokens=256,
top_p=1,
frequency_penalty=0,
presence_penalty=0
)
return response
data_test.append(data_train['text'].apply(lambda x:
    simpleMyPrompt('Provide me with interpretetion of this
    joke',x)))
```

3.2. Classificators

They checked if train data, which was provided could "guess" if test data correctly indicated pun localization. Results were not exacly the best. Score of every classifier was around 0.0 to 0.4. The "score" in the context of a classifier typically refers to a performance metric that quantifies the accuracy or effectiveness of the classifier's predictions. The specific score or metric used depends on the problem type (e.g., binary classification, multi-class classification) and the evaluation goals. Classifiers can sometimes have lower accuracy or "bad scores" for several reasons:

- classifiers require a sufficient amount of training data to estimate the probabilities accurately. If the dataset is small or lacks representative samples, the classifier may not be able to generalize well to unseen data, leading to lower accuracy.
- If the dataset is imbalanced, meaning that one class has significantly more instances than others, classifiers can struggle to handle the imbalance effectively. They may exhibit a bias towards the majority class and perform poorly on the minority class.
- If some features in the dataset are irrelevant or provide limited discriminatory power for the target variable, naive Bayes classifiers may struggle to separate different classes effectively, resulting in lower accuracy.

In this case we provided smaller amount of training data.

4. Conclusions

In conclusion, pun detection, location, translation, and interpretation are important natural language processing tasks that involve understanding and analyzing the use of puns in language. These tasks can be addressed using various machine learning algorithms and techniques, including GPT-3, naive Bayes, random forest, neural networks, and KNN.

GPT-3 is a particularly powerful language model that can be used to solve these tasks due to its natural language understanding capabilities and its ability to process large amounts of text data. By training GPT-3 on pun-specific datasets and leveraging its multilingual capabilities, we can accurately detect, locate, translate, and interpret puns in different languages and contexts.

GPT has the potential to provide better scores in pun detection, pun translation, pun location, and pun interpretation than classifiers due to its extensive training on a diverse range of text data. The model's broad language understanding and ability to capture contextual nuances contribute to its performance in these tasks. However, the specific performance and score improvements can vary depending on the quality of training data, task complexity, and the availability of specialized models or approaches designed for these specific pun-related tasks.

Overall, the ability to accurately detect, locate, translate, and interpret puns can have important implications for various natural language processing applications, such as machine translation, sentiment analysis, and text generation. As such, continued research and development in this area are essential for improving the accuracy and reliability of these language-based applications.

References

[1] Ardi, H., Al Hafizh, M., Rezqi, I., Tuzzikriah, R.: Can machine translations translate humorous texts? Humanus 21(1) (2022). h

[2] Arroubat, H.: CLEF Workshop: Automatic Pun and Humour Translation Task. In: Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum,

Bologna, Italy, September 5th to 8th, 2022. CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy (2022) tps://doi.org/10.24036/humanus.v21i1.115698

[3] https://towardsdatascience.com/naive-bayes-classifier-how-to-successfully-use-it-in-python-ecf76a995069 - Naive Bayes

[4] https://scikit-learn.org/stable/modules/neural_networks_supervised.html- Neural Networks

[5] https://www.kaggle.com/code/prashant111/knn-classifier-tutorial -KNN

[6] https://www.ibm.com/topics/random-forest - Random Forest

[7] Castro, S., Chiruzzo, L., Rosá, A.: Overview of the HAHA task: Humor analysis based on human annotation at IberEval 2018. In: Rosso, P., Gonzalo, J., Martínez, R., Montalvo, S., de Albornoz, J.C. (eds.) Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages. CEUR Workshop Proceedings, vol. 2150, pp. 187–194. Spanish Society for Natural Language Processing (Sep 2018), http://ceur-ws.org/Vol-2150/overview-HAHA.pdf

[8] Ermakova, L., Miller, T., Regattin, F., Bosser, A.G., Mathurin, E., Corre, G.L., Araújo, S., Boccou, J., Digue, A., Damoy, A., Jeanjean, B.: Overview of JOKER@CLEF 2022: Automatic Wordplay and Humour Translation workshop. In: Barrón-Cedeño, A., Da San Martino, G., Degli Esposti, M., Sebastiani, F., Macdonald, C., Pasi, G., Hanbury, A., Potthast, M., Faggioli, G., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022). LNCS, vol. 13390 (2022)

[9] Zhao, Z., Cattle, A., Papalexakis, E., Ma, X.: Embedding lexical features via tensor decomposition for small sample humor recognition. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP–IJCNLP). pp. 6376–6381 (Nov 2019). https://doi.org/10.18653/v1/D19-1669

[10] Nijholt, A., Niculescu, A., Valitutti, A., Banchs, R.E.: Humor in human–computer interaction: a short survey. In: Joshi, A., Balkrishan, D.K., Dalvi, G., Winckler, M. (eds.) Adjunct Proceedings: INTERACT 2017 Mumbai. pp. 199–220. Indus- trial Design Centre, Indian Institute of Technology Bombay (2017)

[11] Kolb, W., Miller, T.: Human–computer interaction in pun translation. In: Hadley, J.L., Taivalkoski-Shilov, K., Teixeira, C.S.C., Toral, A. (eds.) Us- ing Technologies for Creative-Text Translation, pp. 66–88. Routledge (2022). https://doi.org/10.4324/9781003094159-4

[12] Ermakova, L., Regattin, F., Miller, T., Bosser, A.G., Borg, C., Jeanjean, B., Math- urin, E., Corre, G.L., Hannachi, R., Araújo, S., Boccou, J., Digue, A., Damoy, A.: Overview of the CLEF 2022 JOKER Task 3: Pun Translation from English into French. In: Faggioli, G., Ferro, N., Hanbury, A., Potthast, M. (eds.) Proceedings of the Working Notes of CLEF 2022: Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings (2022)

[13] Karoui, J., Benamara, F., Moriceau, V., Patti, V., Bosco, C., Aussenac-Gilles, N.: Exploring the impact of pragmatic phenomena on irony detection in tweets: A multilingual corpus study. In: 15th Conference of the European Chapter of the Association for Computational Linguistics. vol. 1, pp. 262–272. Association for Computational Linguistics (Apr 2017),

https://aclanthology.org/E17-1025.pdf
[14] Glémarec, L., Bosser, A.G., Ermakova, L.: Generating humourous puns in French. In: Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th to 8th, 2022. p. 8. CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy (2022)
[15] Dhanani, F., Rafi, M., Tahir, M.A.: FAST_MT participation for the JOKER CLEF-2022 automatic pun and human translation tasks. In: Proceedings of the Working Notes of CLEF 2022 – Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th to 8th, 2022. p. 14. CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy (2022)