Computational Linguistics in the Netherlands Journal 13 (2024) 63-90 Submitted 02/2024; Published 03/2024

# Benchmarking Zero-Shot Text Classification for Dutch

**Loic De Langhe*** LOIC.DELANGHE@UGENT.BE
**Aaron Maladry*** AARON.MALADRY@UGENT.BE
**Bram Vanroy**** BRAM.VANROY@KULEUVEN.BE
**Luna De Bruyne***** LUNA.DEBRUYNE@UANTWERPEN.BE
**Pranaydeep Singh*** PRANAYDEEP.SINGH@UGENT.BE
**Els Lefever*** ELS.LEFEVER@UGENT.BE
**Orphée De Clercq*** ORPHEE.DECLERCQ@UGENT.BE

* *LT3, Language and Translation Technology Team, Ghent University, Belgium*

** *CCL, KU Leuven, Belgium*

*** *CLiPS, University of Antwerp, Belgium*

## Abstract

The advent and popularisation of Large Language Models (LLMs) have given rise to prompt-based Natural Language Processing (NLP) techniques which eliminate the need for large manually annotated corpora and computationally expensive supervised training or fine-tuning processes. Zero-shot learning in particular presents itself as an attractive alternative to the classical train-development-test paradigm for many downstream tasks as it provides a quick and inexpensive way of directly leveraging the implicitly encoded knowledge in LLMs.

Despite the large interest in zero-shot applications within the domain of NLP as a whole, there is often no consensus on the methodology, analysis and evaluation of zero-shot pipelines. As a tentative step towards finding such a consensus, this work provides a detailed overview of available methods, resources, and caveats for zero-shot prompting within the Dutch language domain.

At the same time, we present centralised zero-shot benchmark results on a large variety of Dutch NLP tasks using a series of standardised datasets. These tasks vary in subjectivity and domain, ranging from more social information extraction tasks (sentiment, emotion and irony detection for social media) to factual tasks (news topic classification and event coreference resolution). To ensure that the benchmark results are representative, we investigated a selection of zero-shot methodologies for a variety of state-of-the-art Dutch Natural Language Inference models (NLI), Masked Language models (MLM), and autoregressive language models. The output on each test set was compared to the best performance achieved using supervised methods.

Our findings indicate that task-specific fine-tuning delivers superior performance in all but one (emotion detection) task. In the zero-shot settings it could be observed that large generative models through prompting seem to outperform NLI models, which in turn perform better than the MLM approach. Finally, we note several caveats and challenges tied to using zero-shot learning in application settings. These include, but are not limited to, properly streamlining evaluation of zero-shot output, parameter efficiency compared to standard finetuned models and prompt optimization.

## 1. Introduction

In the past decade, a paradigm shift has revolutionized machine learning for Natural Language Processing (NLP). Whereas before, large task-specific training sets with labeled examples were required to develop high-performing NLP systems, pre-trained large language models have shown to improve the state of the art when fine-tuned for dedicated NLP tasks (Raffel et al. 2020, Min et al. 2023). In order to avoid constructing manually-labeled datasets for fine-tuning, researchers have recently started to investigate zero-shot learning (ZSL). The underlying idea is that large language models are expected to inherently incorporate all information needed for classification

because of the large pre-training data. To use the pre-trained models to perform prediction tasks, the original input is modified using a template into a *prompt* that has unfilled slots, and then the language model is used to probabilistically fill in the required information (Liu et al. 2023).

Various approaches have been proposed for ZSL . In the proposed research, we will explore three different methodologies for Dutch, i.e., ZSL (1) via Natural Language Inference (NLI) models, in which the inference task requires reasoning (Yin et al. 2019), (2) via leveraging pre-trained Masked Language Models (MLM) by considering predicted label probabilities when filling mask tokens, as well as (3) via prompting generative auto-regressive language models.

We evaluate the different methodologies on various Dutch NLP tasks, ranging from more factual information extraction tasks (i.e., event coreference resolution, news topic classification) to more subjective or opinionated classification tasks (e.g., irony detection, (aspect-based) sentiment analysis and emotion detection). For each task, high-quality manually-annotated benchmark datasets are used for evaluation. These task-specific datasets have only been used for evaluation purposes and were not incorporated for the pre-training of the underlying large language models. Some of them are publicly available, other ones are available on request. The obtained results are also compared to the state-of-the-art results for the respective tasks, usually obtained by fine-tuning transformer-based models. To the best of our knowledge, this is the first large-scale study of zero-shot experiments for Dutch.

The remainder of this paper is organized as follows. Section 2 introduces relevant related research for ZSL, while Section 3 gives a detailed overview of all tasks and datasets that were evaluated in the experiments. Section 4 describes the applied methodologies, viz. ZSL via Natural Language Inference, Masked Language Modeling and generative AI models. Section 5 presents all experimental results for the different tasks and methods, while Section 6 discusses the main findings of the zero-shot experiments for Dutch. Section 7 presents our conclusions and discusses some challenges and limitations of the investigated zero-shot approaches for Dutch.

## 2. Related Work

The advent of large language models (LLMs) has introduced a shift in NLP from learning task-specific representations to using task-agnostic contextualized embeddings (Devlin et al. 2019, Liu et al. 2019), which have been shown to perform very well for a wide range of NLP tasks. In the latter architectural setup, a final step usually consists of fine-tuning these general-purpose language models for a dedicated task by means of labeled training instances. Radford et al. (2019) have demonstrated, however, that a single pre-trained language model can be *zero-shot* adapted to perform standard NLP tasks, making fine-tuning obsolete. Zero-shot learning refers to machine learning systems aimed at predicting labels without being trained by means of labeled examples for the task at hand.

The first approach to ZSL is via Natural Language Inference (NLI). First, a large pre-trained model is finetuned on a collection of NLI data. Then, the finetuned model needs to decide whether the hypothesis, viz. a prompt which represents the class label (e.g. *The sentiment of this tweet is 'positive'.*), entails or contradicts the premise, viz. the instance (e.g. a tweet) to be classified (Plaza-del Arco et al. 2022). Yin et al. (2019) use a sequence-pair classifier that has to decide whether two sentences (a premise and hypothesis) entail or contradict each other. The challenge in the NLI approach lies in how the prompt should be formulated to best fit the dataset and task at hand.

A second approach to ZSL relies on the fill-in-the-gap abilities of Masked Language Models. In practice, when prompting pre-trained MLMs, the input instance is converted into an input with a masked token in the position where one would expect a label. If we have for instance the input instance "City wins the Manchester Derby against United" in a topic detection task set, this instance will be converted to "City wins the Manchester Derby against United. This topic is about [MASK]". A pre-trained language model, such as BERT (Devlin et al. 2019) or RoBERTa (Liu et al. 2019), can then generate probability scores of all tokens in its vocabulary, and the highest scoring token out of the valid labels is then used as the model's prediction. These prompt-based models have shown

to perform very well for classification tasks, where usually a template is used to transform the task into a cloze task. Subsequently, a *verbalizer* maps the model's prediction into the corresponding class label (e.g. "sports" for the prediction "Sports") (Gao et al. 2021, Qin and Eisner 2021, Cui et al. 2022). As such, the verbalizers function as a bridge between the model output and the final prediction.

Brown et al. (2020) trained GPT-3, an autoregressive language model with 175 billion parameters, and tested its performance in zero- and few-shot settings. They demonstrated good performance when applying GPT-3 without any gradient updates or fine-tuning on many NLP datasets, including translation, question-answering, and cloze tasks for reading comprehension. Another interesting approach is presented by Halder et al. (2020), who introduce TARS (Task-Aware Representation of Sentences), where the classification problem is reformulated as a query of a sentence and a potential class label, and the transformer has to predict whether or not the label holds. The input to a binary sentiment classifier would then consist of both the text to be classified and the possible label, as in the following example: ¡"positive sentiment", "I like this food a lot"¿, with the output being either *True* or *False* depending on whether the label matches the text or not. The same decoder can then be used for various tasks, as the model learns to interpret the semantics of the label name. A big advantage of this approach is that it can return predictions even for classes for which no training data exists: it just needs to prepend the textual label of the new class to text and evaluate the result of the "True/False" decoder.

We conclude this section with two alternative approaches to ZSL. Schick and Schütze (2021) developed a semi-supervised training method that reformulates input instances as cloze-style phrases, which are then used to assign soft labels to a large set of unlabeled examples. Subsequently, standard supervised training is performed on the resulting training set. This approach is shown to outperform other semi-supervised and even supervised training for several tasks and languages in low-resourced settings. Finally, Fei et al. (2022) show that zero-shot classification can be improved by clustering texts in the embedding spaces of pre-trained language models.

## 3. Task Descriptions

In this section the various tasks and Dutch datasets are introduced on which the zero-shot experiments have been conducted. We each time offer some examples and report the state-of-the-art results achieved on these particular datasets.

### 3.1 Sentiment Analysis

**Task Description** Sentiment analysis is the process of using NLP and machine learning techniques to assess and understand the expressed sentiment in an utterance. Typically, the task is to predict whether a sentiment is positive or negative, and optionally a "neutral" category is also included. Applications of sentiment analysis are broad, ranging from social media analysis to marketing surveys and reviews, and even to sentiment analysis of news and politics (Liu 2015).

**Data** In this paper we evaluate text-level sentiment analysis on a limited domain, making use of the Dutch Book Review Dataset (van der Burgh and Verberne 2019, DBRD).[1] It consists of user-written book reviews of the site `hebban.nl`, which means that there is no standardized spelling and that reviews often comprise more than one sentence. Due to the popularity of the website in the Netherlands, there is likely some regional bias towards Dutch spoken in the Netherlands. In the raw data, a review is accompanied by a 5-star rating but to accommodate binary sentiment classification the dataset authors have given 1 and 2-star reviews a *negative* label, whereas 4 and 5 stars were labelled as *positive*. In this dataset there is no *neutral* label. The training set and test set

1. `https://huggingface.co/datasets/dbrd`

are balanced (50% positive/negative) and contain 20,028 and 2224 samples, respectively. For our experiments we only use the test set.

Below an example of a positive and a negative review (spelling and punctuation errors are not corrected) are given.

- **Positive**: Genoten van dit boek, het verhaal word vlot vertelt en de spanning in het slot is om te snijden. Voor mij de beste Jackson tot nu toe. Op naar het volgende boek!
  *EN: Enjoyed this book, the story is very fluent and the ending is very exciting. For me the best Jackson so far. On to the next book!*

- **Negative**: Ik ben niet echt enthousiast over dit boek. Af en toe raakte ik de draad van het verhaal een beetje kwijt ,doordat de schrijfstijl niet altijd even lekker las maar ook door de vaak wat letterlijke vertaling van het boek. Het verhaal had wel wat spannende momenten, maar het plot was niet echt verrassend. Dit boek geef ik dan ook maar twee duimpjes.
  *EN: I am not particularly enthousiastic about this novel. I lost track of the story at certain times, because the writing style is not always as good but also because of the many literal translations present in the novel. The story did have exciting moments, but the plot was not surprising as a whole. This book only deserves two thumbs up.*

**SOTA** The state of the art for this task is currently an accuracy of 95.14% (95% CI [94.25, 96.04]) and an F1 score of 95.14% set by fine-tuning a Dutch RoBERTa-based model, RobBERT v2 (Delobelle et al. 2020).

### 3.2 Aspect-Based Sentiment Analysis

**Task Description** Aspect-Based Sentiment Analysis (or ABSA) is a fine-grained sentiment analysis task that aims to identify and extract sentiment below the text or sentence level (Zhang et al. 2022). This is because the task usually also entails the detection of the targets or aspects towards which sentiment is expressed. Most commonly formulated for online reviews, it can consist of a single sample in which multiple aspects can be defined and a different sentiment can be associated with each of these aspects. For example, as part of a restaurant review, a user may criticize the service but praise the quality of the food. Since the detection of aspects is a token-classification task which can be quite difficult for zero-shot methodologies, we rely on gold-standard aspects and, as such, only evaluate the actual sentiment analysis.

**Data** We use the in-house SentEMO dataset (De Geyndt et al. 2022), which is a collection of 1,000 Dutch reviews per domain from three different domains, namely airline, hotel, and retail. The reviews have been scraped from bol.com, Tripadvisor, and TrustPilot. In all reviews, aspects have been manually labeled and assigned to predefined aspect categories and linked to the appropriate sentiment and emotion. For this research, we only focus on the sentiment classification task and assign the following sentiment labels: *very positive, positive, neutral, negative, very negative* (see example below). For each domain, datasets were split in 90% train and 10% test.

- Review text: Mooie kamer, zeer vriendelijk personeel maar al bij al wel aan de dure kant.
  *EN: Nice room, very friendly staff but all in all on the expensive side.*

- **Aspects & sentiment**:
  "Mooie kamer" (EN: "Nice room") = positive
  "zeer vriendelijk personeel" (EN: "very friendly staff") = very positive
  "aan de dure kant" (EN: "on the expensive side") = negative

**SOTA** The best performing sentiment classification system on this particular dataset is a linear Support Vector Machine built using RobBERT embeddings as features (De Geyndt et al. 2022). For

the Airline domain this system reached a micro F1 or accuracy score of 82.8%. While for the retail domain, the system was able to reach a score 80.9%. Finally, for the Hotel domain, the system reached 85.1%.

### 3.3 Emotion Detection

**Task Description** Emotion detection (also referred to as emotion recognition or emotion analysis) is a more specific form of sentiment analysis, in which the goal is to predict more fine-grained emotion categories like *anger* and *sadness*, instead of just *positive*, *negative* or *neutral* sentiment (Mohammad 2016). Although emotion detection is mostly treated as a classification task, the problem is sometimes defined as a regression task, either by predicting the intensity of specific emotion categories, or by representing emotions in terms of emotional dimensions (like *valence*, *arousal* and *dominance*) instead of categories. In this paper we will only look at emotion detection as a classification problem.

**Data** We use the full EmotioNL dataset (De Bruyne et al. 2021), which is a collection of 1,000 Dutch Tweets and 1,000 captions of Flemish reality TV-shows. The instances are labeled with one of the categories *anger*, *fear*, *joy*, *love*, *sadness* or *neutral*. Additionally, each instance is provided with a score for the emotional dimensions *valence*, *arousal* and *dominance*. In this paper, however, we only make use of the categorical labels in EmotioNL. Below are some examples (one for each label) from the dataset.

- **Anger**: Verwijder me dan maar van de prive insta zodat ik jullie fototjes niet zie face-with-rolling-eyes zielig en belachelijk is dat sorry
  *EN: Then just remove me from the private insta so I don't see your pictures face-with-rolling-eyes That's pathetic and ridiculous sorry*

- **Fear**: Over #fakenews gesproken, als dit ooit als app re-leased gaat worden flushed-face Creepy shit, stemmen nabootsen met max 1 min aan bronsmateriaal
  *EN: Speaking of #fakenews, if this ever gets released as an app flushed-face Creepy shit, mimicking voices with a max of 1 minute of source material*

- **Joy**: @transavia Jaaah slightly-smiling-face volgende vakantie Barcelona en na het zomerseizoen Algarve
  *EN: @transavia Yeahhh slightly-smiling-face next vacation to Barcelona and after the summer season Algarve*

- **Love**: En sindsdien zijn wij altijd op zoek naar onze tweede helft, ons betere helft, en ik hoop dak mijn tweede helft hier naast mij gevonden heb.
  *EN: And since then we have always been searching for our other half, our better half, and I hope that I have found my other half right here beside me.*

- **Sadness**: Ik zou liever sterven dan hier te wonen, denk ik.
  *EN: I'd rather die than live here, I think.*

- **Neutral**: Ja, mijn mama is zelf op vakantie. En mijn papa heeft mij afgezet.
  *EN: Yes, my mother is on vacation herself. And my dad dropped me off.*

**SOTA** In the paper presenting the dataset, the authors fine-tuned and evaluated RobBERT (Delobelle et al. 2020) using 10-fold cross-validation, either fine-tuned on one subset (tweets or captions), or on both subsets. The results (in-domain, cross-domain and multi-domain) were reported separately for the two subsets (see Table 1). In this paper, we will report results on the complete dataset. For comparison: the best fine-tuned RobBERT model from De Bruyne et al.

(2021) – i.e., the multi-domain model – achieves a macro F1 of 40% and accuracy of 52% on the full dataset.

| | Tweets | | Captions | |
|---|---|---|---|---|
| **Model** | **F1** | **Acc.** | **F1** | **Acc.** |
| In-domain | 0.35 | 0.54 | 0.37 | 0.48 |
| Cross-domain | 0.28 | 0.43 | 0.27 | 0.35 |
| Multi-domain | 0.38 | 0.52 | 0.40 | 0.52 |

Table 1: State-of-the-art results on EmotioNL (fine-tuned RobBERT).

### 3.4 Irony Detection

**Task Description** Irony detection is a binary classification task that aims to identify whether a text contains irony. Irony is a type of figurative language that is often used to convey the exact opposite of a statement's literal interpretation (Wilson and Sperber 2012). As this allows people to convey criticism and disapproval in a less direct manner, it is often used as a face-protecting communication strategy (Brown and Levinson 1987). Sarcasm is technically considered to be a form of verbal irony that is intended to ridicule, insult or hurt someone (Clift 1999). However, popular use of the term *sarcasm* indicates that it is often used to denote all forms of verbal irony (Wilson and Sperber 2012, Sulis et al. 2016). Therefore, we use the term *irony* to denote both irony and sarcasm in the broad interpretation.

**Data** For irony detection, we employ a balanced corpus of Dutch tweets that was gathered and fully manually annotated by Van Hee et al. (2016). The ironic tweets in this corpus were collected using the irony-related hashtags #ironie, #sarcasme and #not as search terms for the Twitter API. The supplementary non-ironic tweets were gathered from the same users who had also posted the ironic tweets. The resulting balanced corpus of 5,566 tweets was annotated in its original form, after which the irony-related hashtags were removed. This corpus uses the same collection and annotation strategy as the corpus used for SemEval 2018 Task 3 (Van Hee et al. 2018), one of the popular English benchmark datasets for irony detection. Although our Dutch corpus also contains fine-grained annotations describing the types of irony used (irony by clash, situational irony and other), we only use the annotations for binary classification (ironic or not ironic).

- **Ironic**: Verkouden en hooikoorts. Echt een goei combinatie ze #fml.
  *EN: Cold and hay fever. Really nice combination #fml.*

- **Not ironic**: Opstaan met barstende koppijn, hatelijk
  *EN: Waking up with a splitting headache, hate it*

**SOTA** The state of the art for irony detection on this dataset is achieved by fine-tuning a multilingual transformer model: XLM-RoBERTa-large (Conneau et al. 2020). As shown in Table 2, this model reaches a macro F1-score of 77% on a held-out test set of 1,113 instances (Maladry et al. 2023). However, highly similar performance can also be achieved with monolingual fine-tuned models such as BERTje (de Vries et al. 2019) with a score of 70%, or RobBERT (Delobelle et al. 2020) with a score of 74%, and the multilingual Twitter RoBERTa (Barbieri et al. 2022), with a score of 75%.

| Model | Precision | Recall | **F1** | **Acc.** |
|---|---|---|---|---|
| XLM-RoBERTa-large | 0.75 | 0.75 | 0.77 | 0.77 |
| XLM-RoBERTa-twitter | 0.75 | 0.75 | 0.75 | 0.75 |
| RobBERT | 0.74 | 0.74 | 0.74 | 0.74 |
| BERTje | 0.71 | 0.71 | 0.71 | 0.71 |

Table 2: Results of fine-tuned transformer models for irony detection.

### 3.5 News Topic Classification

**Task Description** News Topic Classification allows to automatically derive a first rough idea of the theme of a certain news article. Unlike topic modelling this classification task entails a predefined taxonomy (Kowsari et al. 2019). Such a taxonomy can range from high-level to very fine-grained topics. A typical example for news articles is the IPTC (International Press Telecommunications Council) standard which comprises 17 top-level topics (e.g., crime, education or politics) which are divided in increasingly granular subtopics (e.g., law enforcement, higher education or election)[2].

**Data** We have access to the dataset introduced in De Clercq et al. (2020), i.e., a large number of Dutch news article data from the 2018-2019 calendar year. This collection, sourced from a variety of (online) Flemish newspapers, contains a total of 235,726 instances which were (automatically) labeled by a proprietary IPTC classifier using the aforementioned 17 top-level IPTC topics. Given that this proprietary IPTC classifier also assigns a confidence score, all instances with a confidence score lower than 75% were removed from the dataset, after which a training, develop and test split were created according to a 8:1:1 ratio. This led to datasets of 150,880 train, 18,824 development and 18,864 test instances, respectively. Important to note is that individual articles can be assigned multiple topics, see below for an example. Due to the size of this particular test dataset as well as the length of the individual instances, performing zero-shot experiments using a large number of models would require significant time, computational and monetary resources. Therefore, we opted to create a restricted test set of 1,000 instances which was sampled from the available original test set and maintains the same distribution of news topics (De Clercq et al. 2020).

- Omstreden glyfosaat mag vijf jaar langer gebruikt worden, ondanks tegenstem van België: Onbegrijpelijk. De lidstaten van de Europese Unie hebben maandag in een beroepscomité beslist om de vergunning voor glyfosaat met vijf jaar te verlengen. Voor het eerst schaarden voldoende lidstaten zich achter een nieuwe toelating voor het veelbesproken onkruidbestrijdingsmiddel. Dat is vernomen bij federaal minister van Landbouw Denis Ducarme. België stemde tegen. Greenpace is verrast en geschandaliseerd door de beslissing , Test-Aankoop roept dan weer op om de verkoop aan particulieren te verbieden.
  *EN: Controversial glyphosate may be used for five more years, despite Belgium's negative vote: Incomprehensible. In an appeal committee on Monday the European Union's member states have decided to extend the permit for glyphosate by five years. For the first time sufficient member states endorsed a new permit for the controversial herbicide. This was learned from Federal Minister of Agriculture Denis Ducarme. Belgium was opposed. Greenpace is surprised and scandalized by the decision, Test-Aankoop calls for a ban on sales on the private market.*

- **IPTC topics**: politics, environment

**SOTA** The state-of-the-art results (De Clercq et al. 2020) on the full test dataset are a macro F1 of 86.4% and a micro F1 or accuracy of 90.3%. These results were achieved by fine-tuning BERTje (de Vries et al. 2019) on the classification task at hand.

2. `https://www.iptc.org/standards/media-topics/`

### 3.6 Event Coreference Resolution

**Task Description** Event Coreference Resolution (ECR) is a discourse-oriented NLP task in which the primary goal is to find textual references that refer to the same happening, be it a fictional or real-world event (Lu and Ng 2018). Typically, the textual representation of an event is designated as an *event mention*. Consider the following event mentions:

- SP.A brengt winterjassen bijeen voor kansarmen
  *EN: SP.A gathers winter coats for the underprivileged*
- Op de Werelddag tegen Armoede hield de SP.A van Dendermonde op de binnenkoer van het ABVV in de Dijkstraat een inzameling van winterjassen
  *EN: On the International Day for the Eradication of Poverty the SP.A faction of Dendermonde organised a collection of Winter coats on the ABVV courtyard in the Dijkstraat*

While human readers can easily apply their extra-linguistic knowledge to determine that the two events mentioned above do indeed refer to the same real-word event, this is no trivial task for most AI algorithms. This task is often framed as a binary pairwise classification task in which two event mentions are either deemed coreferent (1) or non-coreferent (0). For the events above, this means that the model would need to assign a coreferential link.

**Data** We use the Dutch ENCORE event coreference dataset (De Langhe et al. 2022), an annotated subset of the news article data described in Section 3.5. The dataset includes 15,000 events and their coreferential links, annotated across the entire document collection. In total, 2,605 event coreference chains exist that contain two or more events in the corpus. Of those coreference chains, 1,018 are intra-document chains, i.e, chains contained within a single document and another 1,587 are cross-document chains, i.e, chains spanning multiple documents. As most events are not part of the same coreferential chain, the data is inherently imbalanced and skewed towards non-coreferring event mentions. In a standard pairwise classification setting only 2% of the event pairs corefer. As was the case in Section 3.5 we work with a restricted test set of 1,000 instances as performing zero-shot classification on all event pairs in the entire training set $(\pm 2 million instances) would pose too much strain on computational resources.$

**SOTA** The state-of-the-art results for this task are currently obtained by combining the output of an event-aware pairwise classifier (Yao et al. 2023) with a series of graph-based coreference reconstruction algorithms (De Langhe et al. 2023). In Table 3 we briefly summarize the best results obtained on the ENCORE corpus using the aforementioned pairwise classifier with a BERTje encoder (de Vries et al. 2019) coupled with both a Graph Auto-encoder (GAE) and probabilistic Variational Graph Auto-Encoder (VGAE) (Kipf and Welling 2016) as the reconstruction algorithm. Note that coreference-based tasks are evaluated using the CONLL F1 metric, an average of three commonly used metrics for coreference evaluation: MUC (Vilain et al. 1995), $B^3$ (Bagga and Baldwin 1998) and CEAF (Luo 2005).

| Base Model | Reconstruction Algorithm | CONLL F1 |
|---|---|---|
| Event-aware pairwise classifier | GAE | 0.74 |
| Event-aware pairwise classifier | VGAE | 0.73 |

Table 3: Results of a SOTA pairwise coreference classifier using 2 graph reconstruction algorithms.

## 4. Zero-Shot Methodologies

In the following sections we describe in more detail the three zero-shot methodologies we aim to evaluate. Note that in this paper our goal is to estimate the performance of baseline zero-shot

approaches. Therefore, we approach the creation of task-specific prompts from the perspective of a "generic" or "average user" and do not perform extensive prompt engineering. We believe that prompts should be user-friendly, intuitive and realistic rather than highly complex phrasings intended to extract optimal results, especially because the general public is ultimately the intended end user for many of these applications. Additionally, we use each of the models with their default out-of-the-box settings and keep all prompting parameters (further described below) consistent. For each of the methods we assume a general text classification setting in which we have access to a text span or instance $s$, a prompt template $p$, to be used with each of the aforementioned text spans, and a restricted label set $Y$, which contains all possible labels to be used in the classification. Finally, note that the examples given in the sections below are often trivial and only meant to illustrate the methods discussed in an intuitive manner. For a comprehensive overview of the various prompts used in each individual task we refer the reader to appendix A.

### 4.1 NLI

Natural Language Inference (NLI) is most often framed as a sentence-pair classification task in which the relationship between two sentences is to be determined. The first sentence in the pair is denoted as the *premise* and the other as the *hypothesis*. Three possible relationships are defined within the NLI task: *entailment*, for which the hypothesis is a natural consequence of the premise, *contradiction* which indicates an impossibility between the two sentences and *neutrality* in which no explicit sentence relationship can be found between the premise-hypothesis pair.

Trained NLI classifiers are effective zero-shot learners as most text classification tasks can be framed as an NLI objective. The only requirements are a prompt template $q$ and a defined label set $Y$. Consider the following example for sentiment classification where $q$ = *This review is* {} and Y = {*Positive, Negative*}:

**Premise**: this movie is great!

**Hypothesis**$_1$: This review is *positive*

**Hypothesis**$_2$: This review is *negative*

Given the premise, we can evaluate the likelihood of possible hypotheses by determining the score for the entailment label. Then, we choose the most likely hypothesis after applying a softmax activation function over all obtained entailment scores.

To use NLI as a zero-shot classification methodology for Dutch, designated Dutch NLI models were fine-tuned for the research presented here. There are two publicly available datasets for NLI classification in Dutch: the SickNL corpus (Wijnholds and Moortgat 2021), which contains around 10,000 sentence pairs that were automatically translated from the larger English SICK (Marelli et al. 2014) dataset and an automatically translated version[3] of the SNLI benchmark dataset (Bowman et al. 2015), which contains around 550,000 sentence pairings. Both datasets were merged, while reserving 10% of the total pairings for evaluation. Three monolingual Dutch NLI classifiers were trained based on BERTje (de Vries et al. 2019), RobBERT (Delobelle et al. 2020) and RobBERTje (Delobelle et al. 2021) models respectively. Each of these models was trained for 4 epochs using a learning rate of 2e-5 and are publicly available[4] through the Huggingface framework (Wolf et al. 2019). The trained models were then evaluated on the aforementioned held-out test set. Additionally, we include a recently developed multilingual NLI model based on DeBERTaV3 (He et al. 2021) in all NLI prompt tasks. DeBERTa v3 was pre-trained on a large variety of languages (including Dutch) and was then fine-tuned for NLI on a subset of those languages.[5] For

3. https://huggingface.co/datasets/jegormeister/dutch-snli
4. https://huggingface.co/LoicDL/bert-base-dutch-cased-finetuned-snli
5. https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7

completeness, we also provide the results for the DeBERTaV3 model on our held-out test set in Table 4 below. Note, however, that this result is only provided to give an indication of the model's capabilities and that there is no overlap or relation between the merged monolingual NLI datasets and the DeBERTaV3's fine-tuning data.

| **model** | **F1-score** | **Accuracy** |
|---|---|---|
| BERTje | 0.86 | 0.86 |
| RobBERT | 0.88 | 0.87 |
| RobBERTje | 0.84 | 0.83 |
| DeBERTaV3 | 0.92 | 0.91 |

Table 4: General performance of the models fine-tuned for NLI.

Overall, the obtained performance of the Dutch models is in line with earlier studies on Dutch NLI fine-tuning (Delobelle et al. 2021). As expected, the multilingual DeBERTaV3 model outperforms the monolingual models on the same evaluation set due to its more complex architecture (He et al. 2021).

### 4.2 MLM

As discussed in Section 2, most masked language models can be effectively used as zero-shot learners by framing a classification problem as a cloze task. For each label $y \in Y$ the cloze probability for $y$ in the construction $s + q$ is evaluated, where $s$ is the instance to be classified and $q$ the prompt for this task using a specific classification token (here *positive* and *negative*). Consider the examples below for which $s$ = *This movie is great* and $q$ = *This review is [MASK]*, where `[MASK]` is a generalization. In practice the token differs per model.

$$\mathrm{P}_{positive} = \text{this movie is great! This review is } positive$$

$$\mathrm{P}_{negative} = \text{this movie is great! This review is } negative$$

The ranking of classification labels is then obtained by simply computing the individual cloze probabilities for $y \in Y$ and applying a softmax activation function over the obtained logits for each of the labels in Y.

However, zero-shot learning using cloze-style MLMs is hindered by two significant problems. First, unlike NLI methods, which are evaluated at the sentence level, masked language models are inherently biased towards some words depending on the prompt template $q$. Consider the case where the instance $s$ would be removed from the example above. The cloze probabilities p(*Positive*| $q$) and p(*Negative*| $q$) will in practice never be equal due to the default skew in training data. This makes zero-shot classification using out-of-the-box masked language models highly impractical as there will always be a bias to some of the labels, depending on the individual model and its training data. Following earlier work on cloze-style zero-shot learning (Zhang and Hashimoto 2021) we attempt to negate the inherent bias towards some labels by first calculating the baseline probability of a label given the bare prompt template $q$ and subtracting it from the obtained cloze probability for each label y given the zero-shot prompt s+p:

$$cloze_{normalized} = p(y \mid s + q) - p(y \mid q)$$

A second problem with cloze-style prompting are out-of-vocabulary labels. As multi-word labels can be composed of multiple subword vocabulary tokens, the cloze probability has to be computed over multiple tokens. For instance, the label *science and technology* in the news topic classification task (Section3.5) can be tokenized as ['science', 'and', 'tech', '#no', '#logy']. Given that most

predictions in MLM-based models are skewed towards predicting single tokens rather than more complex token sequences, this is problematic. A possible mitigation strategy could be to find suitable synonyms for multi-token labels. However, this is not always feasible and is likely to result in a suboptimal label. When it was not possible to find a suitable synonym, we therefore decided to not present the MLM results for that model.

### 4.3 Generative LLMs

Chat-based LLMs provide an intuitive and user-centric approach to leveraging the vast knowledge on which they have been pre-trained. Typically, these models are transformer-based decoder-only architectures which are trained with a causal language modelling objective, followed by a refinement process through reinforcement learning from human feedback (RLHF). [6] This results in a powerful chat-based model which can be interacted with through a prompt $q$, often in the form of a direct question. Additionally, a key part of chat-based LLMs is the system prompt $sp$, which contains an initial set of instructions that serve as the starting point when starting a chat session. These instructions aim to focus the models' capabilities in a certain direction and can often have a significant effect on the responses to the general task-specific prompt/question $q$.

In our experiments for generative LLMs, we evaluated two general chat-based LLMs: OpenAI's gpt-3.5-turbo and Meta's LLama 2 model (both the version with 7B and with 13B parameters). These models are mainly intended for English. For Llama 2, the usage on languages other than English is even noted as "out of scope". Therefore, we also introduce in this paper a Llama 2 (13B) model that has been fine-tuned on Dutch data (Vanroy 2023). With this model, our aim is to investigate whether a language-specific model can attain improved results over the baseline Llama 2 models. For each of these models, we include the system prompt that was used to train the models. For gpt-3.5-turbo, the system prompt is automatically included when using the API. For the Llama models, we used the original English system prompt for the baseline models and the Dutch translation of the system prompt that was used to fine-tune the Dutch Llama 2 model (please refer to appendix A for more details).

As mentioned above we do not perform extensive prompt engineering and maintain all default model parameters. In short, this means we used BitsAndBytes to load the model 4-bit for efficiency reasons (Dettmers et al. 2023) while keeping the temperature set to 1 and using sampling with top_p=1 (disabled) and top_k = 50. For some tasks, such as emotion and irony detection, we found that the systems were prone to diverge from the classification task. As such, they would omit answering the classification question directly and instead provide an answer or explanation that does not match the predefined labels. This makes any automatic evaluation of the system impossible. Therefore, we used the `outlines` library (Willard and Louf 2023) to limit the output of all generative models (Llama 2-based and gpt3.5-turbo) to the predefined classification labels for each task. For each text sample, we always use a new prompt so we did not use batching.

## 5. Results

In this section the zero-shot results for every task are presented. We each time report the evaluation metric(s) as mentioned in Section 3 and compare the different zero-shot methodologies to the state-of-the-art (SOTA) result. The exact prompt templates that were used can be found in Appendix A.

### 5.1 Sentiment Analysis

For classifying book reviews to either positve or negative sentiment, we use the labels "positief" and "negatief". The results (Table 5) indicate that the state-of-the-art finetuned model is hard to beat. MLM performs worse across the board. NLI models perform better, especially the multilingual model

6. https://openai.com/research/instruction-following

based on DeBERTa, which even outperforms the English Llama 2 chat models. `gpt-3.5-turbo` and its 175B parameters outperforms other approaches with an F1 score of 87%, but other generative models seem inconsistent. The Dutch Llama 2 chat model does perform well, with an F1 score of 82%, which is a big step up compared to the English version (41% F1). In fact, this English version performs worse than a tiny 74M finetuned NLI model based on RobBERTje (56% F1). It is unclear why the performance of this specific model is so bad whereas the finetuned Dutch version as well as the 7 billion variant do perform well.

| **model** | **type** | **f1** | **acc.** | **param.** |
|---|---|---|---|---|
| robbert-v2-dutch-base | finetuned | 0.95 | 0.95 | 117M |
| gpt-3.5-turbo | generation | 0.87 | 0.87 | 175B |
| Llama-2-13b-chat-dutch | generation | 0.82 | 0.82 | 13B |
| Llama-2-13b-chat-hf | generation | 0.41 | 0.53 | 13B |
| Llama-2-7b-chat-hf | generation | 0.71 | 0.71 | 7B |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.75 | 0.76 | 279M |
| bert-base-dutch-cased-finetuned-snli | nli | 0.52 | 0.59 | 109M |
| robbert-v2-dutch-finetuned-snli | nli | 0.41 | 0.53 | 117M |
| robbertje-dutch-finetuned-snli | nli | 0.56 | 0.56 | 74M |
| bert-base-dutch-cased | mlm | 0.34 | 0.47 | 109M |
| robbert-v2-dutch-base | mlm | 0.34 | 0.50 | 117M |
| robbertje-1-gb-merged | mlm | 0.34 | 0.50 | 74M |

Table 5: Results on Sentiment Analysis

### 5.2 Aspect-Based Sentiment Analysis

For Aspect-Based Sentiment Analysis, the traditional label set [“very negative”, “negative”, “neutral”, “positive”, “very positive”] was translated to Dutch as [“zeer negatief”, “negatief”, “neutraal”, “positief”, “zeer positief”]. These labels were used consistently for all systems and approaches with the exception of all MLM approaches. As it would not be fair to compare the probability of a multi-word label to a single-word label, the labels “zeer negatief” (very negative) and “zeer positief” were substituted to “vreselijk” (“horrible” or “terrible”) and “geweldig” (“great” or “awesome”). The aspects themselves do not need translating, as they are a piece of the Dutch input text.

| **model** | **type** | **Airline** | **Retail** | **Hotel** | **Mean** |
|---|---|---|---|---|---|
| robbert-v2-dutch-base | finetuned | 0.82 | 0.81 | 0.851 | 0.83 |
| gpt-3.5-turbo | generation | 0.41 | 0.36 | 0.40 | 0.39 |
| Llama-2-13b-chat-dutch | generation | 0.25 | 0.31 | 0.24 | 0.27 |
| Llama-2-13b-chat-hf | generation | 0.33 | 0.36 | 0.36 | 0.35 |
| Llama-2-7b-chat-hf | generation | 0.14 | 0.13 | 0.11 | 0.13 |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.38 | 0.36 | 0.45 | 0.40 |
| bert-base-dutch-cased-finetuned-snli | nli | 0.24 | 0.26 | 0.22 | 0.24 |
| robbert-v2-dutch-finetuned-snli | nli | 0.29 | 0.28 | 0.30 | 0.29 |
| robbertje-dutch-finetuned-snli | nli | 0.15 | 0.14 | 0.10 | 0.13 |
| bert-base-dutch-cased | mlm | 0.20 | 0.24 | 0.17 | 0.20 |
| robbert-v2-dutch-base | mlm | 0.15 | 0.10 | 0.14 | 0.13 |
| robbertje-1-gb-merged | mlm | 0.26 | 0.22 | 0.24 | 0.24 |

Table 6: Results on Aspect-Based Sentiment Analysis

In Table 6, we present the results. The results for each of the zero-shot approaches is significantly lower compared to the fine-tuned model, which reaches a mean F1-score of 83% across the three domains (Airline, Retail and Hotel). The best zero-shot approach for this task is NLI with DeBERTa, which achieves a mean micro F1 or accuracy score of 40%, closely followed by the generative gpt-3.5-turbo model with a score of 39%. Surprisingly, the Dutch Llama 2 model does not perform as well as the original English version, with a score of 27% compared to 35%. The remaining NLI systems achieve higher scores than the MLM approaches with the same base models, with the exception of the smallest model RobBERTje.

### 5.3 Emotion Detection

We used the following Dutch translations for the emotion labels: "neutraal" (neutral), "woede" (anger), "angst" (fear), "vreugde" (joy), "liefde" (love) and "verdriet" (sadness). These words occur in the vocabulary of all MLM models (in contrast to other potential translations, like "boosheid" as a translation for anger). We were thus able to use the same labels for all tested models.

| **model** | **type** | **f1** | **acc.** |
|---|---|---|---|
| robbert-v2-dutch-base | finetuned | 0.40 | 0.52 |
| gpt-3.5-turbo | generation | 0.46 | 0.50 |
| Llama-2-13b-chat-dutch | generation | 0.15 | 0.20 |
| Llama-2-13b-chat-hf | generation | 0.19 | 0.36 |
| Llama-2-7b-chat-hf | generation | 0.04 | 0.09 |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.26 | 0.30 |
| bert-base-dutch-cased-finetuned-snli | nli | 0.16 | 0.19 |
| robbert-v2-dutch-finetuned-snli | nli | 0.08 | 0.18 |
| robbertje-dutch-finetuned-snli | nli | 0.15 | 0.21 |
| bert-base-dutch-cased | mlm | 0.14 | 0.19 |
| robbert-v2-dutch-base | mlm | 0.23 | 0.33 |
| robbertje-1-gb-merged | mlm | 0.15 | 0.18 |

Table 7: Results on Emotion Detection

The results for emotion detection are shown in Table 7. The best performing model for emotion detection is gpt-3.5-turbo. With a macro-F1 score of 46%, this model even outperforms the fine-tuned RobBERT model (F1 of 40%). In terms of accuracy, however, RobBERT does perform slightly better than the best zero-shot model (52% for RobBERT compared to 50% for gpt-3.5-turbo). The other generative models perform considerably worse. Especially the smallest Llama 2 model performs poorly (F1 of 5%) and is in fact the worst performing model across all methodologies on this task. The English model with 13B parameters is the best Llama 2 model, but still only achieves an F1-score of 19%. There is much variability in the performance of the NLI models, with the RobBERT-based model performing worst (F1 of 8%) and the multilingual model performing best (F1 of 26%). For the MLM approach, however, RobBERT is the best model (F1 of 23%). Generally, the performance of all models is low, even for the fine-tuned model. However, fine-tuning was done on only a small number of data points (1,900 instances). This is probably not sufficient to largely outperform a model like gpt-3.5-turbo (even in a zero-shot setting), which is trained on a large amount of data (570GB) and has over 175B parameters. Moreover, the data on which gpt-3.5-turbo is trained includes a wide range of texts coming from various sources, including social media, forums, blogs, etc. We assume that a considerable part of this data contains expressions and discussions related to emotions.

### 5.4 Irony Detection

The usual classification labels used for irony detection are "not ironic" and "ironic". However, we changed the label names for the zero-shot approaches because they rely on generating the label explicitly. In this paper, we changed the name of the not-ironic label to "genuine" to evade the potential of confusion caused by the negation ("not ironic"). The renamed "genuine" still captures the same conceptual meaning as ironic tweets are intended to convey a different figurative meaning compared to the literal expression. As discussed in the task description, the concepts of irony and sarcasm are strongly connected. Furthermore, some research suggests that the popular use of the bterm "sarcasm" is overtaking the meaning of "verbal irony" as a whole (Bryant and Fox Tree 2002, Gibbs 1986). As these zero-shot approaches could be affected by this semantic shift, we investigated the impact of both the terms *irony* and *sarcasm*. A preliminary investigation of the experiments with only "sarcastic", only "ironic" or a combination of both ("ironic—sarcastic") revealed no meaningful difference in the scores. Therefore, we only show the results for "ironic" as irony label.

The results are presented in Table 8 and show that the best performing generative model for irony detection is gpt-3.5-turbo with and F1-score of 60%, followed by the Dutch Llama 2 model, with an F1-score of 51%. For this task, the NLI and MLM approaches perform rather poorly, often not reaching an F1-score of 50%. Compared to the fine-tuned RoBERTa models, which are the SOTA for this dataset, even the best zero-shot approach reaches about 10% lower F1-scores. We hypothesize that this is related to the limited degree of explicitness of irony. As a form of figurative language, irony is inherently non-literal, which makes it especially harder to identify using a single word prompt, which is the case for NLI and MLM. We could not perform some of the MLM experiments because the models had no appropriate tokens in their vocabulary. For the generative models, we notice that systems with a higher parameter count score better which makes sense because this allows them to capture more nuances, which are essential to understand the irony.

| **model** | **type** | **f1** | **acc.** |
|---|---|---|---|
| XLM Roberta-large | finetuned | 0.72 | 0.72 |
| gpt-3.5-turbo | generation | 0.60 | 0.60 |
| Llama-2-13b-chat-dutch | generation | 0.52 | 0.52 |
| Llama-2-13b-chat-hf | generation | 0.51 | 0.51 |
| Llama-2-7b-chat-hf | generation | 0.48 | 0.48 |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.38 | 0.50 |
| bert-base-dutch-cased-finetuned-snli | nli | 0.43 | 0.46 |
| robbert-v2-dutch-finetuned-snli | nli | 0.46 | 0.46 |
| robbertje-dutch-finetuned-snli | nli | 0.35 | 0.48 |
| bert-base-dutch-cased | mlm | 0.44 | 0.46 |
| robbert-v2-dutch-base | mlm | 0. | 0. |
| robbertje-1-gb-merged | mlm | 0. | 0. |

Table 8: Results on Irony Detection

### 5.5 News Topic Classification

As stated in Section 3.5, news texts are classified into the 17 top-level categories of the IPTC standard. These are: *arts, culture, entertainment and media (1)*, *conflict, war and peace (2)*, *crime, law and justice (3)*, *disaster, accident and emergency incident (4)*, *economy, business and finance (5)*, *education (6)*, *environment (7)*, *health (8)*, *human interest (9)*, *labour (10)*, *lifestyle and leisure (11)*, *politics (12)*, *religion (13)*, *science and technology (14)*, *society (15)*, *sport (16)* and *weather (17)*. Note that many of these labels consist of multi-word expressions, making it hard to find fitting in-vocabulary alternatives for the MLM method. Additionally, some of the words in the labels

cover widely different aspects of a topic (e.g. *war* and *peace*), which further complicates finding appropriate single token labels. In general, we found no suitable manner to integrate this complex label set in the MLM setting and it is for this reason that no MLM results are reported for this task.

Considering the results presented in Table 9, we find that generation-based models show the best performance for this task, with gpt-3.5-turbo attaining a macro F1 score of 56%. While the multilingual DeBERTaV3 NLI model's macro F1 is fairly close in performance to those of the lower-scoring generation models (Llama-2-7b-chat-hf an Llama-2-13b-chat-hf), most other NLI models performed quite poorly. Note that there is still a significant gap in performance when compared to the fine-tuned SOTA model (84%) and the best performing zero-shot method (56%). A possible explanation might be found in the fact that most generative models tend to overestimate the number of categories a given text should be assigned. While most instances have 1-2 gold labels for this multiclass task, generative models predict 4-5 labels on average, with labels such as *human interest* or *society* being assigned to most instances. A similar problem arises for most of the NLI models, where we found that the majority of instances (up to 81% for robbertje-dutch-finetuned-snli) were assigned to the more 'general' labels in the label set (i.e. *human interest*, *society* and *lifestyle and leisure*).

| **model** | **type** | **f1** | **acc.** |
|---|---|---|---|
| BERTje | finetuned | 0.84 | 0.81 |
| gpt-3.5-turbo | generation | 0.56 | 0.53 |
| Llama-2-13b-chat-dutch | generation | 0.52 | 0.49 |
| Llama-2-13b-chat-hf | generation | 0.47 | 0.44 |
| Llama-2-7b-chat-hf | generation | 0.43 | 0.38 |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.38 | 0.23 |
| bert-base-dutch-cased-finetuned-snli | nli | 0.26 | 0.09 |
| robbert-v2-dutch-finetuned-snli | nli | 0.14 | 0.01 |
| robbertje-dutch-finetuned-snli | nli | 0.12 | 0.01 |
| bert-base-dutch-cased | mlm | 0. | 0. |
| robbert-v2-dutch-base | mlm | 0. | 0. |
| robbertje-1-gb-merged | mlm | 0. | 0. |

Table 9: Results on News Topic Classification

### 5.6 Event Coreference Resolution

In our prompts, event coreference resolution was framed as a binary pairwise task in which two events are given and a yes or no answer by the model is expected. While this poses no conceptual problem for generation models which are, by default, suited to deal with direct questions, NLI and MLM prompts did require some rephrasing.

Looking at the results (Table 10) we observe that the gpt-3.5-turbo model performs best for the event coreference task, followed by the finetuned Dutch Llama-2 model. The results for this task can be described as a mixed bag. Even though the phrasing of the NLI and MLM prompts is far from ideal, some of these models' performance (robbertje-dutch-finetuned-snli and robbertje-1-gb-merged in particular) are not too far off the best performing generative models. Interestingly, we note a consistent overestimation of coreferring events across all methodologies. Concretely, the majority of event pairs are deemed to be coreferent even when they are not. When examined in more detail, we found that a coreferring relation is almost always drawn between events when the two events have the same overarching type or theme (i.e., The Gulf War and The Vietnam War). This is also true for the generative models, even though a specific definition of coreferring events is included in

the prompts (Appendix A). Overall, this seems to indicate that most of the models struggle with distinguishing coarse from fine-grained events.

| model | type | CONLL f1 |
|---|---|---|
| BERTje | finetuned | 0.746 |
| gpt-3.5-turbo | generation | 0.453 |
| Llama-2-13b-chat-dutch | generation | 0.429 |
| Llama-2-13b-chat-hf | generation | 0.395 |
| Llama-2-7b-chat-hf | generation | 0.408 |
| mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 | nli | 0.400 |
| bert-base-dutch-cased-finetuned-snli | nli | 0.38 |
| robbert-v2-dutch-finetuned-snli | nli | 0.436 |
| robbertje-dutch-finetuned-snli | nli | 0.418 |
| bert-base-dutch-cased | mlm | 0.261 |
| robbert-v2-dutch-base | mlm | 0.328 |
| robbertje-1-gb-merged | mlm | 0.381 |

Table 10: Results on Event Coreference Resolution

## 6. Discussion

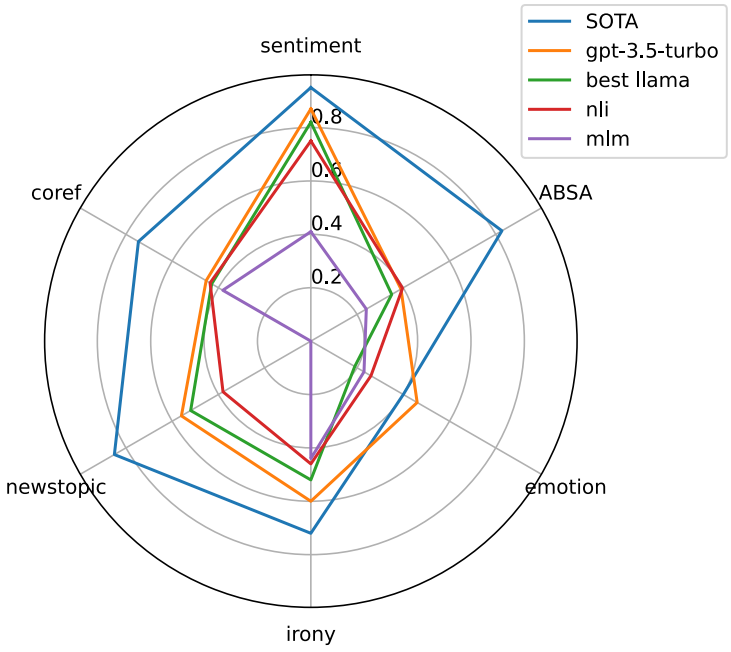


Figure 1: General performance on each task

In the radar chart presented in Figure 1, we compare the performance of each approach's best model together with the SOTA results. The best Llama model is generally the Dutch fine-tuned 13B Llama, with the exception of emotion detection and ABSA, where the English 13B model performed

better. Overall, this English Llama model with 13B, in turn, generally performed better than its 7B counterpart, except for corefence and sentiment analysis. For NLI, DeBERTa reaches the best performance for most tasks, except for coreference resolution and irony detection, where RobBERT produced the best results. For MLM, the best models vary widely depending on the task.

As can be derived from the chart all zero-shot approaches perform notably worse than the SOTA models, except for the emotion detection task. Moreover, prompting with gpt-3.5-turbo is the best zero-shot approach for all tasks except ABSA, where the NLI approach with DeBERTa reaches a 1% higher F1-score. The next best approach is usually prompting with Llama 2, followed by NLI and then MLM.

In Table 11, we present the score differences for all zero-shot approaches and model combinations compared to SOTA performance. After calculating these differences, we also averaged the scores per model making it possible to gauge the overall performance of each approach.

| **model** | **mean** | **median** | **max** | **min** |
|---|---|---|---|---|
| finetuned | 0 | 0 | 0 | 0 |
| gpt-3.5-turbo | -0.1933 | -0.205 | 0.06 | -0.4367 |
| llama2_13B_NL | -0.2978 | -0.2835 | -0.1300 | -0.5600 |
| llama2_13B_EN | -0.3613 | -0.3655 | -0.21 | -0.54 |
| llama2_7B_EN | -0.3830 | -0.3490 | -0.24 | -0.7 |
| mDeBERTa_NLI | -0.321 | -0.343 | -0.14 | -0.47 |
| bertje_NLI | -0.4154 | -0.398 | -0.24 | -0.59 |
| robbert_NLI | -0.4461 | -0.4283 | -0.26 | -0.71 |
| robbertje_NLI | -0.4608 | -0.38 | -0.25 | -0.73 |
| bertje_MLM | -0.4523 | -0.485 | -0.26 | -0.6967 |
| robbert_MLM | -0.4462 | -0.5023 | -0.17 | -0.61 |
| robbertje_MLM | -0.4479 | -0.4658 | -0.25 | -0.61 |
| generation | -0.3088 | -0.3050 | 0.06 | -0.7 |
| nli | -0.4108 | -0.368 | -0.14 | -0.73 |
| mlm | -0.4615 | -0.5125 | -0.17 | -0.6967 |

Table 11: Mean divergence for the individual model + approach (top) and averaged approach (bottom) combinations across all tasks

These results show that, across all tasks, prompting with LLMs reaches 30% lower scores compared to the SOTA, whereas NLI approaches reach 41% and MLM 46% lower scores. Although we are able to generalize the results for our different approaches, the large difference between min and max scores for each approach attest to the fact that the results vary significantly depending on the task, pre-trained model and especially the phrasing of the prompt. Even when using LLMsbbbb of the caliber of Llama 2, there are notable inconsistencies. In some cases, the smaller 7B model performs better for a task than the 13B model. Similarly, the baseline 13B model, which is not intended to be used for other languages sometimes (for Emotion Detection and ABSA) outperforms the 13B Llama model that has been adapted for Dutch. Whilst zero-shot approaches can provide solid baseline models, we do believe that it remains highly relevant to investigate a variety of models and approaches for specific tasks.

Finally, we should point out that our comparisons are limited to state-of-the-art models that are available. Therefore, our model selection shows a variety of model sizes. While the generation models consists of at least 7 billion parameters up to 175 billion for gpt-3.5-turbo, the NLI and MLM models are much smaller. The largest NLI model consists of 279 million parameters, the largest MLM model has 109 million. That is over a hundred times smaller than gpt-3.5-turbo's 175 billion parameters. Therefore it deserves mentioning that even relatively small NLI models such as

`mDeBERTa-v3-base-xnli-multilingual-nli-2mil7` (279M parameters) can achieve good results, often outperforming a Llama 2 7B Chat model and even the 13 billion variant in some tasks, such as (aspect-based) sentiment analysis.

## 7. Conclusion

In this paper, we have described and evaluated different approaches to zero-shot classification in a number of Dutch tasks. Firstly, we found that prompting with LLMs such as LLama 2 and gpt-3.5-turbo reach the highest zero-shot performance, followed by NLI and then MLM approaches. Secondly, zero-shot approaches can achieve results that are relatively close to the fine-tuned SOTA models for some tasks (such as sentiment analysis and emotion detection) whilst still leaving a large performance gap for other tasks, like Aspect-Based Sentiment Analysis.

Although this discrepancy can be partially attributed to the degree of explicitness of the task, we also believe the wording and phrasing of the prompt plays an important role. This indicates that prompt brittleness remains a major concern, especially when working on languages other than English. Overall, across all evaluated tasks, the mean performance gap is still a sizable 30% F1-score at best.

While the experimental results revealed that zero-shot can be a powerful method, it also comes with challenges and limitations. A primary limitation lies in the heavy reliance on prompt engineering. Our experiments illustrated that even slight modifications in prompts can significantly influence the outcomes, indicating a sensitivity to how questions are framed. This aspect underscores the challenge in achieving consistent results across different prompts and contexts. Even more so, prompts behave differently across models; where a prompt may work well for one model, it might not for another. Therefore we decided in this paper to use natural language prompts that an average user would use in conversation. However, this decision might have constrained the potential of each model. Intensive prompt engineering tailored to each specific model might lead to enhanced results, but this was beyond the scope of the research presented here.

Another critical challenge specific to generative models is ensuring that the model generates a valid label. To do so, we made use of the `outlines` library in our experiments. While this library ensures that generative models predict only valid labels by restricting output token probabilities, its effectiveness is closely tied to the quality of prompting and how well models adhere to those prompts. Specifically, models are often inclined to generate explanations or complete sentences instead of simple label responses. In practice, for example, when prompted whether a text is positive or negative and when the model is not constrained, it may generate an explanation or full sentence response such as "The text is positive because ..." instead of simply "positive". But `outlines` disallows the generation of "The" as the first token and only selects the highest probability for one of the (subtokens of) the valid labels by setting all other tokens' probabilities to negative infinity as a form of post-processing on the output probabilities. So even in a position where the model would normally generate "The" (first token), we are now discarding that that was its highest predicted token, and instead only consider the probability of the valid labels, no matter how small those probabilities might be – as we only wanted to know the highest probability. While gpt-3.5-turbo generally was less susceptible to this issue because it followed the instruction to only answer with this given set of labels, open-source models were typically "bmore chatty" and did not easily reply with only the labels. Therefore, using `outlines` is a necessity to ensure that valid labels are generated and to avoid a "Rest" category, but it can lead to potential poor performance when a broad prompt is given or when the models do not follow the prompt instructions of generating only labels correctly. Interestingly, this was most outspoken for the "emotion detection" benchmark where scores with `outlines` were quite low.

Finally, the rescaling of label probabilities in the MLM experiments was a method employed to mitigate prior biases of the model towards any of the labels. Such rescaling was not conducted in other methodologies, particularly generation models, potentially leaving them susceptible to similar biases. This inconsistency is a limitation of our current setup, primarily due to time constraints. We urge future work in this topic to achieve a more balanced and unbiased approach across different methodologies.

## Acknowledgements

We would like to thank the reviewers for their valuable insights. This work was supported by Ghent University under grant BOF.24Y.2021.0019.01., and by the Research Foundation Flanders under grant number I000921N-CLARIAH and FWO.OPR.2020.0014.01.

## References

Bagga, Amit and Breck Baldwin (1998), Algorithms for scoring coreference chains, *The first international conference on language resources and evaluation workshop on linguistics coreference*, Vol. 1, Citeseer, pp. 563–566.

Barbieri, Francesco, Luis Espinosa Anke, and Jose Camacho-Collados (2022), XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, European Language Resources Association, Marseille, France, pp. 258–266. https://aclanthology.org/2022.lrec-1.27.

Bowman, Samuel R., Gabor Angeli, Christopher Potts, and Christopher D. Manning (2015), A large annotated corpus for learning natural language inference, *in* Màrquez, Lluís, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Lisbon, Portugal, pp. 632–642. https://aclanthology.org/D15-1075.

Brown, Penelope and Steven C Levinson (1987), *Politeness: Some Universals in Language Usage*, Politeness: Some Universals in Language Usage, Cambridge University Press. https://books.google.be/books?id=OG7W8yA2XjcC.

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei (2020), Language models are few-shot learners, *in* Larochelle, H., M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., pp. 1877–1901. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Bryant, Gregory A and Jean E Fox Tree (2002), Recognizing verbal irony in spontaneous speech, *Metaphor and symbol* **17** (2), pp. 99–119, Taylor & Francis.

Clift, Rebecca (1999), Irony in conversation, *Language in society* **28** (4), pp. 523–553, Cambridge University Press.

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov (2020),

Unsupervised cross-lingual representation learning at scale, *in* Jurafsky, Dan, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, pp. 8440–8451. https://aclanthology.org/2020.acl-main.747.

Cui, Ganqu, Shengding Hu, Ning Ding, Longtao Huang, and Zhiyuan Liu (2022), Prototypical verbalizer for prompt-based few-shot tuning, *in* Muresan, Smaranda, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Dublin, Ireland, pp. 7014–7024. https://aclanthology.org/2022.acl-long.483.

De Bruyne, Luna, Orphée De Clercq, and Veronique Hoste (2021), Prospects for Dutch emotion detection: Insights from the new emotionl dataset, *Computational Linguistics in the Netherlands Journal* **11**, pp. 231–255.

De Clercq, Orphée, Luna De Bruyne, and Véronique Hoste (2020), News topic classification as a first step towards diverse news recommendation, *Computational Linguistics in the Netherlands Journal* **10**, pp. 37–55.

De Geyndt, Ellen, Orphée De Clercq, Cynthia Van Hee, Els Lefever, Pranaydeep Singh, Olivier Parent, and Veronique Hoste (2022), SentEMO : a multilingual adaptive platform for aspect-based sentiment and emotion analysis, *in* Barnes, Jeremy and De Clercq, Orphée and Barriere, Valentin and Tafreshi, Shabnam and Alqahtani, Sawsan and Sedoc, João and Klinger, Roman and Balahur, Alexandra, editor, *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, Association for Computational Linguistics (ACL), pp. 51–61. http://doi.org/10.18653/v1/2022.wassa-1.5.

De Langhe, Loic, Orphée De Clercq, and Veronique Hoste (2022), Constructing a cross-document event coreference corpus for dutch, *Language Resources and Evaluation* pp. 1–30, Springer.

De Langhe, Loic, Orphee De Clercq, and Veronique Hoste (2023), Filling in the gaps: Efficient event coreference resolution using graph autoencoder networks, *Proceedings of The Sixth Workshop on Computational Models of Reference, Anaphora and Coreference (CRAC 2023)*, pp. 1–7.

de Vries, Wietse, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim (2019), BERTje: A Dutch BERT Model. http://arxiv.org/abs/1912.09582.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2020), RobBERT: A Dutch RoBERTa-based Language Model, *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, Online, pp. 3255–3265.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2021), Robbertje: A distilled dutch bert model, *Computational Linguistics in the Netherlands Journal* **11**, pp. 125–140. https://www.clinjournal.org/clinj/article/view/131.

Dettmers, Tim, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer (2023), Qlora: Efficient finetuning of quantized llms.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019), BERT: Pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171–4186. https://aclanthology.org/N19-1423.

Fei, Yu, Zhao Meng, Ping Nie, Roger Wattenhofer, and Mrinmaya Sachan (2022), Beyond prompting: Making pre-trained language models better zero-shot learners by clustering representations, *in* Goldberg, Yoav, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, pp. 8560–8579. https://aclanthology.org/2022.emnlp-main.587.

Gao, Tianyu, Adam Fisch, and Danqi Chen (2021), Making pre-trained language models better few-shot learners, *in* Zong, Chengqing, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, Online, pp. 3816–3830. https://aclanthology.org/2021.acl-long.295.

Gibbs, Raymond W (1986), On the psycholinguistics of sarcasm., *Journal of experimental psychology: general* **115** (1), pp. 3, American Psychological Association.

Halder, Kishaloy, Alan Akbik, Josip Krapac, and Roland Vollgraf (2020), Task-aware representation of sentences for generic text classification, *Proceedings of the 28th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Barcelona, Spain (Online), pp. 3202–3213. https://aclanthology.org/2020.coling-main.285.

He, Pengcheng, Jianfeng Gao, and Weizhu Chen (2021), Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, *arXiv preprint arXiv:2111.09543*.

Kipf, Thomas N and Max Welling (2016), Variational graph auto-encoders, *NIPS Workshop on Bayesian Deep Learning*.

Kowsari, Kamran, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura Barnes, and Donald Brown (2019), Text classification algorithms: A survey, *Information*. https://www.mdpi.com/2078-2489/10/4/150.

Liu, Bing (2015), *Sentiment analysis: mining opinions, sentiments, and emotions*, 1st ed., New York: Cambridge University Press.

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig (2023), Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3560815.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov (2019), Roberta: A robustly optimized bert pretraining approach, *ArXiv*. https://api.semanticscholar.org/CorpusID:198953378.

Lu, Jing and Vincent Ng (2018), Event coreference resolution: A survey of two decades of research, *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, AAAI Press, p. 5479–5486.

Luo, Xiaoqiang (2005), On coreference resolution performance metrics, *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 25–32.

Maladry, Aaron, Els Lefever, Cynthia Van Hee, and Véronique Hoste (2023), The limitations of irony detection in dutch social media, *Language Resources and Evaluation* pp. 1–32, Springer.

Marelli, Marco, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli (2014), Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment, *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pp. 1–8.

Min, Bonan, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth (2023), Recent advances in natural language processing via large pre-trained language models: A survey, *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3605943.

Mohammad, Saif (2016), Sentiment analysis: Detecting valence, emotions, and other affectual states from text, *in* Meiselman, Herbert L., editor, *Emotion measurement*, Woodhead Publishing, pp. 201–237.

Plaza-del Arco, Flor Miriam, María-Teresa Martín-Valdivia, and Roman Klinger (2022), Natural language inference prompts for zero-shot emotion classification in text across corpora, *Proceedings of the 29th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Gyeongju, Republic of Korea, pp. 6805–6817. https://aclanthology.org/2022.coling-1.592.

Qin, Guanghui and Jason Eisner (2021), Learning how to ask: Querying LMs with mixtures of soft prompts, *in* Toutanova, Kristina, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Online, pp. 5203–5212. https://aclanthology.org/2021.naacl-main.410.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019), Language models are unsupervised multitask learners, *OpenAI blog* **1** (8), pp. 9.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu (2020), Exploring the limits of transfer learning with a unified text-to-text transformer, *Journal of Machine Learning Research* **21** (140), pp. 1–67. http://jmlr.org/papers/v21/20-074.html.

Schick, Timo and Hinrich Schütze (2021), Exploiting cloze-questions for few-shot text classification and natural language inference, *in* Merlo, Paola, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, Association for Computational Linguistics, Online, pp. 255–269. https://aclanthology.org/2021.eacl-main.20.

Sulis, Emilio, Delia Irazú Hernández Farías, Paolo Rosso, Viviana Patti, and Giancarlo Ruffo (2016), Figurative messages and affect in twitter: Differences between# irony,# sarcasm and# not, *Knowledge-Based Systems* **108**, pp. 132–143, Elsevier.

van der Burgh, Benjamin and Suzan Verberne (2019), The merits of Universal Language Model Fine-tuning for Small Datasets – a case with Dutch book reviews.

Van Hee, Cynthia, Els Lefever, and Véronique Hoste (2016), Exploring the realization of irony in twitter data, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 1794–1799.

Van Hee, Cynthia, Els Lefever, and Véronique Hoste (2018), SemEval-2018 task 3: Irony detection in English tweets, *in* Apidianaki, Marianna, Saif M. Mohammad, Jonathan May, Ekaterina Shutova, Steven Bethard, and Marine Carpuat, editors, *Proceedings of the 12th International*

*Workshop on Semantic Evaluation*, Association for Computational Linguistics, New Orleans, Louisiana, pp. 39–50. https://aclanthology.org/S18-1005.

Vanroy, Bram (2023), Language resources for Dutch large language modelling, *arXiv preprint arXiv:2312.12852*.

Vilain, Marc, John D Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman (1995), A model-theoretic coreference scoring scheme, *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995*.

Wijnholds, Gijs and Michael Moortgat (2021), Sick-nl: A dataset for dutch natural language inference, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, Online. https://www.aclweb.org/anthology/2021.eacl-main.126/.

Willard, Brandon T and Rémi Louf (2023), Efficient guided generation for llms, *arXiv preprint arXiv:2307.09702*.

Wilson, Deirdre and Dan Sperber (2012), Explaining irony, *Meaning and relevance* pp. 123–145, Cambridge University Press Cambridge.

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. (2019), Huggingface's transformers: State-of-the-art natural language processing, *arXiv preprint arXiv:1910.03771*.

Yao, Yao, Zuchao Li, and Hai Zhao (2023), Learning event-aware measures for event coreference resolution, *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13542–13556.

Yin, Wenpeng, Jamaal Hay, and Dan Roth (2019), Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, Hong Kong, China, pp. 3914–3923. https://aclanthology.org/D19-1404.

Zhang, Tianyi and Tatsunori Hashimoto (2021), On the inductive bias of masked language modeling: From statistical to syntactic dependencies, *arXiv preprint arXiv:2104.05694*.

Zhang, Wenxuan, Xin Li, Yang Deng, Lidong Bing, and Wai Lam (2022), A survey on aspect-based sentiment analysis: Tasks, methods, and challenges, *IEEE Transactions on Knowledge and Data Engineering* pp. 1–20.

## Appendix A. Prompt Templates

### A.0 General

**MLM** The task-specific prompts are shown in the succeeding sections. Note that the sentence with the mask is given before the instance to be classified. The reason is that some target instances are so long that they need to be truncated. Placing the `[MASK]` near the start ensures that it will not be cut off. Secondly, note that depending on the model's tokenizer a space is added before the mask. The reason for this is that BPE tokenizers of RobBERT-like models will have a prefix added to the token when it is preceded by a space (e.g. `Gpositive`), in which case no space should be used before the mask token. This is not the case for WordPiece tokenizers, as used by `GroNLP/bert-base-dutch-cased`, in which case we do add a space before the mask token. Note finally that the mask token differs across models (typically `[MASK]` or `<mask>`).

**Generative LLMs** For the original English Llama 2, we use the following, default system message:

> You are a helpful, respectful and honest assistant. Always answer as helpfully as possible, while being safe. Your answers should not include any harmful, unethical, racist, sexist, toxic, dangerous, or illegal content. Please ensure that your responses are socially unbiased and positive in nature.
>
> If a question does not make any sense, or is not factually coherent, explain why instead of answering something not correct. If you don't know the answer to a question, please don't share false information.

For the Dutch variant we use the following system message, which is the translation of the English system message. It was also used when finetuning the Dutch Llama 2 model.

> Je bent een behulpzame, respectvolle en eerlijke assistent. Antwoord altijd zo behulpzaam mogelijk. Je antwoorden mogen geen schadelijke, onethische, racistische, seksistische, gevaarlijke of illegale inhoud bevatten. Zorg ervoor dat je antwoorden sociaal onbevooroordeeld en positief van aard zijn.
>
> Als een vraag nergens op slaat of feitelijk niet coherent is, leg dan uit waarom in plaats van iets niet correct te antwoorden. Als je het antwoord op een vraag niet weet, deel dan geen onjuiste informatie.

### A.1 Sentiment Analysis

**NLI**

```
Deze recensie is {positief|negatief}.
```

**MLM**

```
De volgende recensie is[MASK].
{review}
```

**Generative LLMs**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

Is het sentiment in de volgende Nederlandstalige boekrecensie
positief of negatief?

{review} [/INST] """
```

**gpt-3.5-turbo**

```
Is het sentiment in de volgende Nederlandstalige boekrecensie
positief of negatief?

{review}
```

### A.2 Aspect-based Sentiment Analysis

**NLI**

```
{text} het sentiment dat hoort bij {aspect} + in deze review is
{zeer negatief | negatief | neutraal | positief | zeer positief}
```

**MLM**

```
{text} het sentiment dat hoort bij {aspect} + in deze review is
{vreselijk | negatief | neutraal | positief | geweldig}
```

**Llama 2**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

{text} het sentiment dat hoort bij {aspect} + in deze review is
{zeer negatief | negatief | neutraal | positief | zeer positief}
[/INST] """
```

**gpt-3.5-turbo**

```
{text} het sentiment dat hoort bij {aspect} + in deze review is
{zeer negatief | negatief | neutraal | positief | zeer positief}
```

### A.3 Emotion Detection

**NLI**

```
De emotie die in deze tekst wordt uitgedrukt is
{neutraal|woede|angst|vreugde|liefde|verdriet}.
```

**MLM**

```
De emotie die in deze tekst wordt uitgedrukt is[MASK].
{text}
```

**Llama 2**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

Welke emotie wordt in deze tekst uitgedrukt? Antwoord met één
van deze zes emoties: "neutraal", "woede", "angst", "vreugde",
"liefde" of "verdriet". Tekst: {text} [/INST] """
```

**gpt-3.5-turbo**

```
Welke emotie wordt in deze tekst uitgedrukt? Antwoord met één
van deze zes emoties: "neutraal", "woede", "angst", "vreugde",
"liefde" of "verdriet". Tekst: {text}
```

### A.4 Irony Detection

**NLI**

```
{text}
Deze tweet is {ironisch|oprecht}.
```

**MLM**

```
Deze tekst is[MASK].
{text}
```

**Llama 2**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

Geef voor deze tweet aan of de tekst oprecht is of
ironisch/sarcastisch bedoeld is: {text} [/INST] """
```

**gpt-3.5-turbo**

```
Geef voor deze tweet aan of de tekst oprecht is of
ironisch/sarcastisch bedoeld is: {text}
```

### A.5 News Topic Classification

**NLI**

```
Deze tekst gaat over {kunst, cultuur, entertainment
en media|conflicten, oorlog en vrede|criminilatieit,
de wet en justitie|rampen en ongelukken|economie
en financiën|onderwijs|milieu|gezondheid|human
interest|werk|lifestyle en ontspanning|politiek|religie|wetenschap
en technologie|de maatschappij|sport|weer}.
```

**MLM**

```
Deze tekst gaat over [MASK].
{text}
```

**Llama 2**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

Over wat gaat deze tekst? Kies een of meerdere categorieën
uit de volgende lijst: "kunst, cultuur, entertainment en
media", "conflicten, oorlog en vrede", "criminaliteit, de wet
en justitie", "rampen en ongelukken", "economie en financiën",
"onderwijs", "milieu", "gezondheid", "human interest", "werk",
"lifestyle en ontspanning", "politiek", "religie", "wetenschap en
technologie", "de maatschappij", "sport", "weer". Tekst: {text}
[/INST] """
```

**gpt-3.5-turbo**

```
Over wat gaat deze tekst? Kies een of meerdere categorieën
uit de volgende lijst: "kunst, cultuur, entertainment en
media", "conflicten, oorlog en vrede", "criminaliteit, de wet
en justitie", "rampen en ongelukken", "economie en financiën",
"onderwijs", "milieu", "gezondheid", "human interest", "werk",
"lifestyle en ontspanning", "politiek", "religie", "wetenschap en
technologie", "de maatschappij", "sport", "weer". Tekst: {text}
```

### A.6 Event Coreference Resolution

**NLI**

```
Deze gebeurtenissen verwijzen {naar dezelfde gebeurtenis|naar
verschillende gebeurtenissen}.
```

**MLM**

```
Verwijzen deze gebeurtenissen naar dezelfde gebeurtenis? [MASK].
{text}
```

**Llama 2**

```
"""[INST] <<SYS>>
{system_message}
<</SYS>>

Antwoord enkel met ja of nee. Verwijzen deze gebeurtenissen
naar dezelfde gebeurtenis? Twe gebeurtenissen verwijzen naar
elkaar wanneer ze op hetzelfde moment gebeuren, op dezelfde
plaats gebeuren en wanneer dezelfde personen of objecten er een
rol in spelen. Tekst: {text} [/INST] """
```

**gpt-3.5-turbo**

```
Antwoord enkel met ja of nee. Verwijzen deze gebeurtenissen naar
dezelfde gebeurtenis? Twe gebeurtenissen verwijzen naar elkaar
wanneer ze op hetzelfde moment gebeuren, op dezelfde plaats
gebeuren en wanneer dezelfde personen of objecten er een rol
in spelen. Tekst: {text}
```