FRI Natural language processing course 2023
Project report

University of Ljubljana
Faculty of Computer and
Information Science

Literacy situation models knowledge base creation

Nace Gorenc, Jernej Vrhunc in Žan Pečovnik

Abstract

Advisors: Slavko Žitnik, Aleš Žagar

Introduction

Natural language processing mainly focuses on different approaches of processing, analyzing and understanding large amounts of text. Extracting information from text can have great added value in many aspects of life such as science or economy. In recent years natural language processing became one of the hottest research topic. One of the driving forces was public availability of ChatGPT, which caused quite a stir since people started using it for various task which were formerly done by hand and with a little bit of help from search engines like Google. This report focuses on the analysis of personal relationships between fictional characters, using coreference resolution, named entity recognition (NER) and entity co-occurence graph (ECG).

Initial idea

In this assignment we will focus on creating a model, which will successfuly recognize the relationships (positive, negative, neutral relation) between the characters in any short story. The initial idea is to collect sufficient amount of English short stories and annotate them in such a way that the model we are going to create, will be able to learn general relationships between characters from them. Because acquiring a large enough and exact amount of data is very important important in this kind of assignment, we will do this very precisely and on a large scale of books/stories.

Nextly our plan is to train a model on top of the dataset we created and see how it performs. With this results, we will obtain useful informations on how to improve the model, our dataset and how to possibly change the annotations we created for our dataset.

For extracting information about characters we will use named entity recognition (NER), which is a type of text processing which locates and classifies named entities into predefined categories such as *person*, *organization*, etc. We will use different NERs (Stanza [1], NLTK [2], Flair [3]) and compare the computed results to see which model prodcues the most accurate results.

Before extracting the information about entities from the text we will also perform coreference resolution which is a task of finding referring expressions (he, she, that, this, etc.) that refer to the same entity. For this task we will use the implementation from spaCy [4] and Allennlp [5] libraries to see which works best.

For detecting the actual relationship between characters we will construct a entity co-occurence graph (ECG) and find out from the graph, who interacts with whom and in which way.

Related work

A relation extraction boggled minds of the scientists long before NLP's renaissance driven by the invention of transformers.

In 2011, authors of [6] reviewed multiple supervised and semi-supervised classification approaches to the relation extraction task along with critical analysis. Among the supervised approaches, dependency path kernels stood out as the best both in term of computational complexity and performance. On the other hand, semi-supervised approaches seemed to be well suited for open domain relation extraction systems since they could easily scale with the database size and could extend to new relations easily.

Groza and Corde [7] focused on extracting information about literary characters from folktales. One of the tasks was to identify the main characters and the parts of the story where these characters are described or act. Their system relies on the folktale ontology which is based on Propp's model for folktales morphology and it defines three knowledge sources. The first one is *folktale morphology*, the second one is *folktale main entities* and the last one is *family relationships in folktale*. The authors used three sequential algorithms for retrieving information about characters. The main goals of these algortihms were to extract characters from the folktales, use decoreference resolution and finding the perspective of the characters.

More recently the authors of [8] proposed method of a hybrid approach which combines the features of unsupervised and supervised learning methods, which also uses some rules to extract relationships. The method identifies the main characters and collects the sentences related to them. Then these sentences are analyzed and classified to extract relationships. For testing purposes, 100 short stories were used with around 300 character pair relations. This model identified parent-child relationships, friendships or if there was no relation.

In a recent survey [9], Vincent Labatut and Xavier Bost represented a character network, which can be used for extraction of relations between characters in works of fiction (e.g. novels, movies, plays, TV series). The main goal of this survey was to explain necessary steps, such as character identification, graph extraction and interaction detection, to construct such network.

Dataset analysis

We were not successful in finding any short stories that would already have annotated which characters appear in the stories and the relationships between them. Thus we decided to create our own dataset of stories and annotations for them. In order to create a dataset of sufficient size, we decided to find stories which would have approximately 1000 words or even less in some cases, to not spend too much time on just annotating the data. We found the stories on two different websites, the first one is American Literature [10] which offers a bunch of different stories of different length and from various authors for public reading. The second website is Mom Loves Best [11], which contains parenting tips from real moms and within these tips we also found some moral stories for children.

For the creation of the annotated corpus we created two new files for each selected story. The first file contained the raw text of the story and the second one contained the annotation of this story in JSON format. For each story we annotated:

- which characters appear in the story,
- what are the relationships between these characters.

To get this information we read and analyzed every story, the annotation can be seen below in listing 1. We can see that for each character we evaluated the sentiment relationship between him and everyone else and classified the relationship in 3 different classes:

- **-1**: negative sentiment,
- **0**: neutral sentiment,
- **1**: positive sentiment.

Note that each character has a neutral sentiment with itself. We plan to expand the dataset with additional stories and annotations in the near future. All of the gathered data is available in our GitHub repository [12].

Listing 1. Annotation example

```
{
    "characters": [
        "john wick",
        "dog",
        "killer"
    ],
    "sentiments": {
        "john wick": {
            "john wick": 0,
            "dog": 1,
            "killer": -1
        },
        "dog": {
            "john wick": 1,
            "dog": 0,
            "killer": -1
        },
        "killer": {
            "john wick": -1,
            "dog": -1,
            "killer": 0
        }
    }
}
```

Methods

0.1 NER (Named entity recognition)

To identify characters, we employed Named Entity Recognition (NER), which is a text processing technique that aims to identify and classify named entities into predetermined categories such as PERSON, ORGANIZATION, LOCATION, etc. We utilized various NER models and libraries in our project, including Flair, Spacy and Stanza. Our primary focus was on recognizing people's names to create a character list.

0.1.1 Flair

Flair is a robust NLP library that enables the utilization of cutting-edge natural language processing models on text, including Named Entity Recognition (NER), part-of-speech (PoS) tagging, and specialized support for biomedical data, sense disambiguation, and classification, with a constantly growing range of languages. Additionally, Flair functions as a text embedding library, which means it is used for representing text with its own embeddings, as well as BERT embeddings and ELMo embeddings. Furthermore, it is directly built on PyTorch, which facilitates the training of custom models. Flair representations are based on a bi-LSTM character-based monolingual model that has been pretrained on Wikipedia.

0.1.2 Spacy

Spacy is a Python-based open-source library for advanced natural language processing. While it is primarily designed for production use rather than research, it is still quite user-friendly. Spacy offers a wide variety of models and supports many foreign languages. Spacy's NER model works by using machine learning algorithms to analyze the linguistic features of text, such as part-of-speech tags, dependency parse trees, and context clues. The model is trained on large datasets of annotated text to recognize patterns and associations between words and named entities.

0.1.3 Stanza

Stanza is a suite of efficient and precise linguistic analysis tools that includes advanced NER models for eight different languages. These modules are developed on PyTorch framework, which makes it easy to train and evaluate models using annotated data.

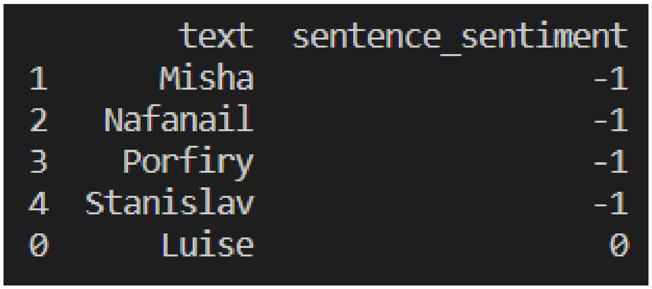
0.2 CR (Coreference resolution)

Coreference resolution is an essential task in natural language processing that identifies all the referring expressions in a text that refer to the same real-world entity. The aim is to cluster these expressions, replacing them with the most appropriate mention to represent the entity, and thus reducing ambiguity in the text. This task can be performed using various libraries, such as spaCy and Allennlp. By performing coreference resolution before named-entity recognition, the ambiguity in the text is reduced, leading to improved performance in NER tasks.

0.3 Sentiment analysis

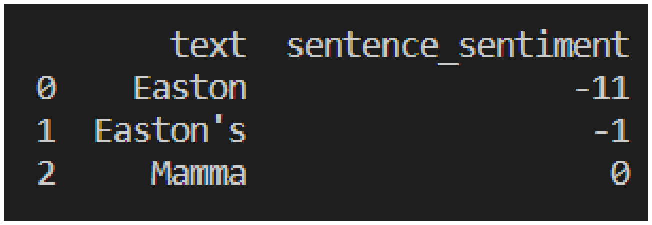
0.3.1 Stanza

After Coreference Resolution (CR) and Named Entity Recognition (NER) were ran on a specific story, we then ran the sentiment analysis on the given processed story with Stanza's Pipeline. This was done on a sentence level, meaning that we knew exactly in which sentence a certain character appears and thus we could evaluate each sentence separately and sum up the scores of each sentence to get the final evaluation. We then got a score for a certain character which could either be less than 0, 0 or greater than 0. If the score was less than 0, then the conclusion would be that he had a negative sentiment. If the score was greater than 0, then the conclusion would be that he had a positive sentiment. And if the score was exactly 0, the character had a neutral sentiment. We can see the examples below in figure 1 and figure 2.



```
        text  sentence_sentiment
1      Misha                  -1
2   Nafanail                  -1
3    Porfiry                  -1
4  Stanislav                  -1
0      Luise                   0
```

Figure 1. Example of sentiment analysis with Stanza's Pipeline of story 003.txt



```
        text  sentence_sentiment
0     Easton                 -11
1   Easton's                  -1
2      Mamma                   0
```

Figure 2. Example of sentiment analysis with Stanza's Pipeline of story 005.txt

Results

NER	Flair		Spacy		Stanza	
CR	**No**	**Yes**	**No**	**Yes**	**No**	**Yes**
Precision	0.81	-	0.90	-	0.85	-
Recall	1.27	-	1.14	-	1.27	-
F1	0.77	-	0.74	-	0.79	-

Table 1. Results

We were not able to run any NERs with the addition of CR, because we always ran out of RAM in Google Colab and the session crashed, meaning we lost all data in the middle of runtime.

Future directions and ideas

Discussion

References

[1] Stanza - nlp package for many human languages. https://stanfordnlp.github.io/stanza/ner.html. Accessed: 19.3.2023.

[2] Natural language toolkit. https://www.nltk.org/. Accessed: 19.3.2023.

[3] Flair - a very simple framework for state-of-the-art nlp. https://github.com/flairNLP/flair. Accessed: 19.3.2023.

[4] Spacy - industrial-strength natural language processing. https://spacy.io/. Accessed: 19.3.2023.

[5] Allen nlp. https://allenai.org/allennlp. Accessed: 19.3.2023.

[6] Nguyen Bach and Sameer Badaskar. A review of relation extraction. 05 2011.

[7] Adrian Groza and Lidia Corde. Information retrieval in falktales using natural language processing. In *2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 59–66. IEEE, 2015.

[8] V. Devisree and P.C. Reghu Raj. A hybrid approach to relationship extraction from stories. *Procedia Technology*, 24:1499–1506, 2016. International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015).

[9] Vincent Labatut and Xavier Bost. Extraction and analysis of fictional character networks. *ACM Computing Surveys*, 52:1–40, 2020.

[10] 100 great short stories. https://americanliterature.com/100-great-short-stories/. Accessed: 2023-04-05.

[11] 20 good short moral stories for kids. https://momlovesbest.com/short-moral-stories-kids. Accessed: 2023-04-05.

[12] Github repository - skupina123. https://github.com/UL-FRI-NLP-Course-2022-23/nlp-course-skupina-123. Accessed: 2023-04-05.