Project titles & abstracts from previous years

Marcel Bollmann

This is a selection of project titles & abstracts that were submitted in previous course instances. They are purely meant as inspiration for your own projects – you can work on the same or similar topics, use the same or similar methods and data, but you can also be as creative as you want and choose something entirely different!

Classifying movie genres based on their summary

The purpose of this project is to see how good and efficient different classifiers are in accurately predicting movie genres based on their plots. We will compare the performance of a Naive Bayes Classifier against a Most Frequent Baseline and a Support Vector Machines Classifier to determine which classifier provides the best and most accurate result. The dataset is retrieved from the website Kaggle and contains thousand different movies, from the Internet Movie Database (IMDB). We will focus solely on extracting the genre and overview (summary). The dataset will be split into two parts, into training and test data, distributing 80% to training data and 20% to test data. We evaluated different classifiers using accuracy, recall, precision and F1-score. From the evaluation we discovered that Naive Bayes had the best results in all measurements except from precision, where Most Frequent Baseline performed better. Most Frequent Baseline had the lowest accuracy but the highest precision, and Support Vector Machines produced average good results through all tests. In general our classifiers predicted overall quite low on accuracy, recall, precision and F1-score – only precision on Most Frequent Baseline got higher than 50%.

Tip: Don't write what you will do in an abstract; write about what you have done!

Is it possible to know where Reddit comments originate from?

This study explores the feasibility of predicting the origin subreddit of a comment based on its content. Utilizing a dataset consisting of one million comments from the top 40 most popular subreddits. Two distinct text classification approaches were employed: a Naive Bayes classifier and a Long Short-Term Memory (LSTM)-based neural network classifier. The performance of these classifiers was evaluated and compared through intrinsic metrics such as accuracy, recall, and precision, aiming to ascertain the effectiveness of content-based subreddit prediction. The results demonstrated that the LSTM-based classifier outperformed the Naive Bayes model in most metrics, indicating a higher effectiveness of LSTM networks in content-based subreddit prediction.

Comparative Analysis of a LSTM and Word2Vec Model for Sarcasm Detection in News Headlines Abstract

This project explores the performance of a Long Short-Term Memory (LSTM) model and a Word2Vec model for sarcasm detection in news headlines. The project utilized pre-existing models created for a specific dataset from the platform Kaggle, featuring both sarcastic and non-sarcastic headlines from “The Onion” and “HuffPost”. The project aims to compare these models on the task to classify sarcasm accurately. Through evaluation involving accuracy metrics and a detailed analysis of preprocessing and training differences, the project findings reveal reasons for different performance of the two models. The result indicates that Word2Vec outperforms LSTM in detecting sarcasm, this due to its suitable handling of word embeddings and semantic relationships. The project also highlights the impact of preprocessing techniques and model configurations on the models’ performance such as lemmatization and hyperparameter settings.

Sentiment analysis over time using senSALDO

Our project has focused on sentiment analysis over time. The application we developed can measure the sentiment of a given article regarding any subjects, such as technological developments or related occurrences in Swedish. The data we used

comes from Mediearkivet, this application allows us to modify and filter words, publications and year. In our program the data is then processed to a desired format and tagged using the POS tags before the sentiment analysis occurs. SenSALDO is the sentiment database used to identify attitudes and sentiment regarding individual words. Our hypothesis revolves around using well known events with an obvious sentiment such as the financial crisis 2008 or the war in Ukraine to evaluate our model. We believe that our application will work for exploring sentiment regarding practically any topic found in Mediearkivet. There should be a direct correlation between worldwide events and sentiment. Illustrated by the examples where we used articles with keywords “finans” (eng: finance), “ekonomi” (eng: economy) and “aktier” (eng: stocks) and looked at their sentiment scores before, during, and after the 2007-2008 financial crisis. Initial results indicate a positive correlation between sentiment and historical events.

Cooking time estimation with LSTM

When it comes to following a recipe, it can be hard to estimate how long the cooking will take. To solve this problem, this project aimed to build a model that can predict cooking time based on just the recipe instructions and ingredient list. The data the model trained and tested on was collected by downloading a dataset of scrapped recipes from an online recipe website and a filter was applied that capped the cooking time at one day maximum and with a z-value smaller than 1 within those recipes. In total 218,813 recipes were used. The model employed a custom word-embedding layer feeding directly into an LSTM-layer. For training we used regression with MSE as a metric. To evaluate the model, we used two evaluation methods – mean squared error (MSE) and root mean squared error (RMSE). These methods calculate the mean squared difference and the root of the mean squared difference between the predicted cooking-time and the gold standard. The value of MSE came out to 329.5 while the value of RMSE came out to 18.15. These results show that the model was within an average error margin of around 18 minutes with respect to the gold standard.

Creativity on Autopilot: Algorithm for Creating Headlines

This is a study where we let AI generate random news headlines, and the goal is to evaluate and experiment with the quality of the text-generating LSTM. To evaluate the LSTM, we decided to use Intrinsic evaluation. Intrinsic evaluation is an evaluation

method that is used in natural language processing (NLP) and machine learning. It is used to assess the performance of a model based on internal capabilities without relying on external tasks or real-world factors. NLP use a metric that is called Cosine similarity, and the metric measures the similarities between two vectors in a multi-dimensional space. Cosine similarity calculate the cosine of the angle between the two vectors, representing their orientation and similarity (Safjan, 2023): LSTM is a type of architecture for recurrent neural networks (RNN), and it is designed to overcome limitations that exists in regular RNNs. These limitations mainly exist when trying to capture and learn long term dependencies in sequential data. LSTM stands for Long Short-Term Memory and addresses the issues in regular RNNs by introducing a more complex cell structure connected to memory. There are four key components of an LSTM unit, these are Cell state, Forget state, Input gate and Output gate (Jurafsky & Martin, 2023).

Tip: This is actually a bit too much technical detail for an abstract; it's much better to include a summary of your results instead!