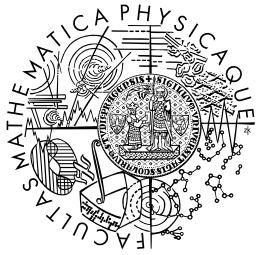

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Anna Kuznetsova

Multilingual Multimodal Detection of Humour in Stand-Up Comedy

Institute of Formal and Applied Linguistics

Supervisor of the master thesis: RNDr. Martin Holub, Ph.D.

Study programme: Computer Science

Study branch: Language Technologies and Computational Linguistics

Prague 2023

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ....

Author's signature

I would like to express my heartfelt gratitude to my family and friends for their incredible support throughout this challenging journey. Even though we have been physically apart and navigating through very uncertain times, their encouragement and understanding have been a constant source of strength.

I am also deeply thankful to the Erasmus+ Programme for providing me with the scholarship that made it possible for me to participate in this program. This experience has played a pivotal role in shaping my academic and personal growth.

Lastly, I would like to express my gratitude to Fondazione Bruno Kessler for providing me with a valuable internship experience. I would also like to thank my supervisor, Carlo Strapparava from the University of Trento, for his invaluable guidance and support.

Title: Multilingual Multimodal Detection of Humour in Stand-Up Comedy

Author: Anna Kuznetsova

Department: Institute of Formal and Applied Linguistics

Supervisor: RNDr. Martin Holub, Ph.D., Institute of Formal and Applied Linguistics

Abstract: This thesis focuses on the multimodal and multilingual detection of humor in stand-up comedy videos. A novel multilingual dataset was collected, primarily targeting the Russian language, to address the lack of specific multimodal datasets for humor detection in this language. The dataset was obtained from stand-up comedy videos with subtitles sourced from YouTube. The thesis investigates various aspects of the data preparation process, including word-level forced alignment, segmentation, and labeling with laughter detection. Two automatic laughter detection approaches are explored: the peak detection approach, which employs preprocessed voiceless audio and an energy-based peak detection algorithm with clusterization filtering, and the machine learning approach, which utilizes a pretrained model to detect laughter presence and duration. Results indicate that for now the machine learning approach outperforms the peak detection approach in terms of accuracy and generalization, however the peak detection approach is considered promising. Additionally, thesis delves into the unimodal textual and multimodal humor detection on the new dataset. The results demonstrate the ability of neural models to capture humour in both languages even in the textual only setting. While multimodal experiments showed that even in simple models the addition of visual modality improves the results. However, further experiments and research are needed to enhance the laughter detection labeling quality and investigate the influence of different modalities in the multimodal and multilingual approach.

Keywords: humour automatic detection stand-up comedy multilingual multimodal

Contents

Introduction

Humour is an integral part of our lives and plays a crucial role in our communication. People use humour for various reasons: to persuade others, make themselves more likeable or credible, lighten up serious topics, express their identity, or even empower or marginalize others. It's a cognitive process that can be expressed in many ways, like laughter, smiles, nods, and other vocalizations. Funny messages can take different forms, such as verbal storytelling, text, funny pictures, pantomimes, or just facial expressions. The topic of humour is vast and has many unanswered questions. That's why researchers from different fields have tried to explain its origins, functions, and how we process and perceive it.

In this research, my objective is to contribute to the existing body of research on multilingual multimodal humour detection through two main aspects. Firstly, I aim to collect a novel multilingual dataset comprising Russian and English stand-up routines. While numerous datasets in the field are based on sitcoms, stand-up routines have been relatively underrepresented. Considering other mediums in Russian posed a challenge as suitable sources such as Russian sitcoms with subtitles or transcripts were scarce. Additionally, another source of multimodal humour datasets - official TED Talks in Russian were not readily available, with only TEDx event videos accessible, which are not as well transcribed. Fortunately, the Russian stand-up scene is thriving, and many comedy clubs upload videos to platforms like YouTube, although with limited transcriptions. In the end, I managed to gather a sufficient number of videos for the dataset. Secondly, I aim to explore different approaches to humour annotation using laughter. By experimenting with these techniques, I hope to reduce the need for manual annotation. Lastly, I want to determine if the collected dataset is suitable for multimodal humour detection and investigate if including additional modalities improves the performance. I won't be focusing on audio in my experiments as I'll be using it primarily for laughter detection. I believe that the visual modality deserves more attention and examination.

The structure of this work is as follows: In Chapter 1, the background of multimodal humour detection is presented, highlighting the significance of humour detection and the role of multiple modalities in this task. The chapter also discusses existing datasets and models in the field. Chapter 2 focuses on the new datasets. It describes the collection process of the datasets, the annotation techniques employed, and the steps involved in forced alignment, segmentation, and laughter detection. Different approaches for laughter detection, including the peak detection approach and machine learning approach, are explored in this chapter. Chapter 3 delves into the models used for humour detection. It includes text-based models, as well as multimodal models that combine text, facial, and visual features. The chapter provides an overview of the models and their implementation in the study. Chapter 4 presents the results of the experiments conducted on Russian, English, and multilingual datasets. It includes the evaluation of the models' performance and the analysis of the obtained results. In Chapter 5, the discussions revolve around various topics related to the research findings. This includes the quality of labelling, different laughter detection approaches, advanced multimodal modelling techniques, multilinguality, and the

ethical considerations surrounding humour detection.

The list of videos used for the dataset is available in Appendix A.1 and A.2. Appendix B.1 describes the dataset structure and auxiliary files available, Appendix B.2 addresses the scripts that were used during this research.

1. Background

1.1 Multimodal humour

Humour is a very complex and widespread phenomenon. It is still challenging to come up with a definitive definition of humour [Attardo, 2010]. Nonetheless, there are three main theories of humour: the superiority theory, the relief theory, and the incongruity theory [Mulder and Nijholt, 2002]. According to the superiority theory, people laugh at the failures or shortcomings of others as a way to assert their superiority. It's an old theory that still holds true in many cases. The relief theory, influenced by psychology and Freud's ideas, suggests that laughter helps release build-up tension resulting from suppressed taboo thoughts. However, it doesn't fully explain why something is considered funny. The incongruity theory states that we find humour in the mismatch between concepts and reality. Jokes often involve presenting two objects as belonging to the same concept, only to reveal that one of them doesn't really fit. The difference between the objects then becomes the source of our amusement.

Another interesting aspect to explore is non-verbal or paraverbal humour. It is easy to imagine how the performance of a joke can influence the audience's reaction, so when analyzing humour, we need to consider not only the language but also visual and auditory cues. Actions, gestures, and delivery style all play a significant role in how a joke is perceived by the audience [Norrick, 2004]. Irony and sarcasm, for example, rely heavily on non-verbal cues to enhance their effects, even though the primary indicator of irony is the incongruity with the context.

In a study conducted by Attardo et al. [2003], 41 ironic utterances from American situation comedies (sitcoms) aired in 1999 were collected and analyzed. The researchers focused on phonological cues extracted from recordings, while facial cues were classified by a small group of participants under two conditions. Although they did not find a specific "ironical intonation", they did identify some intonation-based ironic cues. Additionally, through the visual modality, a "blank face" facial expression was identified in the second part of the study.

González-Fuente et al. [2015] investigated "gestural codas," which are cues produced after the utterance, in spontaneous speech in the Catalan language. Their interest was in the timing of these cues relative to the ironic utterance and their influence on irony detection. They collected 47 ironic utterances from friendly dialogues in an experimental setting. The annotation revealed that there is no specific intonation for irony; instead, speakers employ a variety of prosodic modulations. Regarding gestures, they can occur both during and after ironic utterances. The main cues observed were "Smile/Laughter" during the production of the utterance and in post-utterance codas, as well as deviations in gaze, including unfocused gaze and gaze directed at the interlocutor. Interestingly, in non-ironic utterances, head nodding was the most common visual cue, while in ironic utterances, shaking, tilting, and mouth stretching was more prevalent, particularly after the ironic sentence. Another notable cue in ironic codas was eyebrow raising. Overall, gestures were more frequent in ironic utterance codas compared to non-ironic ones.

In another experimental study by Deliens et al. [2018], the perception of irony

was explored. In the first experiment, participants performed a categorization task on specific modalities, either audio or video alone. The results confirmed that ironic prosody and facial expressions can be accurately discriminated within their respective modalities. In the second and third experiments, the researchers analyzed the effect of non-verbal cues compared to the impact of contextual incongruity using eye-tracking data and reaction time measurements. The data indicated that non-contextual cues were less reliable but enhanced processing speed.

In another classification task conducted by Aguert [2022], three conditions were examined: prosody only, facial expressions only, and both prosody and facial expressions. The results revealed specific vocal cues associated with irony, such as increased pauses, the use of filler vocalizations (e.g., "uh", "um"), a slower speech rate, and lower intensity. Facial cues observed included reduced smiles, downward gaze, eye-rolling, expressive movements in the mouth area, and eyebrow flashes.

Lastly, Ellis [2022] compared the detection of irony in computer-mediated communication and face-to-face interactions. The findings suggest that even in mediums lacking non-verbal cues, there are medium-specific cues that help in irony detection. For computer-mediated communication, these cues can include typical or atypical punctuation, capitalization, emoticons, emojis, and hashtags.

It is worth noting that situational comedies serve as a popular data source for humour and irony research. Although sitcoms are scripted, the genuine audience reactions in shows filmed in front of a live audience contribute to a natural perception of humour. Furthermore, sitcoms often exhibit various forms of humour. Hasyim and Hanidar [2022] conducted an annotation study on the popular TV series "The Office" (US), focusing on different forms of verbal irony. According to Gibbs [2000] categorization, all five types of verbal irony (sarcasm, jocularity, rhetorical question, hyperbole, and understatement) were identified in the show.

1.2 Humour detection

Due to the complex and universal nature of humour, automatic detection of humour poses significant challenges in natural language processing (NLP). However, it is an important task as people interact with robots and smart assistants more each year, and successful interactions often depend on understanding and producing humour from both sides. As a result, humour recognition has become a promising field of research, with extensive work conducted over the past decades.

In their comprehensive literature review on computational humour recognition, Kalloniatis and Adamidis [2023] systematized datasets, features, and algorithms. Firstly, they classified four types of datasets based on their content: short text, long text, multimodal (discussed in the next section), and dialogues. Short text datasets are typically sourced from social media platforms like Twitter or collections of puns and one-liners, as well as news headlines. Long texts are collected from news articles, web comments, essays, narrative jokes, and dialogues taken from computer-mediated communication platforms, such as messenger chats, as well as product and community-based question-answering platforms.

While collecting data is the initial step, the subsequent step of data annotation is even more critical. Humour is subjective, so annotating whether something is funny is not a straightforward task. The most common annotation strategy

is manual annotation by domain experts. However, manual annotation is time-consuming and expensive. Furthermore, the inter-annotator agreement can vary depending on the annotators' personalities and familiarity with the humour domain. Another approach is distant supervision, where training data is generated using existing domain knowledge. For example, on Twitter, users can employ hashtags to indicate humour, and labelling is done by identifying messages with relevant hashtags. Although this approach is faster, it often leads to noisy data as the appropriate use of domain labels cannot be guaranteed. Consequently, it is often followed by crowd-sourcing, which involves distributed manual annotation by multiple users.

Another aspect covered in the review is feature engineering. Various types of features can be extracted from the data before passing it to the classifier. Semantic features play a significant role in data preprocessing. Based on knowledge of how humour works, features such as ambiguity (the number of meanings a word or phrase has), incongruity (mixing inconsistent frames), emotion-based features (sentiment, mood), unexpectedness, subjectivity, and negation can be extracted. The next step involves automated features, which involve numerical representations of the data. Conventional NLP features are typically used, such as word embeddings, bag-of-words, and part-of-speech tags. Additionally, features can also be extracted from lexical resources, such as dictionaries and databases like WordNet. These features include alliteration, antonymy, polarity, the presence of adult slang and profanities, and rhyming.

The choice of the classification model is often influenced by advancements in NLP. Rule-based approaches were initially employed and demonstrated satisfactory performance on appropriate datasets. Supervised learning models like Support Vector Machine (SVM), Naive Bayes, and Random Forest are commonly used as baseline models. The next step involves deep learning approaches, which have shown impressive results. Various models have been utilized, including long short-term memory network (LSTM) [Bertero and Fung, 2016b], convolutional neural network (CNN) [Chen and Lee, 2017], and recurrent neural network (RNN) [Zhang et al., 2019]. Models capable of capturing temporal dependencies, such as LSTM and RNN, are particularly suitable for humour classification, given its context dependence. Transformers, another family of models capable of capturing context dependence, have also been utilized. The best results are typically achieved with large pretrained models fine-tuned specifically for humour detection tasks [Peyrard et al., 2021]. Additionally, custom modifications have been explored, such as modelling congruity and other relationships between the context and target sentence [Annamoradnejad and Zoghi, 2022].

It is important to note that while most research focuses on English data, there are also datasets available for other languages. For example, Russian datasets have been developed by researchers such as Ermilov et al. [2018], Blinov et al. [2019], and Baranova-Bolotova et al. [2019]. Italian datasets have been collected by Boccignone et al. [2017] and Buscaldi and Rosso [2007], and there are even multilingual datasets available, such as English, Spanish, French and Italian dataset by Cignarella et al. [2020].

1.3 Multimodal humour detection

The recognition of multimodal aspects in humour has gained significant attention in recent years. Multimodality has become a popular topic within the machine learning community, driven by advancements in both image and text processing. Combining these modalities has led to the development of various models and tasks, such as visual question answering, visual reasoning, and image generation from text prompts. Humour detection has also benefited from the interest in multimodality, resulting in the proposal of numerous datasets and models for multimodal humour detection.

1.3.1 Datasets

Dataset	Language	Source	Annotation
MUStARD [Castro et al., 2019]	English	Video clips from Friends, The Golden Girls, Sarcasmaholics Anonymous. Seasons 1-8 of TBBT.	Laughter detection software [Ryokai et al., 2018] and manual annotation by two annotators.
UR-FUNNY [Hasan et al., 2019]	English	TED talks	'Laughter' marker in the transcript
TBBT [Kayatani et al., 2021]	English	10 seasons of TBBT	Laughter extraction from the audio - channel subtraction, Hilbert transform, low-pass filter, manual filtering of music.
MHD [Patro et al., 2021]	English	TBBT	Manual annotation of laughter
Open Mic [Mittal et al., 2021]	English	Stand-up (humourous), TED Talks (non-humourous)	Manual segmentation, laughter detection software [Gillick et al., 2021]
Passau-SFCH [Christ et al., 2022]	German	Press conferences of professional football coaches from the German Bundesliga	Manual transcription and annotation on sentiment and direction (9 annotators)
SHEMuD Chauhan et al. [2022]	Hindi, English	TV Series "Shrimaan Shrimati Phir Se"	Manual English translation and sentiment annotation

Table 1.1: Multimodal humour datasets presented in the literature.

Many studies on multimodal humour detection have utilized datasets sourced from situational comedies like Friends, The Big Bang Theory (TBBT), and Seinfeld [Castro et al., 2019, Kayatani et al., 2021, Patro et al., 2021]. Despite the fact that sitcoms are scripted, most of them are filmed in front of a live audience

and the laughter track is usually a genuine reaction of that audience, and as we discussed laughter is one of the indicators of humour. I collected and described the most notable available datasets in Table 1.1.

The MUStARD (Multimodal Sarcasm Detection) dataset, introduced by Castro et al. [2019], was collected using video clips from popular sitcoms like Friends, The Golden Girls, Sarcasmaholics Anonymous, and seasons 1-8 of The Big Bang Theory. To segment The Big Bang Theory part, the authors utilized laughter detection software [Ryokai et al., 2018]. Manual annotation of sarcasm was performed by two annotators, with an inter-annotator agreement (Kappa score) of 0.2326 for The Big Bang Theory segment and 0.5877 for the remaining portion. In cases where videos lacked transcription, manual transcriptions were created.

The Big Bang Theory (TBBT) dataset from Kayatani et al. [2021] was collected from 10 seasons of the sitcom. The authors associated character-annotated transcripts with timed subtitles and performed automatic labelling using laughter extraction. By subtracting the left and right channels of the source audio track, which cancels out character utterances typically centred in the audio, an audio track with a merged laugh track and music track was obtained. Noise reduction and other transformations were applied, and the resulting candidate segments were manually filtered to exclude non-laughter segments.

Patro et al. [2021] proposed the Multimodal Humour dataset (MHD) also based on episodes of The Big Bang Theory. This dataset was manually annotated for humour, utilizing laughter as a marker. It also includes additional information such as scene descriptions, speakers, recipients, participants, and dialogue turns.

TED talks, although not primarily intended to be humourous, often incorporate humour as a persuasive tool. Hasan et al. [2019] introduced the UR-FUNNY dataset, which is based on TED talks. The highly reliable transcriptions of the talks include audience markers like "laughter," which were used to label the utterances. However, official transcripts are typically only segmented into paragraphs, so forced alignment techniques were employed to assign beginning and end times for sentences and words.

Stand-up comedy has indeed been recognized as a valuable source for creating humour detection datasets. In their work, Mittal et al. [2021] collected stand-up shows from the web, with each show lasting approximately one hour. These shows were manually segmented into smaller clips of around 2 minutes in length, ensuring that each clip contained the full context of a joke. Transcripts of the stand-up performances were also obtained from the web. To assess the humour content in the clips, the authors used a different scoring system compared to binary classification. They introduced a "humour quotient" rating based on the audience laughter. A laughter detection model [Gillick et al., 2021] was employed to measure the intensity and time intervals of the audience laughter in each clip. The duration of all the laughter intervals in a clip was then summed and divided by the clip duration to obtain the final humour quotient score. Additionally, human annotation was performed using a Likert-scale of 5 points, based on the laughter feedback. The average pairwise Cohen's Kappa for the human annotation was reported to be 0.634.

In recent years, there have also been efforts to create datasets that capture spontaneous humour production and perception. Christ et al. [2022] introduced the Passau-Spontaneous Football Coach Humour (Passau-SFCH) dataset, which

consists of pre-match press conference videos featuring football coaches from the Bundesliga (the national premier soccer league). The dataset was manually annotated for the presence of humour and its dimensions, such as sentiment and direction, using Martin's Humour Style Questionnaire. It should be noted that most of the discussed multimodal datasets are primarily available in English, except for *Sentiment, Humor, and Emotion aware Multilingual Multimodal Multiparty Dataset* (SHEMuD) based on Hindi and English by Chauhan et al. [2022], as well as the Passau-SFCH dataset, which is in German.

1.3.2 Models

Model	Features	Architecture
SVM [Castro et al., 2019]	T - BERT, A - MFCC, melspectogram, etc., V - ResNet-152	SVM. Early fusion - feature concatenation. Scale features for speaker dependent.
[Kayatani et al., 2021]	T - BERT, V - OpenFace, C - character multi-hot embedding	BERT and LSTM for punchline modeling (with character encoding). FC for visual and character features. Modality Attention with weights from softmax of MLP.
HKT [Hasan et al., 2021]	T - ALBERT, A - MFCCs, fundamental frequency, V - OpenFace, H - linguistic sentiment and ambiguity	Transformer encoders, Bimodal Cross Attention Layer with Multimodal Fusion
FunnyNet [Liu et al., 2022]	T - BERT and LSTM, A - Mel spectrogram and BYOL-A, V - TimeSformer, F - InceptionResNet and LSTM	Projection module from linear layers, Cross-Attention Fusion - 3 cross-attention and 1 self-attention module. Self-Supervised Contrastive Loss.

Table 1.2: Humour detection models architectures and features. The following modalities are considered: T - text, A - audio, V - visual, C - character, H - humour-centric, F - facial.

In many multimodal humour detection studies, baseline models such as SVM [Castro et al., 2019] or Conditional Random Field (CRF) are often used with extracted features from different modalities. For instance, in the work of Bertero and Fung [2016a], CRF, RNN, and CNN were compared using word-level language features (e.g., n-grams, parts of speech, sentence lengths, word senti-

ment) and acoustic features (e.g., Mel Frequency Cepstral Coefficients (MFCCs), pitch, intensity). The CNN architecture achieved the best performance with an F-score of 68.5%. More recently, with the introduction of attention mechanisms and transformers [Vaswani et al., 2017], new models based on multimodal transformers have been proposed. I collected the most recent models architecture in Table 1.2 and compared their best F-scores in Table 1.3.

Model	Modalities	MUStARD	TBBT	UR-FUNNY
SVM [Castro et al., 2019]	T+V (s-dependent)	71.6		
	T+A (s-independent)	63.1		
[Kayatani et al., 2021]	T+C		65.07	
	T+V+C+Attention		65.03	
HKT [Hasan et al., 2021]	T+A+V+H	79.25		77.36
FunnyNet [Liu et al., 2022]	V+F+A	81.4	69.6	83.7
	V+F+T	75.2	76.0	82.3

Table 1.3: Best F-scores of the recent humour detection models on different datasets.

Hasan et al. [2021] introduced the Humour Knowledge enriched Transformer (HKT), which encoded language, audio, vision, and humour-centric features using transformer -based encoders. They achieved F-scores of 79.25% on the MUStARD dataset and 77.36% on UR-FUNNY. Firstly, they used forced alignment to cut audio and video segments corresponding to each word. Next, their approach involved using the ALBERT language model [Lan et al., 2020] for language features and extracting acoustic features such as MFCCs and fundamental frequency from audio segments. Visual features were extracted using OpenFace [Baltrusaitis et al., 2016] for facial Action Units and facial shape parameters. Lastly, a novel aspect of their approach was the incorporation of humour-centric features, specifically ambiguity and sentiment. To capture ambiguity, the authors extracted N senses for each word in the dataset and obtained corresponding Glove embeddings from WordNet. They then calculated the ambiguity metric by summing the cosine distances between all pairs of senses for each word. Regarding sentiment, the authors extracted valence, arousal, and dominance (VAD) scores from the NRC-VAD dictionary [Mohammad, 2018] for each word. These scores provide information about the emotional valence, arousal level, and perceived dominance associated with each word, which can be indicative of the humour expressed. The model architecture consists of transformer encoders for each modality. Then they combine language features with humour-centric features to create an enriched language embedding. Additionally, audio and visual features are used to create non-verbal embedding. These two embeddings are then fed into a Bimodal Cross Attention layer, which consists of Cross Attention and Self Attention heads with residual connections and a feed forward layer. This facilitates the fusion of multimodal information within the model.

In the study by Kayatani et al. [2021], an attention-based model was proposed for multimodal humour detection. They employed Bidirectional Encoder Representations from Transformers (BERT) [Devlin et al., 2019] and LSTM to model the relationship between the setup and punchline of jokes. Additionally, the au-

thors represented speakers as character encodings, which were concatenated with the BERT encoding before being fed into the LSTM. The visual modality was encoded using facial action units extracted from each face in the video segment through OpenFace and a linear layer. Modality attention was then performed using an attention mechanism to weigh the importance of each modality. Through modality ablation studies, the authors found that the performance improvement from additional modalities was limited, and the best performance was achieved by solo punchline modelling, achieving an accuracy of 70.50% on the TBBT dataset.

In another work by Liu et al. [2022], a cross- and self-attention model called FunnyNet was proposed. The authors utilized various encoders for different modalities: BYOL-A [Niizumi et al., 2021] for audio, TimeSformer [Bertasius et al., 2021] for encoded video frames, and InceptionResNet [Szegedy et al., 2016] and LSTM for facial features. The outputs of these encoders were then projected into a common feature space using a Projection Module. A Cross-Attention Fusion module was applied to capture cross-domain correlations among the modalities. Finally, a classification layer was added to make predictions. Notably, FunnyNet employed not only the softmax loss but also a self-supervised contrastive loss to capture mutual multimodal information. Although text modality was not included in the experiments, BERT and LSTM were used for it during the exploration phase. Instead of relying on subtitles, FunnyNet leveraged audio as the primary modality and achieved state-of-the-art performance on most datasets.

In conclusion, multimodal humour detection is a challenging task that has received considerable attention in research. Significant efforts have been made in developing well-annotated datasets and designing multimodal attention-based models, which have shown promising performance. However, there are still some areas that require further exploration. One observation is that the audio modality often plays a dominant role in humour detection, although there is a lack of studies focusing on explaining which specific audio features contribute to the classifier's decisions. Similarly, the influence of visual modality and its specific features on humour detection is also an open question. Furthermore, it is worth noting that the state-of-the-art results in humour detection have predominantly been achieved on English datasets. Given that humour is heavily influenced by culture, it would be interesting to explore the differences and nuances that may exist when working with different language datasets. Overall, while there has been significant progress in multimodal humour detection, there is still room for further investigation to gain a better understanding of the specific modalities and features that contribute to humour detection and to explore the cross-cultural aspects of humour in different languages.

1.4 Goals of this work

The main goals of this work are to explore multimodal humour in two new aspects. Firstly, I aim to focus on the domain of stand-up comedy, which hasn't been extensively studied in this field. Stand-up comedy offers a rich source of humour content, and the audience's laughter serves as a straightforward indicator of humour, enabling automatic annotation and dataset labeling, which I aim to investigate. Secondly, I plan to advance the multilingual aspect of humour de-

tection by introducing a new dataset not only in English but also in the Russian language, which has not been represented before. This will open up opportunities for investigating cross-cultural aspects of humour production. Lastly, I want to conduct initial experiments with unimodal and multimodal humour detection on the new dataset to assess its validity and potential.

2. Datasets

In this chapter, I will describe the steps involved in creating a comprehensive dataset for multimodal humor analysis. I will begin by discussing the process of data collection (Section 2.1), where I gathered a diverse set of stand-up comedy videos. Next, I will explore the annotation phase (Section 2.2), where laughter was manually annotated on a subset of videos to provide a foundation for subsequent analysis. I will then delve into forced alignment techniques (Section 2.3), which ensured accurate synchronization between subtitle text and audio. The segmentation process (Section 2.4) will be described, which involved re-segmenting the subtitle boundaries according to time alignment and sentence markers. I will also explore laughter detection methods (Section 2.5), including both the peak detection approach (Section 2.5.1) and machine learning approaches (Section 2.5.2). Lastly, I will discuss the results of the automated labelling process (Section 2.6).

The dataset collection process began with a focus on the Russian language, as it offered greater novelty and value for the research. All the experiments were primarily conducted using the Russian dataset. Once the Russian dataset was finalized, a similar pipeline was employed to collect the English dataset. This ensured consistency in the data collection process and facilitated comparisons between the two languages.

2.1 Collection

To collect the Russian dataset, I began by manually compiling a list of YouTube channels that featured stand-up performances in Russian with Russian subtitles. Using the pytube library [1], I iterated through the videos on these channels and filtered out those that lacked the keyword "standup" in the title or did not have Russian subtitles. To verify the subtitle language, I downloaded the subtitles and used the langdetect library[2] to detect the actual language, as sometimes the indicated language in the video did not match the subtitles (i.e. subtitles language was Russian, but subtitles were in English). In the end, I collected a total of 46 videos from 8 channels, the list is available in the Appendix A.1. The majority of the videos (31) were from a stand-up club channel in Vladivostok, a region in the Far East of Russia. These videos featured various comedians, including both individual performances and duets, most of them were male. Other videos were sourced from personal channels of comedians from different cities in Russia, also mostly male. It is worth noting that the stand-up industry in Russia is still predominantly male, although there have been some recent changes. Unfortunately, it was challenging to achieve gender balance in the dataset. The collected videos had a total duration of 17 hours, with an average duration of 22.7 minutes (Figure 2.1).

Collecting the English dataset was relatively easier. There is a YouTube channel called "standup" that hosts over 1600 videos. Similar keyword filtering was applied, with the addition of the phrase "full special," and an upper limit of 20 hours on the total length of the videos. In total, 56 videos were collected, with an

[1] `https://github.com/pytube/pytube`

[2] `https://github.com/Mimino666/langdetect`

average duration of 21 minutes (Figure 2.1), the list is available in the Appendix A.2. The majority of the videos featured individual performances by comedians, with only a small percentage including occasional sketches within longer features. The English dataset demonstrated greater diversity, with 24 performances (42%) by female comedians and 22 performances (39%) by comedians of color.

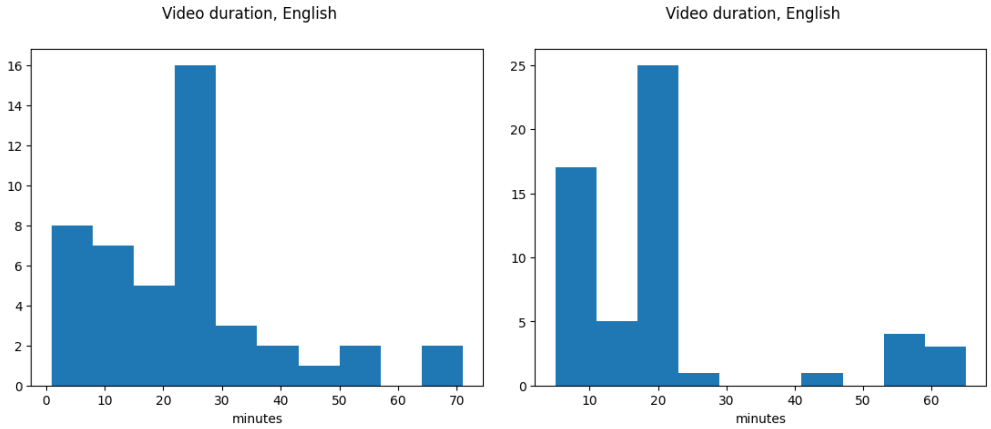

Figure 2.1: Distribution of video lengths in the datasets in minutes.

For downloading video, audio, and subtitles, I utilized various tools including pytube, youtube-transcript-api[3], and youtube-dl[4]. During the collection of the Russian dataset in March 2023, pytube was sufficient for the task. However, when working on the English dataset in June 2023, YouTube changed its API, necessitating the use of alternative tools for media download.

The collected videos showcased a diverse range of non-verbal humour expressions, encompassing facial expressions and body gestures (see Figure 2.2). Additionally, the videos often featured different camera angles, providing multiple perspectives of the comedian's face in close-ups and their body movements in wider shots. Another notable aspect was the consistent use of black or red backgrounds, which is a common aesthetic choice in standup venues, typically featuring stage curtains or exposed brick walls. This shared style could facilitate the extraction of human movement from the background, although it may pose a challenge if one intends to apply the model trained on our dataset to a different domain. Another potential limitation is the microphone held in front of the comedian's face during their performance, which may affect the accuracy of facial landmark detection which is often used for emotion analysis.

2.2 Annotation

To facilitate further experiments on laughter detection, I conducted manual annotations on 5 short videos from each dataset. For the Russian dataset, I selected videos from different channels to ensure diversity. Using the ELAN software [ELAN], I annotated segments of laughter and applause, exporting the annotations into a tabular format. In the annotated files, the mean duration of laughter

[3] `https://github.com/jdepoix/youtube-transcript-api`
[4] `https://github.com/ytdl-org/youtube-dl`

Figure 2.2: Frames from the collected videos expressing different non-verbal laughter cues.

segments was approximately 2 seconds, with a standard deviation of 1.3 seconds. The shortest laughter segment observed was 0.24 seconds, while the longest was 9.4 seconds.

During the annotation process of the Russian subset, I observed certain challenges with the subtitles. Many cases exhibited wrong segmentation, where a single subtitle phrase included multiple sentences or a sentence was divided between multiple subtitles. Additionally, some subtitle time spans included pauses with laughter, where the end of one phrase, a laughter pause, and the start of a new phrase were grouped together. Ideally, I aim to detect laughter after each phrase, rather than inside. Figure 2.3 shows some examples of this behaviour - red spans represent subtitles boundaries, while green spans represent word boundaries. It is visible that subtitle spans include non-word segments or have a border in the middle of the utterance production. Furthermore, different YouTube channels had varying transcription styles. Some channels included labels for auxiliary sounds and turn shifts, while others censored swear words or lacked proper punctuation. This variation resulted in a high degree of noise that required normalization in subsequent stages (as described in the following sections).

In contrast, the transcriptions of the English dataset exhibited less noise and followed a relatively consistent style. Importantly, many subtitles included audio descriptions such as "(crowd cheering)" or "(music playing)," along with laughter labels like "(audience laughing)." These annotations can be utilized for distant supervised annotation of humour.

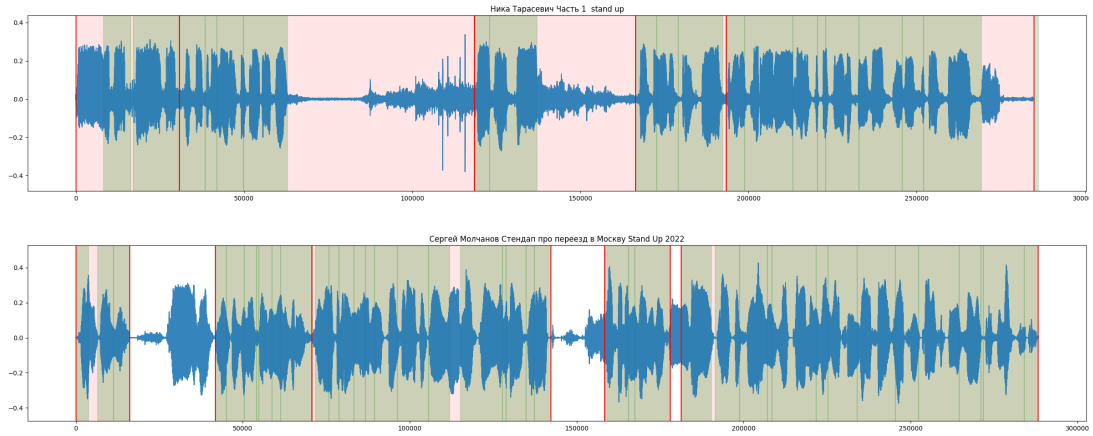

Figure 2.3: Example of the relationship between original subtitle spans and word segments. Red spans represent subtitle time codes, with red vertical lines showing the borders. Green spans represent word spans obtained from forced alignment (Section 2.3).

2.3 Forced alignment

To adjust the timing of subtitles and perform segmentation, I explored different methods and tools. Initially, I experimented with resegmenting the subtitles using rules and aligning the newly segmented sentences using synchronization tools that utilize the Dynamic Time Warping algorithm, such as aeneas[5] and afaligner[6]. Dynamic Time Warping (DTW) is an algorithm used for aligning sequences with varying lengths by warping the time axis. It calculates the optimal alignment path between two sequences by minimizing the distance between corresponding elements, allowing for flexible matching despite differences in timing. For forced alignment, the speech signal is synthesized from the given text, and then it is aligned with the provided audio using the DTW algorithm. Although both libraries support the Russian language, the results obtained from aeneas were consistently poor, so I decided to focus on afaligner.

Afaligner operates on text fragments rather than individual words, so I attempted to align segments using the entire audio. The results were satisfactory, with the mean perfect match score (indicating the number of perfectly aligned fragments) for the Russian annotated videos being 66%, and the less strict score with overlaps reaching 78%.

To further enhance accuracy, I explored the possibility of splitting the audio into smaller chunks, as this would provide the algorithm with less room for error. The chunks were segmented based on the existing subtitle boundaries, with punctuation marks serving as the markers. Chunk alignment yielded a slight improvement, with a perfect match score of 68% and an overlapping match score of 81%. However, despite the progress, the results were still not sufficient, and the algorithm exhibited unpredictable behaviour. Additionally, word alignment proved to be challenging since these algorithms are not specifically designed for that purpose.

[5] `https://github.com/readbeyond/aeneas`
[6] `https://github.com/r4victor/afaligner`

Another family of forced alignment algorithms utilizes phone and word alignment through the use of transcription and a Gaussian Mixture Model-Hidden Markov Model (GMM-HMM) framework. The process involves converting the audio input into its corresponding transcription and training a GMM-HMM acoustic model. This model is then used to align the phonemes (individual speech sounds) and words in the audio by finding the best alignment that maximizes the likelihood of the observed acoustic features given the phonetic and linguistic information. This alignment process enables the precise synchronization of the audio and the corresponding transcriptions. The Montreal Forced Aligner (MFA) is an example of such a forced alignment system that incorporates acoustic models built using the Kaldi ASR toolkit[7]. MFA offers a variety of pre-trained acoustic models and grapheme-to-phoneme models for different languages, including Russian. However, working with MFA is more difficult compared to previous libraries, requiring several steps to obtain word alignments.

The first step in using MFA is to preprocess the text to minimize the number of out-of-vocabulary words. This preprocessing involves removing punctuation, replacing censored swear words, and converting numerals, symbols, and units (such as "$", "%", and "h") into their textual representations. In the case of Russian, which is an inflective language, inflecting numerals and units was challenging, particularly in distinguishing between ordinal and cardinal numbers. As I couldn't find a consistently reliable tool for this task, I resorted to using regular expressions and the num2words library[8]. For swear words, I extracted all words containing the "*" character and attempted to compile a list of regular expressions with the most common roots. However, in many cases, inflexion wasn't evident, and manual verification was necessary. This preprocessing was performed on each subtitle phrase, and alignment was subsequently carried out within these phrases.

To handle out-of-vocabulary words, I utilized the Russian acoustic model[9] to identify such words. Most of these out-of-vocabulary tokens consisted of abbreviations, names, neologisms, misspellings, English words, and vocalizations, amounting to a total of 1861 tokens. Using a grapheme-to-phoneme model, I obtained the transcriptions for these tokens, validated them, and added them to the dictionary. Additionally, I attempted to create transcriptions for English words in the dataset using an English dictionary. In cases where English phonemes were not present in the Russian model, I manually aligned each non-present phoneme with the closest Russian phoneme based on the International Phonetic Alphabet (IPA) chart. For alignment, I used 100 beams and 400 retry beams. Remarkably, only one utterance could not be aligned, resulting in a mean word-level error rate of only 2% on the annotated videos.

The preprocessing steps were also applied to the English dataset, with the additional removal of audio signals such as 'audience laughter' and turn shifting indications like 'Male announcer:'. After validation, a total of 702 out-of-vocabulary tokens were identified in the English dataset - the same categories as in Russian. The grapheme-to-phoneme model was then applied to generate transcriptions for these tokens, which were subsequently added to the dictionary.

Alignment was performed using the same parameters as before, and all seg-

[7] https://github.com/MontrealCorpusTools/Montreal-Forced-Aligner

[8] https://github.com/savoirfairelinux/num2words

[9] https://mfa-models.readthedocs.io/en/latest/acoustic/Russian

ments were successfully aligned. The word level error rate on the annotated videos was 11% mainly due to skewed subtitle boundaries because of the fast speech rate.

After obtaining the word alignment, my next step was to align the preprocessed words with the tokens in the unprocessed text. I made a conscious decision to refrain from further preprocessing the original subtitles text, apart from normalizing white spaces and removing artifacts like time codes. I believe that excessive preprocessing could potentially result in the loss of valuable information encoded in punctuation and spelling. Instead, I opted to align the spans of the unprocessed tokens, which were lazily split using white spaces, with the corresponding forced aligned words. This approach allows users of the dataset to decide on their preferred level of preprocessing based on their specific needs.

The alignment was done using the two-pointer technique. This technique involves using two pointers, each pointing to a specific position in different arrays or sequences. The algorithm compares the elements at the current positions of the pointers, and based on the comparison result, it moves the pointers accordingly. If the elements are the same, both pointers are advanced to the next position. If the elements are different, the second pointer is moved to the next position while the first pointer remains unchanged or moves next of none of the second sequence elements matched it. The first sequence in my case was the unprocessed tokens and the second was the aligned words.

2.4 Segmentation

As mentioned previously, the quality of Russian subtitles posed challenges, particularly in the area of segmentation. When annotating laughter, it is ideal for subtitles to align with laughter borders. The transcription quality also influences the style of separation. The following examples (1-4) illustrate problematic subtitles. The "—" symbol indicates the subtitle border, and translations are provided in brackets. Examples 1-2 demonstrate well-segmented annotations with borders aligning with sentence or clause boundaries. In example 3, subtitles are lengthy, comprising multiple sentences, and the border may appear within a phrase. However, punctuation remains standardized. In example 4, the segmentation is somewhat meaningful, though the lack of punctuation makes sentence segmentation challenging. Some sentences have a capitalized first letter, while others lack even that.

1. Итак, как... | Как вы? Новости смотрите? | Да, просто... | просто ужас происходит.

   So how... | How are you? Do you watch news? | Yes, just... | just awful things happen.

2. Там еще, кстати, группа "Любэ" выступала. | К ним подошли... | "Николай, можете как-нибудь завуалировано?" | И он такой: | "Ой-на-и-на, ой-на-и-на-и-на"

   By the way, band Lyube also performed there. | They were approached... |

"Nikolai, can you do it secretly?" | And he went: | "Oi-na-i-na, oi-na-i-na-i-na"

3. Привет. Как дела у Вас? Всё хорошо у вас? Пошёл к психологу, все ходят, я тоже пошёл к психологу. К | одному, конечно, потому что у меня был бесплатный купон. За 3500 не так уж у меня много проблем. Ясно бл за такие деньги. Вот, но оказалось у меня сейчас в связи с диетой типо очень много проблем. | Есть позитивные стороны, кстати, я начал худеть. Они ждали хаха. Типо круто, что ты похудел, | жирный. Ну короче, я как обнаружил это, я недавно сегодня завязывал шнурки, завязал и пошёл просто.

   Hello. How are you doing? Are you all right? I went to a psychologist, everyone goes, I also went to a psychologist. To | the only one, of course, because I had a free coupon. For 3500 I don't have too many problems. Clearly, for such money. Well, it turned out that I now have a lot of problems in connection with the diet. | There are positive aspects, by the way, I started to lose weight. They were waiting haha. Like it's cool that you've lost weight, | fatty. Well, in short, as I discovered this, I recently tied my shoelaces today, tied it up and just went on.

4. У многих в этом зале, зубы не такие белые | Да и на этой сцене, скажем так, не чем похвастать | ты прямо кашечка | да мур понимаете

   A lot of people in this room don't have white teeth | And on this stage, let's say, there is nothing to brag about | you're just a kitty | yes purr do you understand

There are many ways to address this problem. One option is to perform a straightforward merge of all subtitles using regular expressions. However, this method may encounter difficulties with subtitles lacking sentence boundary punctuations, as seen in example 4. Another possibility is to utilize tools that specialize in sentence segmentation. I conducted a pilot study with a few of these tools, but unfortunately, they did not yield satisfactory results.

Alternatively, given that I already have timestamps for the words through forced alignment, I can leverage the timing information to guide the segmentation process. By identifying pauses between words, I can separate phrases accordingly. Figure 2.4 illustrates the distribution of pause durations between words, which appears to follow an exponential pattern. As a result, determining an explicit threshold becomes challenging. However, after experimenting with various heuristics, I established the following parameters.

To achieve segmentation, I combined two approaches: punctuation-based segmentation and segmentation based on pauses. A phrase is segmented if there is a full stop followed by a word with a capital letter; if there is another punctuation sign and the pause before the next word exceeds 0.3 seconds, or if there is no punctuation but the pause lasts longer than 0.6 seconds.

To evaluate this algorithm, I manually assessed its performance on an annotated subset. I classified the segmentation as satisfactory if there was no interruption within a clause, or if there was, it was accompanied by a substantial pause.

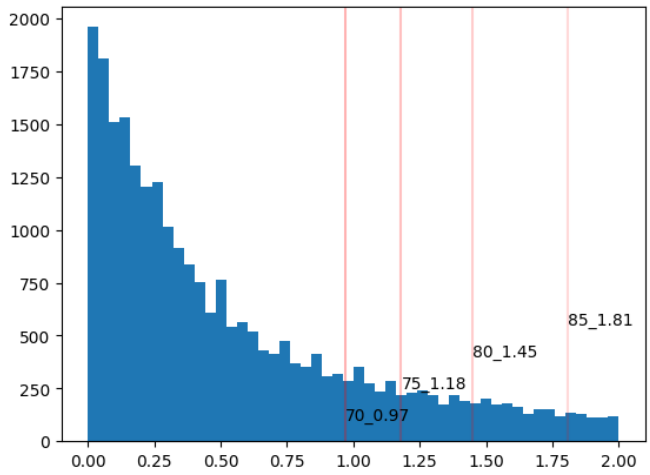

Figure 2.4: Distribution of duration of pauses between words. The x-axis is in seconds, and the red vertical lines represent percentiles and are labelled with their respective values.

The average accuracy of this evaluation was 94%. No other textual preprocessing was done. After segmentation, the number of utterances in the Russian dataset was 18813 and in English 20314.

2.5 Laughter detection

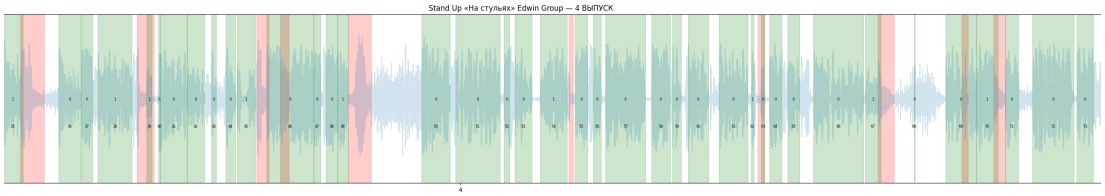

Figure 2.5: Example of laughter occurrences in audio. Green spans are utterances, red spans are audience laughter.

To evaluate laughter detection, I need to consider different approaches. Laughter can be seen as both an event and a sequence. Since most laughter detection tools predict the duration of laughter, a duration-based approach could be used to measure the accuracy of laughter detection at each time step. However, for the current purposes, I don't require such fine-grained precision. Instead, I can treat laughter as an event and focus on identifying whether it occurs or not within specific time spans.

To determine suitable time spans for evaluating laughter detection, I analyzed the patterns of laughter in the annotated videos (Figure 2.5). Based on this examination, I observed that laughter tends to occur either immediately after an utterance or a little bit after - inside the pause between utterances. In some cases, if the pause between utterances is not sufficient, the laughter starts within the next utterance. Therefore, I decided to consider a window after each subtitle to capture laughter occurrences.

To determine an appropriate window length, I calculated the distance of each laughter event from the nearest preceding subtitle and plotted the distribution with the percentiles (see Figure 2.6). The analysis showed that 95% of laughter events occur within 0.7 seconds after the corresponding utterance. This duration seems reasonable to use as the window length. The mean time for a laughter event to occur is 0.26 seconds, suggesting a minimum pause of 0.2 seconds between utterances to consider the pause between them sufficient.

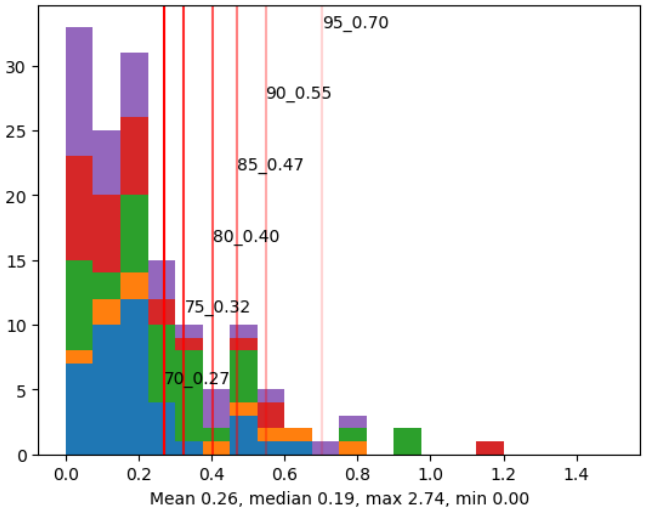

Figure 2.6: Duration of the pause after the utterance and laughter. The bars are stacked up from 5 annotated videos, the x-axis represents the duration in seconds, and vertical lines are the percentiles, labelled with the percentile value.

Based on these findings, I developed an algorithm for defining the laughter window. The window is bounded by either the pause before the next subtitle or, if that pause is smaller than 0.2 seconds, by the end of the subtitle plus 0.7 seconds. Using this approach, the annotated videos were labelled based on the presence or absence of laughter within the defined window. If a laughter event was annotated within the window, the corresponding subtitle was labelled as 1; otherwise, it was labelled as 0. The labelled data allowed for subsequent comparison using basic binary classification scores like accuracy, precision, recall and F-score.

2.5.1 Peak detection approach

In many research papers that automatically labelled their datasets using laughter detection, a common approach is the peak detection method. Typically, these papers utilize audio from sitcoms with high production quality. Importantly, the audio tracks in these cases often contain two channels (left and right), with the actors' voices centred between them. By subtracting the left channel from the right channel, the voice component can be removed, leaving only laughter and music in the remaining audio track.

The peak detection process involves applying a simple signal energy-based detector, such as the auditok library[10], to identify peaks in the audio track. The

[10] `https://github.com/amsehili/auditok`

energy of an audio signal refers to the magnitude or intensity of the signal at different points in time. In the context of audio processing, it is often computed as the squared amplitude of the signal. The energy provides an indication of the overall power or strength of the audio signal.

When using the auditok library for audio activity detection, the energy of the audio signal is analyzed to determine segments with significant audio activity. The library typically applies a threshold-based approach, where a certain energy threshold is set. If the energy of a particular segment exceeds this threshold, it is considered an active or speech segment. Conversely, if the energy falls below the threshold, it is considered a non-speech or silent segment.

Afterwards, the detected segments can be filtered either manually, using signal postprocessing techniques (as demonstrated by Kayatani et al. [2021]), or by employing clustering algorithms (as shown by Liu et al. [2022]). Although this approach seems straightforward, there is currently no available code from any of these papers for direct use.

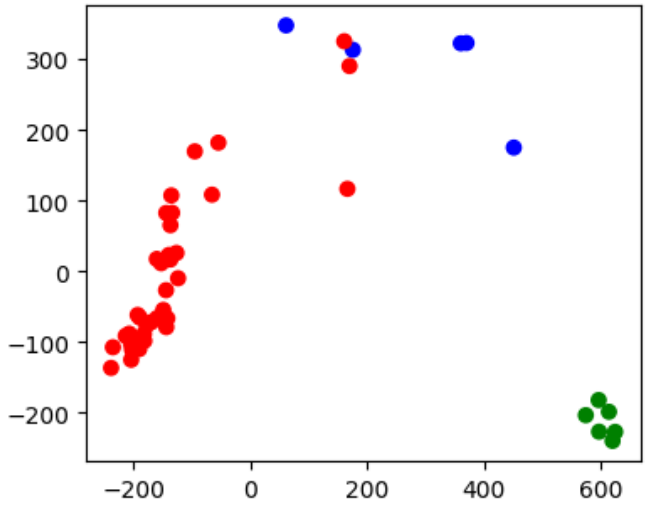

Figure 2.7: Audio after channel subtraction. Features were obtained from Audio Spectrogram Transformer [Gong et al., 2021] and reduced to 2 dimensions using PCA. The audios were manually labelled as unclear (red), successful (blue), and unsuccessful (green).

For the voice removal step, I subtracted the audio channels for the entire Russian dataset. Afterwards, I extracted features from the subtracted audio 1 minute segments using the Audio Spectrogram Transformer [Gong et al., 2021] and applied Principal Component Analysis (PCA) to visualize the resulting groupings. This allowed me to assess how the channel subtraction process affected the audio signals in the dataset. Figure 2.7 displays the subtracted audio visualized as PCA-reduced features. The colour of the point represents the success of the subtraction determined by me. It is evident that the majority of audio exhibits unclear results, represented by the red colour. Additionally, there is a distinct cluster of damaged audio, indicated by the 6 green-coloured data points, this audio resemble no sound, but only random bursts of noises. The 5 successful subtractions are mixed together with unclear ones. Two of the successful videos were part of the annotated section of the dataset.

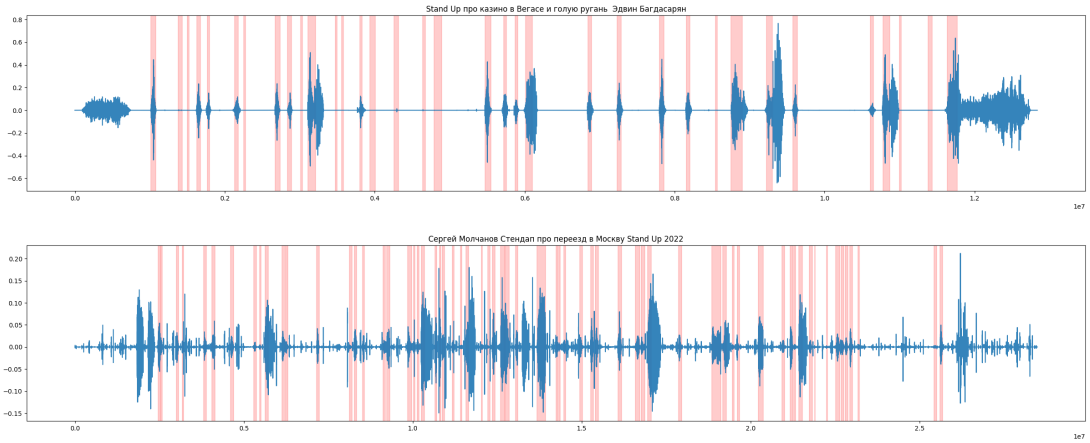

Figure 2.8: Audio tracks after the channels have been subtracted. The annotated laughter segments are highlighted in red spans. First row - ideal audio subtraction case, second row - non-ideal.

The first video achieved almost perfect separation, resulting in a track that contained only laughter, applause, and music. The subtracted audio track for this video is depicted in the first row of Figure 2.8, along with the annotated laughter segments. In this track, some laughter segments are clearly distinguishable, while others are missing after the subtraction process. This video can be considered as the ideal case for voice separation.

The second video also underwent voice separation, but it left more artifacts in the track. The subtracted audio track for this video is shown in the second row of Figure 2.8. Although there are some artifacts present, such as abrupt noises and residual low volume vocals, the laughter segments can still be identified to some extent. This video represents a non-ideal case for voice separation. Although the number of data points for experimentation is limited, it is essential to investigate peak detection technique in case the algorithm can be improved.

Following Liu et al. [2022] I used auditok library to separate audio segments from segments without audio. Auditok is an AUDIo TOKenization tool, it performs audio activity detection based on the energy of the audio signal. To determine the optimal energy threshold for peak detection, I conducted a hyperparameter search. First part of Table 2.1 displays the results.

The mean F-score was used to evaluate the performance of the peak detection algorithm on laughter labeling on two annotated videos. The best-performing energy threshold was found to be 37, resulting in a precision of 77%, recall of 74%, and an F-score of 74%. However, since the audios in the dataset exhibit variations, it is worth considering the audio-specific best results as well. For the ideal audio, the optimal threshold was 25, resulting in a precision of 90%, recall of 73%, and an F-score of 81%. On the other hand, the non-ideal audio achieved its best performance with a threshold of 40, yielding a precision of 66%, recall of 78%, and an F-score of 71%.

Figure 2.9 provides a visual representation of the extracted audio segments along with the annotated laughter (first row - ideal, second row - non-ideal). This approach has the potential to be highly effective if the vocal subtraction step is successful. Furthermore, it appears that if the subtraction process yields

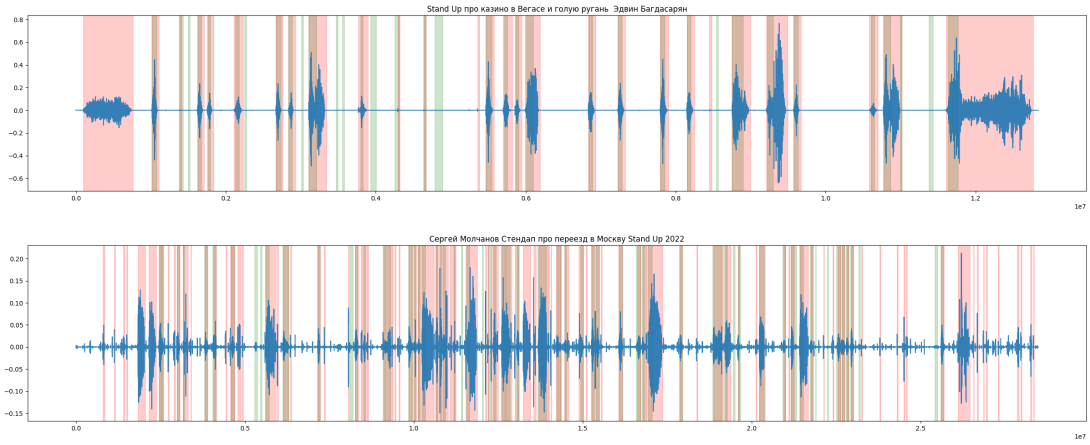

Figure 2.9: Subtracted audio tracks along with the detected peaks (shown in red) and the annotated laughter segments (shown in green). First row - ideal audio with peak detection threshold of 25, second row - non-ideal audio with threshold of 40.

satisfactory results, the energy threshold can be lowered to increase recall while maintaining a reasonable level of precision.

Threshold	Precision		Recall		F-score	
	ideal	non-ideal	ideal	non-ideal	ideal	non-ideal
No cluster filtering						
37	93	62	68	81	78	70
25	90	45	73	100	**81**	62
40	92	66	65	78	76	**71**
With cluster filtering						
54 clustering_avg	96		65		**77**	
40 clustering_ward		79		67		**72**
37 kmeans	92	71	65	64	76	67
25 clustering_avg	92	57	65	43	76	49

Table 2.1: Subtitle labelling results using peak detection with different energy thresholds and clusterization algorithms.

To improve the precision of filtering music, applause, and noises, I explored clusterization techniques inspired by the work of Liu et al. [2022]. For feature extraction, I utilized the above mentioned pre-trained Audio Spectrogram Transformer, a model introduced by Gong et al. [2021], implemented using the huggingface transformers library by Wolf et al. [2020]. The audio segments were resampled to a sample rate of 16000 and trimmed to a duration of 0.5 seconds. This trimming step helped enhance processing speed while still allowing for differentiation between different types of noises. The resulting output vectors had a size of (1024, 128).

Next, I conducted experiments with various clustering methods, including k-means and AgglomerativeClustering with different linkage criterions available in the sklearn library by Pedregosa et al.. These clustering methods were applied to the feature vectors obtained from the audio segments. To aid in the analysis and visualization of the clustering results, I employed PCA for dimensionality reduction. This allowed me to visualise the clusters in lower-dimensional space.

To determine the optimal clustering algorithm for filtering audio segments, I conducted a comprehensive search for the best threshold parameter and clustering algorithm. The clustering was performed with the assumption that there are two classes: laughter and other. For each audio segmentation, I extracted audio features and applied six different clustering algorithms commonly used in machine learning and data analysis that allow a set number of clusters: k-means, agglomerative clustering with four different linkages ('ward', 'complete', 'average', 'single'), and k-means on PCA-reduced points. K-means is an iterative clustering algorithm that aims to partition a given dataset into a predefined number of clusters (k). The algorithm starts by randomly initializing k cluster centroids. It then assigns each data point to the nearest centroid based on the Euclidean distance. After the assignment step, the algorithm recalculates the centroids by taking the mean of all data points assigned to each cluster. This process of assignment and centroid update is repeated until convergence, where the centroids no longer change significantly or a maximum number of iterations is reached. Agglomerative clustering is a hierarchical clustering algorithm that begins with each data point in its own cluster and progressively merges the clusters until a stopping criterion is met. One of the key decisions in agglomerative clustering is how to measure the distance or similarity between clusters - linkage criteria. Ward's linkage criterion aims to minimize the variance within clusters. It calculates the sum of squared differences between all pairs of points in the merged clusters. Ward's linkage tends to produce compact, well-separated clusters. Complete linkage considers the maximum distance between any two points in the merged clusters. It tends to produce clusters with more uniform diameters, but can be sensitive to outliers. Average linkage computes the average distance between all pairs of points in the merged clusters. It strikes a balance between compactness and sensitivity to outliers. Single linkage considers the minimum distance between any two points in the merged clusters. It tends to form long, chain-like clusters and is sensitive to noise and outliers.

To evaluate the quality of the clusterings, I utilized three metrics: the Rand index adjusted for chance, Normalized Mutual Information (NMI), and the Fowlkes-Mallows index (FMI). The Rand index measures the similarity between two clusterings, considering both the agreement and disagreement between pairs of samples, while adjusting for chance agreement. The Normalized Mutual Information quantifies the amount of shared information between two clusterings, taking into account the entropy of the clusterings to provide a normalized measure. The Fowlkes-Mallows index evaluates the similarity between two clusterings by considering pairwise precision and recall, measuring the agreement based on true positive, false positive, and false negative pairs of samples.

To establish the true clusters, I compared the annotated laughter segments with the audio segments to identify any overlaps. For each threshold setting the two best clusterings were selected based on the mean score of the clustering

metrics, and these clusterings were used to filter the audio segments. The final validation was performed for each video by classifying the subtitles using the filtered audio segments. The results of this validation can be found in the second part of Table 2.1.

I initially expected that filtering the audio segments would improve the precision of the classification by eliminating non-laughter segments. However, when comparing the results before and after filtering using the same threshold, I observed an increase in precision of only 2 percentage points for the ideal case and 13 percentage points for the non-ideal case. Unexpectedly, the recall decreased by 8 percentage points for the ideal case and 11 percentage points for the non-ideal case.

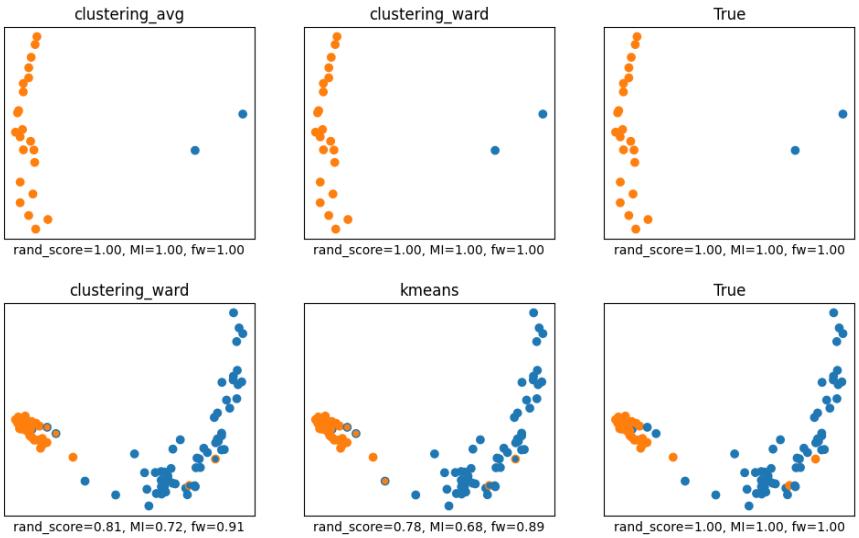

Figure 2.10: The ideal audio (first row) after the clustering analysis with peak detection energy threshold of 54 and non-ideal audio (second row) after the clustering analysis with peak detection energy threshold of 40. The laughter segments are represented by orange points, while the non-laughter segments are represented by blue points.

Interestingly, the best performing clustering method did not outperform the non-clustering approach for the ideal video, with an F-score of 77% compared to 81%. However, for the non-ideal case, there was a slight increase in the F-score from 71% to 72%. It is worth noting that the main challenge in the ideal video is the disappearance of some laughter segments after the channel subtraction, and filtering primarily eliminates music segments (as shown in the first row of Figure 2.10). Therefore, the impact on the overall score is not substantial. On the other hand, for the non-ideal audio, there are many artifacts to filter, which explains why filtering helps improve the score. Second row of Figure 2.10 illustrates that there are more segments available for clustering, and most of them are classified as negative.

Based on these observations, it is clear that the impact of clustering and filtering on the overall performance of laughter detection varies depending on the characteristics of the audio.

Ultimately, the peak detection with clusterization did not result in a significant improvement in the initial scores. However, it holds promise for application in

more noisy audio scenarios. The classification results using the labeling technique are still promising. Unfortunately, due to the limited number of successful vocal subtraction audios, further exploration of this approach has been put on hold for now.

2.5.2 Machine learning approach

In my literature review, I came across only two papers that utilized machine learning-based laughter detection tools. Mittal et al. [2021] and Castro et al. [2019] used the laughter-detection library developed by Gillick et al. [2021] to extract laughter duration. Inspired by their work, I sought to find a similar solution. After conducting thorough research on laughter detection models, I discovered that the laughter-detection library by Gillick et al. [2021] appeared to be the only open-source project available. However, I also experimented with the HUME[11] Baird et al. [2022], but it did not yield satisfactory results, causing me to exclude it from further consideration.

To begin my experimentation, I initially explored whether the laughter detection approach would be effective when applied to the entire audio. I conducted a hyperparameter search, exploring different minimum probability threshold values for the laughter segments. The results of the experiments are in Table 2.2. The best outcome was achieved using a threshold of 0.1, resulting in a precision of 48%, recall of 89%, and an F-score of 61%. However, I noticed that this approach detected a lot of noise, so I decided to limit the search to the specific subtitle window described earlier. Within this context, using a threshold of 0.2, I achieved improved performance with a precision of 63%, recall of 79%, and an F-score of 69%. Although these results show progress, I acknowledge that further improvements are necessary, given the critical importance of accurately labelling this segment of the dataset. If the labelling is incorrect, the subsequent analysis by the models may be affected.

To improve the accuracy of laughter detection, reducing the amount of noise and false positives is crucial. To address this, I explored noise reduction and voice separation techniques. I discovered a deep-learning-based library called vocal-remover[12] that was originally designed for extracting instrumental tracks from songs. Although it was primarily intended for music, I found that it could potentially be applied to separate the main vocals (the comedian's voice) from the background sounds. The library proved to be effective in many cases, successfully isolating the vocals. However, when I attempted to apply the peak detection algorithm to these modified audio files, I encountered some challenges. There were artefacts from the subtraction process, including abrupt noises and residual low-volume vocals in certain videos. Figure 2.11 shows the result of channel subtraction and vocal remover on audio that was unsuccessful with channel subtraction. Additionally, different videos required varying energy thresholds, making it difficult to generalize the peak detection approach for all audios.

Despite the limitations in using the modified audios for peak detection, I realized that they could still serve as valuable input for the laughter prediction model. By conducting the same hyperparameter search as before, I observed an

[11] `https://hume.ai`

[12] `https://github.com/tsurumeso/vocal-remover`

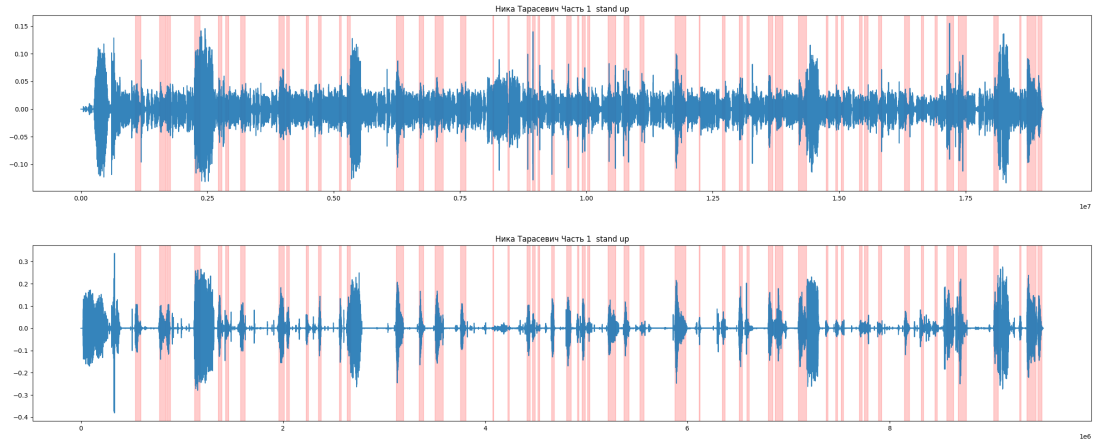

Figure 2.11: Comparison of an audio track after channels subtraction (first row) and vocal-remover (second row). Annotated laughter is shown in highlighted red spans.

improvement in the detection scores. For the entire audio, with a threshold of 0.1, I achieved a precision of 59%, a recall of 75%, and an F-score of 66% (Table 2.2). For the subtitle windows, using a threshold of 0.3, the precision was 69%, the recall was 74%, and the F-score was 71%. Although the increase in scores was not significant, it represented an overall improvement. Therefore, this modified pipeline became the final laughter detection pipeline. For the English dataset, the final F-score was 77%, with a precision of 80% and a recall of 76%.

		Recall	Precision	F-score
Russian	Whole audio	89	48	61
	Window	79	63	69
	Whole audio, vocal-removal	75	59	66
	Window, vocal-remover	74	69	71
English	Window, vocal-remover	76	80	77

Table 2.2: Laughter detection labelling results.

2.6 Labelling

For labelling of the segmented utterances I followed the window annotation approach described in the beginning of this chapter and machine learning approach described in the previous section. The description of the final dataset can be found in Appendix B.1. The dataset statistics are as follows:

For the Russian dataset, the number of negative examples is 9,696, accounting for 52% of the dataset, while the number of positive examples is 9,117, representing 48% of the dataset.

For the English dataset, there are 9,654 negative examples, making up 48% of the dataset, and 10,660 positive examples, constituting 52% of the dataset.

3. Models for humour detection

In this chapter, I will focus on the different models employed for humor analysis. Firstly, I will delve into text-based models (Section 3.1), which utilize various natural language processing techniques to analyze the textual content of humorous messages. This section will explore models such as SVM with TF-IDF vectors and BERT-based models - I will consider them the baselines. Moving on to multimodal models (Section 3.2), I will describe the set up of the simple case of integration of multiple modalities, including text, audio, and visual features, to capture a more comprehensive understanding of humor. Feature extraction and early fusion will be described. Finally, I will delve into describing the experiments conducted (Section 3.3) to evaluate the performance of the different models on our humor dataset.

3.1 Text-based

In the literature review, various models for humour detection in text were discussed. To evaluate their performance on our dataset, I experimented with different models. The first model I used was SVM with an rbf kernel on TF-IDF vectors, implemented using the sklearn library[1]. SVM is a popular supervised machine learning algorithm used for classification and regression tasks. It works by finding an optimal hyperplane that separates the data points into different classes. In the case of classification, SVM aims to find the hyperplane that maximizes the margin between the data points of different classes, allowing for better generalization to unseen data. SVM can handle both linearly separable and non-linearly separable data by using kernel functions that transform the input data into a higher-dimensional space. TF-IDF (Term Frequency-Inverse Document Frequency) is a numerical representation of text documents that quantifies the importance of each word in a document relative to a collection of documents. It calculates the term frequency (TF), which measures the frequency of a word in a document, and the inverse document frequency (IDF), which measures the rarity of a word across all documents in the collection. By multiplying these two measures, TF-IDF assigns higher weights to words that appear frequently in a document but are rare across the entire collection. This helps to capture the distinguishing characteristics of a document. For word tokenization in Russian, I utilized the razdel library[2], while for English, I relied on the nltk library[3].

Another family of models I explored were transformer-based models, with a focus on BERT. BERT is a state-of-the-art language representation model that has revolutionized NLP. BERT is a pretrained model that learns contextualized word representations by training on a large corpus of unlabeled text data. It uses a transformer architecture, which allows it to capture dependencies and relationships between words in a sentence. For classification tasks, BERT can be fine-tuned on a specific dataset by adding a classification layer on top of the pretrained model. During fine-tuning, BERT learns to map input sentences to

[1] `https://scikit-learn.org`

[2] `https://github.com/natasha/razdel`

[3] `https://github.com/nltk/nltk`

corresponding labels. It takes the entire sentence as input, including both left and right context, and generates contextualized word embeddings. These embeddings capture the meaning and nuances of words based on their surrounding context. I leveraged the hugging face transformers library[4] to access pretrained models. For Russian, I employed Conversational RuBERT from DeepPavlov [Burtsev et al., 2018]. This model was trained on OpenSubtitles and social media data, domains that are similar with our dataset. For English, I utilized the original BERT base cased model. The hyperparameters were standard for both models with 5 epochs for training, learning rate warm-up ratio of 0.3, batch size of 16, weight decay of 0.1 and evaluation step every quarter of the epoch.

Additionally, I conducted experiments by firstly fine-tuning Conversational RuBERT on another textual dataset described in Blinov et al. [2019]. This dataset consists of expanded one-liner Stierlitz and Puns dataset from Ermilov et al. [2018], which was further enriched with jokes from social media, non-humorous proverbs, news headlines, and forum posts. The fine-tuning was done on 2 epochs with learning rate warm-up ratio of 0.3, and training with the same parameters described for the model without fine-tuning.

Furthermore, I explored specific models proposed specifically for humour detection. One such model is ColBERT, which was build based on the incongruity theory of humour, as presented in Annamoradnejad and Zoghi [2022]. ColBERT generates embeddings using BERT for both the context sentences and punchline sentence. These embeddings then pass through separate hidden layers and are concatenated to capture the congruity and other relationships between the sentences. Since the implementation code was not provided by the authors, I created the model myself using the pytorch library[5]. The training was done on 5 epochs, with learning rate warm-up ratio of 0.2 and batch size of 16. During the training process, I experimented with two settings: freezing the BERT encoder model and training it along with the other layers.

3.2 Multimodal

In the multimodal setting, I opted for a simple approach using SVM. However, the features used for classification were obtained from different encoders.

For the textual modality, I utilized the BERT [CLS] embeddings of the context and utterance. The context and utterance were separated using the [SEP] token to model attention. Specifically, Conversational RuBERT was used for Russian, BERT cased base for English, and Multilingual BERT base cased for the multilingual setting. The size of the embeddings was 768.

To extract video features, I employed VideoMAE [Tong et al., 2022], a video masked autoencoder pretrained on the Kinetics-400 dataset [Kay et al., 2017], that has 400 human action classes. A video masked autoencoder is a type of deep learning model designed for processing and reconstructing video data. It is a variant of the autoencoder architecture, which is a neural network model that aims to learn an efficient representation of the input data by reconstructing it from a compressed latent space. It applies a mask to certain regions of input

[4] `https://huggingface.co/docs/transformers/index`

[5] `https://pytorch.org/`

video frames during encoding, forcing the model to focus on important features. By learning to encode and decode the masked frames, the model captures essential information while discarding irrelevant details. The encoded representations can then be used as inputs for classification. The VideoMAE model is trained on 16-frame inputs, so I sampled 16 equally spaced frames from the entire context plus utterance window. It's important to note that processing video frames can consume a significant amount of memory, so I used a small batch size of 32 to mitigate memory constraints. Furthermore, during the inference process, I loaded a video into memory and selected a window of 5 context splits. I loaded all the frames from that window and then extracted the 16 frames from each context split and performed feature extraction on them. This loading in batches allowed me to not overload the memory and increase the processing speed, as context splits overlap between each other. The resulting output vector size for each item was (1568, 768). However, considering the large size of this feature vector, I decided to average it along the first dimension to obtain a 768-dimensional vector, which matches the dimensionality of the BERT embeddings.

To incorporate facial features, I utilized OpenFace [Baltrusaitis et al., 2016], a toolkit for facial behavior analysis. Based on the literature review, I selected specific facial features including gaze direction (vertical and horizontal gaze angles), facial action units (18 presence values and 17 intensity values), and non-rigid face shape parameters (deformation due to expression and identity, 34 values). Facial Action Units (FAUs) are a set of specific facial muscle movements or configurations that are used to describe and analyze facial expressions. The detection and analysis of FAUs play a crucial role in understanding and interpreting facial expressions in fields such as emotion recognition, psychology, and human-computer interaction.

To process the facial features, I averaged the context frames into one vector and the utterance frames into another vector. These two vectors were then concatenated, allowing the model to capture the discrepancies between them. The resulting vector size for the facial features is 142, representing the combined information from the context and utterance frames.

3.3 Experiments

To conduct the experiments, I needed to provide context for the models to work with, as humour often involves referencing. For this I divided the dataset into context-utterance chunks. The context consisted of four sentences, while the fifth sentence served as the target utterance. This approach was chosen initially to test the datasets, but other techniques for splitting could also be explored, such as considering temporal information or randomly splitting the data into funny and non-funny parts.

The number of context-utterance splits was determined by evaluating the number of sentences between the positively labeled sentences. As shown in Figure 3.1, it was observed that most contexts contained 2-4 sentences. Following this observation, the Russian dataset was split into 18,629 context-utterance pairs, consisting of 9,582 negative and 9,047 positive examples. Similarly, the English dataset was split into 20,090 pairs, with 9,501 negative and 10,589 positive examples.

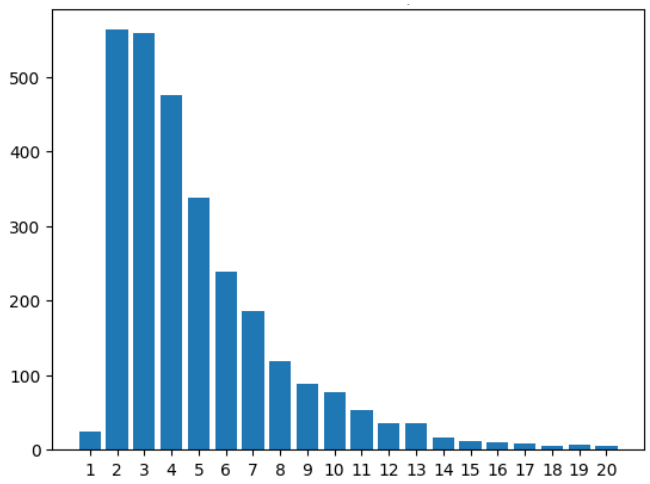

Figure 3.1: Distribution of number of sentences between two positive sentences.

To conduct the experiments with SVM, I utilized the StratifiedShuffleSplit method with 4 splits and a test size of 0.2 and train size of 0.8. This approach ensured that the data was properly stratified during the splitting process. In addition to the SVM experiments, I also conducted random classification and classification using only positive labels on the same splits. I did not perform cross-validation on the neural models because of the time constraints, so the dataset was split only once randomly with a test size of 0.2 and train size of 0.8. For the multimodal experiments, I did classification using each modality separately, in pairs and using all of them.

Furthermore, I selected a random sample of 100 examples from each language dataset to evaluate manually whether humans can detect humor based solely on the text in the dataset. The sample was labeled by a single annotator.

4. Experimental results

In this chapter, I present the results obtained from the humour detection models, both unimodal and multimodal. Specifically, I examine the performance of the models on the Russian subset of the dataset (Section 4.1) as well as the English subset (Section 4.2). Additionally, I explore the performance in a multilingual setting (Section 4.3). Through these sections, I provide an evaluation of the models performance and discuss the findings and implications of the results across two target languages.

4.1 Russian

		Precision	Recall	F-score	Accuracy
Human 100 examples		50	37	42	57
All ones		48.6	100	65.4	48.6
Chance		48.4	49.4	48.9	49.9
SVM + TF-IDF		60.9	58.2	59.5	61.6
Conversational	No pretrain	58.	**68.8**	**62.9**	**66**
RuBERT	Pretrain on FUN	64.4	42	50.8	65.9
ColBERT	BERT freeze	57.4	41.5	48.2	62.6
	BERT unfreeze	**74**	32.8	45.4	**66**

Table 4.1: Best classification results on the Russian dataset using only textual modality.

Table 4.1 displays the classification scores for the text-only models on the Russian dataset. Considering the class distribution, the baseline scores for chance prediction are an F-score of 49% and an accuracy of 50%. It is important to note that even for humans, classifying humor based solely on text proves to be challenging, as evidenced by the human score on the 100 examples, which yielded an F-score of only 42% and an accuracy of 57%. Thus, it is again confirmed that humour detection only based on text is a hard task and we can expect only moderate results from the text-only baseline. However, I did not confirm my expectations of multimodality increasing the humour detection score with a multimodal human annotation.

Examining the results based on the textual modality, the SVM baseline performed relatively well, achieving an F-score of 59.5%, which is higher than chance prediction by 10 percentage points. This indicates that the dataset is suitable for classification tasks. The best score was obtained by the Conversational RuBert classifier without additional pretraining, with an F-score of 63%. The model that was fine-tuned on the FUN dataset performed poorly, which could be attributed to the slight difference in domain between the FUN dataset and the target task. It is worth considering whether fine-tuning on a language modeling task, rather than a humor detection task, would have yielded better results. Additionally, the ColBERT model performed even worse than the SVM baseline, suggesting that further investigation is required to determine if the implementation was correct or if the structure of stand-up jokes differs significantly from other textual formats.

When I analyze the multimodal results in Table 4.2, one key observation is that they do not surpass the performance of the textual BERT model. This was expected, as this multimodal approach relies on SVM and cannot capture complex relationships as effectively. However, what matters more is comparing the results across different modalities, regardless of the textual baselines. We can observe that incorporating additional modalities enhances the overall score. Interestingly, the combination of textual and visual modalities achieves the highest F-score of 61%, while a close second is the combination of all three modalities with an F-score of 60.9%. It is worth noting that the visual modality seems to have the greatest influence, as even when used alone, it achieves a relatively high F-score of 59.2%. On the other hand, the facial modality appears to contribute the least to the overall performance.

		Precision	Recall	F-score	Accuracy
	T	58.5	54.9	56.6	59.2
	V	63	55.9	59.2	62.6
	F	61.5	52.3	56.5	60.9
SVM	TV	63.3	**59**	**61**	63.4
	TF	60.4	56.7	58.5	60.9
	FV	62.9	55.7	59	62.5
	TVF	**63.5**	58.6	60.9	**63.5**

Table 4.2: Multimodal classification results on the Russian dataset. Letters represent the modalities used. T - textual, V - visual, F - facial.

4.2 English

	Precision	Recall	F-score	Accuracy
Human 100 examples	56.3	55.1	55.7	57
All ones	52.7	100	69	52.7
Chance	52.6	50	51.3	49.9
SVM + TF-IDF	62.1	69.3	65.5	61.5
BERT	**63.1**	**76.5**	**69.2**	**63.6**

Table 4.3: Best classification results on the English dataset using only textual modality.

The results of the textual classification can be found in Table 4.3. Similar to the Russian dataset, the human evaluation exhibited poor performance, achieving an F-score of 55.7%. However, it performed better than the Russian dataset but it does not say much since the annotation was done only by one annotator. The SVM baseline showed relatively good performance with an F-score of 65.5%, surpassing chance by 1.42 percentage points. Once again, the best result was achieved by the BERT model, obtaining an F-score of 69.2%. This outcome reinforces the versatility of transformer models. No further BERT modification experiments were conducted on the English dataset since they did not yield any improvement on the Russian dataset.

Turning to the multimodality experiments and referring to Table 4.4, we can once again observe that incorporating additional modalities enhances the score. In contrast to the Russian dataset, the highest score in this case was achieved by the visual and facial model, with an F-score of 67.8%. Interestingly, for the English dataset, the textual modality appeared to be the least influential, while the visual modality once again demonstrated exceptional performance, even when used independently, resulting in an F-score of 67.3%. Furthermore, the combination of all three modalities yielded a comparable result to the visual-facial setting, with an F-score of 67.5%.

		Precision	Recall	F-score	Accuracy
	T	54.7	68.5	60.9	53.5
	V	61.8	74	67.3	62.1
	F	62.2	70.9	66.3	61.9
SVM	TV	61.2	**74.5**	67.2	61.6
	TF	59.3	68.5	63.6	58.7
	FV	**62.7**	73.7	**67.8**	**63.1**
	TVF	62.3	73.8	67.5	62.6

Table 4.4: Multimodal classification results on the English dataset. Letters represent the modalities used. T - textual, V - visual, F - facial.

4.3 Multilingual

The multilingual text-based classification using Multilingual BERT produced a similar result to the language-specific models, achieving an F-score of 65.9% (Table 4.5).

	Precision	Recall	F-score	Accuracy
Multilingual BERT	53.6	85.4	65.9	55.3

Table 4.5: Multilingual classification results on text.

In the multilingual multimodal setting, I conducted evaluations not only on the mean scores but also for each language, as shown in Table 4.6. Generally, we observe the same pattern where adding modalities increases the score. The best score was obtained by using all modalities, resulting in an F-score of 64.7%. Specifically, for the Russian part, the F-score was 60.6%, while for the English part, it was 67.7%. The English result is very close to the best result obtained with the English BERT embeddings, with only a 0.1 percentage point difference. And for the Russian part, the result is worse by 0.4 percentage points, which is still acceptable. It's possible that the Multilingual BERT model has more representation for English than for Russian, contributing to the discrepancy in performance.

Our multilingual experiments provide evidence that using both datasets together is feasible. However, to effectively investigate the language and culture-specific differences, it would be necessary to employ more advanced models and conduct a more comprehensive research study.

	Precision		Recall		F-score		Accuracy	
	eng	rus	eng	rus	eng	rus	eng	rus
T	56		64.8		60.1		56.3	
	54.4	58.8	73.9	54	62.6	56.3	53.4	59.5
V	61.9		65		63.5		62	
	61.6	62.5	73.9	54.6	67.2	58.3	62.1	61.9
F	61.5		61.2		61.4		60.9	
	62.9	59.6	67.8	53.4	65.2	56.3	62	59.7
TV	62		**67.2**		64.5		62.5	
	61.1	63.5	**74.9**	**57.9**	67.3	**60.6**	61.8	63.3
TF	60.1		63.8		61.9		60.2	
	59.3	61.4	70.3	56.1	64.3	58.6	59	61.4
FV	**62.8**		64.8		63.8		62.6	
	62.8	62.7	73.2	54.8	67.6	58.5	**63.2**	62.1
TVF	62.6		66.8		**64.7**		**63**	
	61.9	**63.8**	74.6	57.6	**67.7**	**60.6**	62.5	**63.5**

Table 4.6: Multimodal multilingual classification results on both datasets.

5. Discussions

In this chapter, I discuss the findings and implications of the study. In Section 5.1, I delve into the role of laughter in stand-up comedy, highlighting its unique nature and its impact on audience response. Section 5.2 focuses on the quality of labelling and laughter detection approaches, addressing the challenges and potential improvements in accurately annotating and detecting laughter in the dataset. In Section 5.3, I discuss potential advanced multimodal modeling techniques and their possible application in detecting language-specific humour features in multilingual setting. Lastly, in Section 5.4, I reflect on the ethics of humor, considering the potential implications and responsibilities associated with the development and use of humor detection models.

5.1 Laughter in stand-up comedy

It is important to acknowledge that using laughter as the primary marker of humor for labeling the datasets has its limitations, particularly in the context of stand-up comedy. Stand-up comedy laughter differs in several aspects that need to be considered.

Firstly, laughter during a stand-up show is not as spontaneous as laughter in dialogues. Audience members attend these shows with the expectation of being entertained and experiencing laughter. As a result, they may find things amusing that would not typically produce laughter in other contexts. Additionally, the relationship between a stand-up comedian and the audience is unique. It can be seen as a form of dialogue, however an unequal one, with trust playing a significant role in this dynamic [Abrahams, 2020].

Secondly, it is crucial to recognize that not all laughter in stand-up comedy is the same. Bochkarev [2022] distinguishes three types of laughter commonly found in stand-up comedy: warming-up, main, and follow-up laughter. Warming-up laughter occurs in response to secondary jokes that serve to prepare the audience for the main joke. It is often shorter, less loud, and involves a smaller portion of the audience. This type of laughter can also be triggered by the use of explicit or vulgar language. Main laughter, on the other hand, is long, loud, and comes from the entire audience. Lastly, follow-up laughter refers to laughter that occurs after the punchline of a joke, which can sometimes be a reaction to non-humorous utterances. It may arise from the comedian paraphrasing the same joke again or from the audience remaining in a funny mood.

Therefore, any conclusions drawn from the stand-up comedy data should be analyzed with these factors in mind. The unique nature of stand-up comedy laughter and the specific dynamics between the comedian and the audience should be considered when interpreting the findings.

5.2 Quality of labelling and laughter detection approaches

Due to the use of automated labeling and the limited subset of videos for evaluation, the quality of the dataset may not be optimal and it may influence model performances, since I use it for training and evaluation and there is no human annotated gold standard. Although I achieved a reasonably good labeling result with an F-score of 71% for Russian and 77% for English, there is room for improvement to ensure consistent results.

One potential approach to enhance the labeling process is to switch to a different method, such as peak detection. I consider this approach more reliable than machine learning-based detection. However, there are some prerequisites that need to be addressed, particularly the separation of vocals from the audio. Currently, vocal reduction techniques tend to introduce artifacts and noise. To overcome this challenge, a more sophisticated method could be implemented, which may involve filtering out noise or employing alternative noise reduction techniques. Clusterization filtering could also be explored more, for example by using different feature extraction techniques.

Another issue to consider is the variation in audio recordings occurring in different environments. Even if background noises are extracted, they may differ in volume and quality across various videos. Normalizing these noises or applying filters to enhance their separability could be potential solutions. Additionally, the current approach employed clustering into two classes, but it might be necessary to consider using three or more classes to account for music, artifacts, non-laughter noise, and laughter.

By addressing these challenges and refining the labeling approach, we can strive for improved dataset quality and more accurate and consistent results.

5.3 Advanced multimodal modelling and multilinguality

Relying solely on SVM is insufficient, and it is necessary to explore and implement additional models to investigate the contribution of different features to laughter detection. Liu et al. [2022] conducted a study in which they analyzed the average attention values on cross-attention modules to determine the most influential features. Their findings suggest that audio features contribute the most to the classification, followed by visual features, and then facial expressions. Facial features have a greater impact when clear emotional facial expressions are present. They also examined the modality attention and discovered that each modality primarily attends to itself but also captures correlations between audio and visual expressions. Additionally, there were instances where modalities also attended to facial features.

Another study conducted by Hasan et al. [2021] focused on feature importance but using features aligned with text tokens, enabling them to map weights to specific tokens in the sentences. They identified "humor anchors" using the integrated gradient method [Sundararajan et al., 2017]. "Humour anchors" are textual and visual tokens that play a significant role in humor detection. Their

findings indicate that the model attributes high importance to meaningful patterns such as eyebrow raises, exaggerated facial expressions, stress on tone, high valence, and arousal.

Employing techniques like attention models can provide valuable insights into our dataset, particularly considering that there are two languages. It allows us to investigate differences between cultures and determine which features are more prominent in each dataset. Furthermore, we can explore regional or gender differences, and in the case of the English dataset, racial differences, as there may be variations among individuals who have been racialized in different communities.

Also, it is crucial to take into account not only the context frames before the target utterance but also the frames after it. This is particularly important in the case of visual comedy, where it has been observed that there can be codas or additional comedic elements that occur after the initial utterance. In some instances, the main punchline or humorous aspect of a joke may be delivered after the utterance itself. By considering the frames after the target utterance, we can capture and analyze these essential elements that contribute to the overall humor and comedic effect.

5.4 Ethics of humour

Since I have not personally reviewed the entire dataset, the quality of the content is mainly dependent on the filtering done by the individuals who uploaded the stand-up routines on YouTube. Therefore, I cannot ensure that the content is not offensive, does not propagate hurtful stereotypes of marginalized groups, and avoids explicit language. The English part of the dataset was collected from a well-known source, which may provide some level of reassurance. However, the Russian part is sourced from different channels, and during the annotation process, I encountered instances of disparaging humor and a significant amount of profanity. It is essential to acknowledge that while this content falls within the realm of humor, it can also be hurtful.

Despite these considerations, if our goal is to study humor, it is necessary to include this aspect and not filter it out completely. However, it is crucial to recognize the potential harm that can arise from such content. As we focus on detecting humor, our models need to be aware of these nuances without necessarily endorsing or promoting offensive material.

Conclusion and future work

This thesis aimed to explore the detection of humor through multimodal approaches, focusing on stand-up comedy in Russian and English languages. By collecting a novel multilingual dataset and conducting experiments, I have gained valuable insights into the challenges and potential solutions in multimodal humor detection.

Throughout the research process, the collection and annotation of the dataset proved to be crucial steps. Our primary focus was on the Russian language, as there was a lack of multimodal datasets specifically designed for humor detection in this language. Since there is practically no TV-shows with subtitles in Russian stand-up comedy was chosen as the source for collection. Stand-up has gained significant popularity and a large online audience, particularly on platforms like YouTube, where creators can also upload subtitles. To gather the dataset, I crawled through stand-up comedy videos on YouTube, specifically targeting those with appropriate language subtitles. Forced alignment and segmentation techniques helped ensure accurate and precise data splitting, providing a gold standard for labeling experiments. The exploration of various laughter detection approaches, including peak detection and machine learning-based techniques, showed that laughter detection is still a very challenging task, which is highly influenced by the audio quality.

The peak detection approach involves preprocessing the audio signal to separate vocals from background noises. The algorithm then identifies non-silent segments of the signal, including laughter, background noise, and music, by detecting peaks that exceed a threshold. These detected peaks can be further filtered using clustering techniques. The peak detection approach offers a relatively straightforward and efficient method for identifying laughter in audio signals. However, it can be sensitive to variations in audio characteristics and background noise levels, which may require careful tuning of parameters to achieve optimal results. On the other hand, the machine learning approach utilizes an end-to-end pretrained model to detect the presence or duration of laughter without the need for preprocessing. However, the audio quality can still impact the results, and an appropriate threshold must be chosen. While the peak detection approach offers higher accuracy potential, it requires higher audio quality and more advanced preprocessing and postprocessing steps. As a result, the machine learning approach was chosen for its simplicity and generalization, although it lacks interpretability and fine-grained control. To improve the machine learning approach, vocal removal preprocessing using another neural model was employed. The final labeling scores for the Russian dataset achieved an F-score of 71% and 77% for English.

After exploring various laughter detection algorithms on a subset of audio samples, I employed the best performing approach for labeling the dataset with humor annotations. The labeled datasets were subsequently subjected to validity checks through the training of baseline humor detection models, including experiments in multimodal humor recognition. For both languages in only textual setting BERT based models performed the best with a 14 percentage points increase in F-score from the random baseline for Russian (48.9% compared to 62%)

and 18 percentage points increase for English (51.3% compared to 69.2%).

In the multimodal setting, I conducted a simple experiment combining different modality features in an SVM classifier, therefore I did not expect the results to surpass the neural textual baseline. Nonetheless, the experiment revealed the influence of different modalities. For the Russian dataset, the visual modality had the most significant impact in the unimodal setting, achieving an F-score of 59.2% compared to 56.6% for textual and 56.4% for facial modalities. In the multimodal setting, the best performance was observed with the combination of textual and visual modalities, achieving an F-score of 61%, followed closely by using all three modalities together with a score of 60.9%.

Similarly, for the English dataset, the visual modality performed the best in the unimodal setting with an F-score of 67.3%, compared to 60.9% for textual and 66.3% for facial modalities. In the multimodal setting, the combination of facial and video modalities achieved the highest performance with an F-score of 67.8%, while using all three modalities yielded an F-score of 67.5%. Notably, the models with textual input underperformed compared to those with visual input in the multimodal setting. However, since SVM does not allow exploration of feature influence, it remains unclear which specific features contributed to this behavior.

Overall, the results of the detection models were satisfactory, considering the inherent difficulty of the task. Furthermore, the multimodal experiments hint that visual modality has potential to increase the detection performance. Although more advanced experiments are needed to draw stronger conclusions.

For future work, several improvements can be suggested. Firstly, it is crucial to further investigate laughter labeling techniques to find a more reliable and scalable approach. Secondly, expanding the dataset to include more languages, such as a separate English dataset based on British comedy or an Italian dataset, would enable the evaluation of cross-cultural differences in humor production. Lastly, exploring more advanced multimodal humor detection models that can facilitate the investigation of feature importance would be beneficial.

In summary, this thesis contributes to the field of humor detection by providing insights into the complexities and potential strategies for detecting humor in multimodal data. The main goals, stated in Section 1.4 were achieved by collecting the new dataset and utilizing laughter detection to label it automatically. However, it is important to acknowledge that certain ambitions could not be fully realized due to the time-consuming nature of dataset collection, investigation and processing. Specifically, the experimental part of the work was limited to simple models, and the exploration of desired language and cultural settings, such as British vs. American humor, was constrained. As a result, the multilingual aspects of humor detection was not as complete as originally intended. Nevertheless, this thesis serves as a foundation for further research, such as exploring cultural and linguistic influences on humor perception and developing more advanced and culturally sensitive models for humor analysis and development of intelligent systems capable of recognizing and appreciating humor in diverse contexts.

Bibliography

Daniel Abrahams. Winning Over the Audience: Trust and Humor in Stand-Up Comedy. *The Journal of Aesthetics and Art Criticism*, 78:491–500, September 2020. doi: 10.1111/jaac.12760.

Marc Aguert. Paraverbal Expression of Verbal Irony: Vocal Cues Matter and Facial Cues Even More. *Journal of Nonverbal Behavior*, 46:1–26, March 2022. doi: 10.1007/s10919-021-00385-z.

Issa Annamoradnejad and Gohar Zoghi. ColBERT: Using BERT Sentence Embedding in Parallel Neural Networks for Computational Humor, December 2022. URL `http://arxiv.org/abs/2004.12765`. arXiv:2004.12765 [cs] version: 7.

Salvatore Attardo. Linguistic Theories of Humor. In *Linguistic Theories of Humor*. De Gruyter Mouton, January 2010. ISBN 978-3-11-021902-9. doi: 10.1515/9783110219029. URL `https://www.degruyter.com/document/doi/10.1515/9783110219029/html?lang=en`.

Salvatore Attardo, Jodi Eisterhold, Jennifer Hay, and Isabella Poggi. Multimodal markers of irony and sarcasm. *Humor: International Journal of Humor Research*, 16(2):243–260, 2003. ISSN 1613-3722. doi: 10.1515/humr.2003.012. Place: Germany Publisher: Walter de Gruyter.

Alice Baird, Panagiotis Tzirakis, Jeffrey A. Brooks, Christopher B. Gregory, Björn Schuller, Anton Batliner, Dacher Keltner, and Alan Cowen. The ACII 2022 Affective Vocal Bursts Workshop & Competition: Understanding a critically understudied modality of emotional expression, October 2022. URL `http://arxiv.org/abs/2207.03572`. arXiv:2207.03572 [cs, eess].

Tadas Baltrusaitis, Peter Robinson, and Louis-Philippe Morency. OpenFace: An open source facial behavior analysis toolkit. pages 1–10, March 2016. doi: 10.1109/WACV.2016.7477553.

Valeriya Baranova-Bolotova, Vladislav Blinov, and Pavel Braslavski. Lightning Talk - Humor Recognition in Russian Language. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 1268–1269, San Francisco USA, May 2019. ACM. ISBN 978-1-4503-6675-5. doi: 10.1145/3308560.3316473. URL `https://dl.acm.org/doi/10.1145/3308560.3316473`.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is Space-Time Attention All You Need for Video Understanding?, June 2021. URL `http://arxiv.org/abs/2102.05095`. arXiv:2102.05095 [cs].

Dario Bertero and Pascale Fung. Deep Learning of Audio and Language Features for Humor Prediction. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 496–501, Portorož, Slovenia, May 2016a. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1079`.

Dario Bertero and Pascale Fung. A Long Short-Term Memory Framework for Predicting Humor in Dialogues. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 130–135, San Diego, California, June 2016b. Association for Computational Linguistics. doi: 10.18653/v1/N16-1016. URL https://aclanthology.org/N16-1016.

Vladislav Blinov, Valeria Bolotova-Baranova, and Pavel Braslavski. Large Dataset and Language Model Fun-Tuning for Humor Recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4027–4032, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1394. URL https://aclanthology.org/P19-1394.

Giuseppe Boccignone, Donatello Conte, Vittorio Cuculo, and Raffaella Lanzarotti. AMHUSE: a multimodal dataset for HUmour SEnsing. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, ICMI '17, pages 438–445, New York, NY, USA, November 2017. Association for Computing Machinery. ISBN 978-1-4503-5543-8. doi: 10.1145/3136755.3136806. URL https://doi.org/10.1145/3136755.3136806.

Arsentiy I. Bochkarev. Classification of Laughter in Stand-Up Comedies. *European Proceedings of Educational Sciences*, Topical Issues of Linguistics and Teaching Methods in Business and Professional Communication - TILTM 2022, December 2022. ISSN 2672-815X. doi: 10.15405/epes.22104.6.

Mikhail Burtsev, Alexander Seliverstov, Rafael Airapetyan, Mikhail Arkhipov, Dilyara Baymurzina, Nickolay Bushkov, Olga Gureenkova, Taras Khakhulin, Yuri Kuratov, Denis Kuznetsov, Alexey Litinsky, Varvara Logacheva, Alexey Lymar, Valentin Malykh, Maxim Petrov, Vadim Polulyakh, Leonid Pugachev, Alexey Sorokin, Maria Vikhreva, and Marat Zaynutdinov. DeepPavlov: Open-Source Library for Dialogue Systems. In *Proceedings of ACL 2018, System Demonstrations*, pages 122–127, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-4021. URL https://aclanthology.org/P18-4021.

Davide Buscaldi and Paolo Rosso. Some Experiments in Humour Recognition Using the Italian Wikiquote Collection. In Francesco Masulli, Sushmita Mitra, and Gabriella Pasi, editors, *Applications of Fuzzy Sets Theory*, Lecture Notes in Computer Science, pages 464–468, Berlin, Heidelberg, 2007. Springer. ISBN 978-3-540-73400-0. doi: 10.1007/978-3-540-73400-0_58.

Santiago Castro, Devamanyu Hazarika, Verónica Pérez-Rosas, Roger Zimmermann, Rada Mihalcea, and Soujanya Poria. Towards Multimodal Sarcasm Detection (An _obviously_ Perfect Paper). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4619–4629, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1455. URL https://aclanthology.org/P19-1455.

Dushyant Singh Chauhan, Gopendra Vikram Singh, Aseem Arora, Asif Ekbal, and Pushpak Bhattacharyya. A Sentiment and Emotion Aware Multimodal

Multiparty Humor Recognition in Multilingual Conversational Setting. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6752–6761, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.587.

Lei Chen and Chong MIn Lee. Predicting Audience's Laughter Using Convolutional Neural Network, May 2017. URL http://arxiv.org/abs/1702.02584. arXiv:1702.02584 [cs] version: 2.

Lukas Christ, Shahin Amiriparian, Alexander Kathan, Niklas Müller, Andreas König, and Björn W. Schuller. Multimodal Prediction of Spontaneous Humour: A Novel Dataset and First Results, September 2022. URL http://arxiv.org/abs/2209.14272. arXiv:2209.14272 [cs, eess].

Alessandra Teresa Cignarella, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, Paolo Rosso, and Farah Benamara. Multilingual Irony Detection with Dependency Syntax and Neural Models, November 2020. URL http://arxiv.org/abs/2011.05706. arXiv:2011.05706 [cs].

Gaétane Deliens, Kyriakos Antoniou, Elise Clin, Ekaterina Ostashchenko, and Mikhail Kissine. Context, facial expression and prosody in irony processing. *Journal of Memory and Language*, 99:35–48, April 2018. ISSN 0749-596X. doi: 10.1016/j.jml.2017.10.001. URL https://www.sciencedirect.com/science/article/pii/S0749596X17300839.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019.

ELAN. ELAN (Version 6.5) [Computer software]. Nijmegen: Max Planck Institute for Psycholinguistics, The Language Archive., 2023. URL https://archive.mpi.nl/tla/elan.

Raissa Ellis. Testing the impact of paraverbal irony signals. Experimental study on verbal irony identification in face-to-face and computer-mediated communication. *Psychology of Language and Communication*, 26:65–84, March 2022. doi: 10.2478/plc-2022-0004.

Anton Ermilov, Natasha Murashkina, Valeria Goryacheva, and Pavel Braslavski. Stierlitz Meets SVM: Humor Detection in Russian. In Dmitry Ustalov, Andrey Filchenkov, Lidia Pivovarova, and Jan Žižka, editors, *Artificial Intelligence and Natural Language*, Communications in Computer and Information Science, pages 178–184, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01204-5. doi: 10.1007/978-3-030-01204-5_17.

Raymond Gibbs. Irony in Talk Among Friends. *Metaphor and Symbol - METAPHOR SYMB*, 15:5–27, April 2000. doi: 10.1207/S15327868MS151&2_2.

Jon Gillick, Wesley Deng, Kimiko Ryokai, and David Bamman. Robust Laughter Detection in Noisy Environments. In *Interspeech 2021*, pages 2481–2485. ISCA, August 2021. doi: 10.21437/Interspeech.

2021-353. URL https://www.isca-speech.org/archive/interspeech_2021/gillick21_interspeech.html.

Yuan Gong, Yu-An Chung, and James Glass. AST: Audio Spectrogram Transformer, July 2021. URL http://arxiv.org/abs/2104.01778. arXiv:2104.01778 [cs].

Santiago González-Fuente, Victoria Escandell-Vidal, and Pilar Prieto. Gestural codas pave the way to the understanding of verbal irony. *Journal of Pragmatics*, 90:26–47, December 2015. ISSN 0378-2166. doi: 10.1016/j.pragma.2015.10.002. URL https://www.sciencedirect.com/science/article/pii/S0378216615002714.

Md Kamrul Hasan, Wasifur Rahman, Amir Zadeh, Jianyuan Zhong, Md Iftekhar Tanveer, Louis-Philippe Morency, Mohammed, and Hoque. UR-FUNNY: A Multimodal Language Dataset for Understanding Humor. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2046–2056, 2019. doi: 10.18653/v1/D19-1211. URL http://arxiv.org/abs/1904.06618. arXiv:1904.06618 [cs, stat].

Md Kamrul Hasan, Sangwu Lee, Wasifur Rahman, Amir Zadeh, Rada Mihalcea, Louis-Philippe Morency, and Ehsan Hoque. Humor Knowledge Enriched Transformer for Understanding Multimodal Humor. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12972–12980, May 2021. ISSN 2374-3468. doi: 10.1609/aaai.v35i14.17534. URL https://ojs.aaai.org/index.php/AAAI/article/view/17534. Number: 14.

Nabila Hasyim and Sharifah Hanidar. Verbal Irony in a TV Series The Office (US) Season 2. *Lexicon*, 9:63, October 2022. doi: 10.22146/lexicon.v9i2.68005.

Antony Kalloniatis and Panagiotis Adamidis. Computational Humor Recognition: A Systematic Literature Review, February 2023. URL https://www.researchsquare.com.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics Human Action Video Dataset, May 2017. URL http://arxiv.org/abs/1705.06950. arXiv:1705.06950 [cs] version: 1.

Yuta Kayatani, Zekun Yang, Mayu Otani, Noa Garcia, Chenhui Chu, Yuta Nakashima, and Haruo Takemura. The Laughing Machine: Predicting Humor in Video. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2072–2081, Waikoloa, HI, USA, January 2021. IEEE. ISBN 978-1-66540-477-8. doi: 10.1109/WACV48630.2021.00212. URL https://ieeexplore.ieee.org/document/9423313/.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, February 2020. URL http://arxiv.org/abs/1909.11942. arXiv:1909.11942 [cs].

Zhisong Liu, Robin Courant, and Vicky Kalogeiton. FunnyNet: Audiovisual Learning of Funny Moments in Videos. pages 3308–3325, 2022. URL https://openaccess.thecvf.com/content/ACCV2022/html/Liu_FunnyNet_Audiovisual_Learning_of_Funny_Moments_in_Videos_ACCV_2022_paper.html.

Anirudh Mittal, Pranav Jeevan, Prerak Gandhi, Diptesh Kanojia, and Pushpak Bhattacharyya. "So You Think You're Funny?": Rating the Humour Quotient in Standup Comedy, October 2021. URL http://arxiv.org/abs/2110.12765. arXiv:2110.12765 [cs].

Saif Mohammad. Obtaining Reliable Human Ratings of Valence, Arousal, and Dominance for 20,000 English Words. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1017. URL https://aclanthology.org/P18-1017.

M. Mulder and Anton Nijholt. Humour Research: State of the Art. November 2002.

Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. BYOL for Audio: Self-Supervised Learning for General-Purpose Audio Representation, April 2021. URL http://arxiv.org/abs/2103.06695. arXiv:2103.06695 [cs, eess].

Neal Norrick. Non-verbal humor and joke performance. *Humor-international Journal of Humor Research - HUMOR*, 17:401–409, January 2004. doi: 10.1515/humr.2004.17.4.401.

Badri N. Patro, Mayank Lunayach, Deepankar Srivastava, Sarvesh Sarvesh, Hunar Singh, and Vinay P. Namboodiri. Multimodal Humor Dataset: Predicting Laughter tracks for Sitcoms. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 576–585, Waikoloa, HI, USA, January 2021. IEEE. ISBN 978-1-66540-477-8. doi: 10.1109/WACV48630.2021.00062. URL https://ieeexplore.ieee.org/document/9423266/.

Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, and David Cournapeau. Scikit-learn: Machine Learning in Python. *MACHINE LEARNING IN PYTHON*.

Maxime Peyrard, Beatriz Borges, Kristina Gligorić, and Robert West. Laughing Heads: Can Transformers Detect What Makes a Sentence Funny?, August 2021. URL http://arxiv.org/abs/2105.09142. arXiv:2105.09142 [cs].

Kimiko Ryokai, Elena López, Noura Howell, Jon Gillick, and David Bamman. Capturing, Representing, and Interacting with Laughter. pages 1–12, April 2018. doi: 10.1145/3173574.3173932.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic Attribution for Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3319–3328. PMLR, July 2017.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, August 2016. URL http://arxiv.org/abs/1602.07261. arXiv:1602.07261 [cs].

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training, October 2022. URL http://arxiv.org/abs/2203.12602. arXiv:2203.12602 [cs].

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, December 2017. URL http://arxiv.org/abs/1706.03762. arXiv:1706.03762 [cs].

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. HuggingFace's Transformers: State-of-the-art Natural Language Processing, July 2020. URL http://arxiv.org/abs/1910.03771. arXiv:1910.03771 [cs].

Shiwei Zhang, Xiuzhen Zhang, Jeffrey Chan, and Paolo Rosso. Irony detection via sentiment-based transfer learning. *Information Processing & Management*, 56(5):1633–1644, September 2019. ISSN 0306-4573. doi: 10.1016/j.ipm.2019.04.006. URL https://www.sciencedirect.com/science/article/pii/S0306457318307428.

List of Figures

List of Tables

List of Abbreviations

BERT - Bidirectional Encoder Representations from Transformers
CNN - Convolutional Neural Network
CRF - Conditional Random Field
DTW - Dynamic Time Warping
FAUs - Facial Action Units
FC - Fully connected layer
FMI - Fowlkes-Mallows index
GMM-HMM - Hidden Markov Models and Gaussian Mixture Models
HKT - Humour Knowledge enriched Transformer
IPA - International Phonetic Alphabet
LSTM - Long Short-Term Memory network
MFA - Montreal Forced Aligner
MFCCs - Mel Frequency Cepstral Coefficients
MHD - Multimodal Humour dataset
MLP - Multilayer Perceptron
MUStARD - Multimodal Sarcasm Detection Dataset
NLP - Natural Language Processing
NMI - Normalized Mutual Information
NRC - National Research Council Canada
Passau-SFCH - Passau-Spontaneous Football Coach Humour
PCA - Principal Component Analysis
RNN - Recurrent Neural Network
SHEMuD - Sentiment, Humor, and Emotion aware Multilingual Multimodal Multiparty Dataset
Sitcom - Situational Comedy
SVM - Support Vector Machine
TBBT - The Big Bang Theory
TFiDF - Term Frequency-Inverse Document Frequency
VAD - Valence, Arousal, and Dominance
VideoMAE - Video Masked Autoencoder

A. List of sourced YouTube videos and channels

A.1 Russian

- Stand Up — Edwin Group
  - Stand Up 2020 Эдвин Багдасарян - сольный концерт Ему это не нравится 18+
    https://youtube.com/watch?v=E-BXpZAAnQ4
  - Stand Up 2021 Закрытый микрофон (август) Edwin Group — Stand Up
    https://youtube.com/watch?v=oJab2USfD5E
  - Stand Up 2021 Закрытый микрофон (июль 2) Edwin Group — Stand Up
    https://youtube.com/watch?v=aOuz5U-f8FE
  - Stand Up 2021 Закрытый микрофон (июль) Edwin Group
    https://youtube.com/watch?v=XeIKAh_edRo
  - Stand Up 2021 Закрытый микрофон (июнь 2) Edwin Group
    https://youtube.com/watch?v=uAJqSeurJLY
  - Stand Up 2021 Закрытый микрофон (июнь) Edwin Group
    https://youtube.com/watch?v=ztvgsaEo4gQ
  - Stand Up 2021 Константин Бутаков — сольный концерт
    https://youtube.com/watch?v=i3OTskskU1M
  - Stand Up 2022 Edwin Group Закрытый микрофон Выпуск 3
    https://youtube.com/watch?v=pZEYIvLAIYA
  - Stand Up 2022 Edwin Group Закрытый микрофон (март)
    https://youtube.com/watch?v=MtTPcm7aT7A
  - Stand Up 2022 Edwin Group Закрытый микрофон Выпуск 5
    https://youtube.com/watch?v=36N7m_LI9QQ
  - Stand Up 2022 Edwin Group Закрытый микрофон Выпуск 6
    https://youtube.com/watch?v=c4HM0o3o56s
  - Stand Up 2022 Edwin Group Леонид Кулаков - про Дальний Восток
    https://youtube.com/watch?v=GO9oUI4UIFo
  - Stand Up 2022 Edwin Group Сергей Агафонов «Добился всего» 18+
    https://youtube.com/watch?v=CBEYv-31wT8
  - Stand Up 2022 Edwin Group Сергей Агафонов «Ничего не добился» 18+
    https://youtube.com/watch?v=oFR9EfJPpHw
  - Stand Up 2022 Edwin Group Эдвин Багдасарян — Сольный концерт «Работать нужно над отношениями» 18+
    https://youtube.com/watch?v=G735DyFpHhQ

  - Stand Up 2022 Edwin Group Эдвин Багдасарян — про отца
    https://youtube.com/watch?v=iG4vqEUUljM
  - Stand Up 2023 Edwin Group Закрытый микрофон Выпуск 6
    https://youtube.com/watch?v=BazNLEDIoDI
  - Stand Up 2023 Edwin Group Закрытый микрофон Выпуск 8
    https://youtube.com/watch?v=To5OK76ncBw
  - Stand Up 2023 Edwin Group Ольга Егорчева «Я не знаю как надо» 18+
    https://youtube.com/watch?v=WuVlqggcQOA
  - Stand Up 2023 Импровизация Эдвин Багдасарян VS Руслан Гасанов
    https://youtube.com/watch?v=bEyTVx7PiSo
  - Stand Up Edwin Group 2021 Закрытый микрофон (август 2)
    https://youtube.com/watch?v=R8FObVhfcHk
  - Stand Up Edwin Group 2021 Закрытый микрофон (ноябрь)
    https://youtube.com/watch?v=WLC9AgRiOEU
  - Stand Up Edwin Group 2021 Закрытый микрофон (октябрь)
    https://youtube.com/watch?v=sWG7G7v3-UY
  - Stand Up Edwin Group 2021 Закрытый микрофон (сентябрь)
    https://www.youtube.com/watch?v=aE7VNxspBXs
  - Stand Up Festival 2022 Edwin Group
    https://youtube.com/watch?v=qmy1PXa5xP0
  - Stand Up Show Edwin Group «На стульях» 1 ВЫПУСК Stand Up 2022
    https://youtube.com/watch?v=5huyLH2Mof4
  - Stand Up Show Edwin Group «На стульях» 2 ВЫПУСК Stand Up 2022
    https://youtube.com/watch?v=lUJHH9QVhOE
  - Stand Up «На стульях» Edwin Group — 4 ВЫПУСК
    https://youtube.com/watch?v=UyjYUQVDVlI
  - Stand Up про анализы Эдвин Багдасарян
    https://youtube.com/watch?v=1_krz-aj7ao
  - Stand Up про казино в Вегасе и голую ругань Эдвин Багдасарян
    https://youtube.com/watch?v=MChJmJ1jLo0

- Денис Чужой
  - Stand Up Братья Стругацкие мужская красота воровские понятия
    https://youtube.com/watch?v=irt7999i1R8
  - «Второстепенный персонаж» (стендап-2019)
    https://youtube.com/watch?v=19Q60VzR84o
  - Дальше сам (Stand Up 2021) Денис Чужой
    https://youtube.com/watch?v=xT-IxupQJyo

- СЧАСТЛИВЦЫ
  - Алексей Колупаев — 91521 Стендап 2022 RUS+ENG SUBS
    https://youtube.com/watch?v=ZzNssZ6Mz50
  - Егор Котыченко — «Stand Up хиты под водочку» Стендап 2023 18+ (субтитры)
    https://youtube.com/watch?v=fOxpOFUntc8
- Ника Тарасевич
  - Ника Тарасевич Часть 1 stand up
    https://youtube.com/watch?v=ZVeSP5D0CWY
- ВСЕ СВОИ
  - СТЕНДАП КОМИК Лёха Лиховец Эффект Манделы (Беларускія English SUB)
    https://youtube.com/watch?v=2s11KJGfrys
- Илья Соболев
  - Стендап СОБОЛЕВА Целый час импровизировал с полным залом людей на неудобные темы
    https://youtube.com/watch?v=wWQ9tLsVMFo
  - Стендап СОБОЛЕВА Целый час смешил зал а потом заставил целоваться гостей
    https://youtube.com/watch?v=B5Tz7S742xs
  - Стендап Соболева на который нет билетов!!!
    https://youtube.com/watch?v=zC3Hj5QK4BU
- Сергей Молчанов | Stand Up
  - Сергей Молчанов Стендап импровизация Stand Up 2022
    https://www.youtube.com/watch?v=BYZwZtR02dY
  - Сергей Молчанов Stand Up про сиськи и дикпики
    https://youtube.com/watch?v=7bcmHVDSrXk
  - Сергей Молчанов Стендап про переезд в Москву Stand Up 2022
    https://youtube.com/watch?v=RH4xeJX3_xs
- Артём Ионов
  - СУД НАД НАВАЛЬНЫМ ДВОРЕЦ ПУТИНА ИНАУГУРАЦИЯ Артём Ионов Стендап в Самаре
    https://youtube.com/watch?v=Ygqu4OS8ry8
  - Стендап про Путина Лукашенко и всё происходящее Артём Ионов Стендап в Самаре
    https://youtube.com/watch?v=g0H892xmmbo

  - Стендап про самоизоляцию и всё вкусное Артём Ионов Стендап в Самаре
    https://youtube.com/watch?v=h8yzeB_S1do

A.2 English

- Comedy Central Stand-Up
  - A Lesbian with a Brazilian - Ashley Gavin - Stand-Up Featuring
    `https://youtube.com/watch?v=E7W3n2zB0Rc`
  - Angelina Martin Quit Weed But Never Quit the Culture - Stand-Up Featuring
    `https://youtube.com/watch?v=yQWI9WOI1jo`
  - Auguste White's New Revenge Personality - Stand-Up Featuring
    `https://youtube.com/watch?v=oYxF4I9kruk`
  - Bill Burr I'll Never Own a Helicopter - Full Special
    `https://youtube.com/watch?v=oedGGTLCSYo`
  - Chris Distefano I Just Drank Pinot Grigio and Listened to Michael Buble - Full Special
    `https://youtube.com/watch?v=-anx8VI02z8`
  - Chris Distefano Size 38 Waist - Full Special
    `https://youtube.com/watch?v=6rAGt1UD0wo`
  - Dating Is Too Expensive - Max Thomas - Stand-Up Featuring
    `https://youtube.com/watch?v=xvkAcFKLVVY`
  - David Drake Will Do Your Podcast - Stand-Up Featuring
    `https://youtube.com/watch?v=QtKgU-1ZA6s`
  - Decoy Nutella in My House- Naomi Ekperigin - Full Special
    `https://youtube.com/watch?v=N0rzyWbJH74`
  - Doing Mushrooms and Watching the History Channel - Skyler Higley - Stand-Up Featuring
    `https://youtube.com/watch?v=wDnZ9zQl2qY`
  - Don't Sleep Naked in New York - Godfrey - Full Special
    `https://youtube.com/watch?v=CwblshRlZz8`
  - Dulce Sloan I Was Forced to Move to New York Because of Success - Full Special
    `https://youtube.com/watch?v=l_LdiBC5tyc`
  - Dylan Sullivan Couldn't Wait to Inherit His Mom's Subaru Outback - Stand-Up Featuring
    `https://youtube.com/watch?v=VHwyRi48KqM`
  - Everything is Like An 8-Mile Moment - Daniel Simonsen - Stand-Up Featuring
    `https://youtube.com/watch?v=8ZEoeI-jVk8`

– Experiencing a Drive-by Queer Eye - Hoodo Hersi - Stand-Up Featuring
https://youtube.com/watch?v=X7YeBBZm0A4
– Hanna Dickinson's Dog Is the Drunk Version of Her - Stand-Up Featuring
https://youtube.com/watch?v=vj-LfLvs-jI
– Hannibal Buress Animal Furnace - Full Special
https://youtube.com/watch?v=AzcvtW3zj7I
– I Did My Best to Look Decent But I Ended Up Looking Amazing - Solomon Georgio - Full Special
https://youtube.com/watch?v=yBjr8KQgSDM
– I Didn't Even Know How Gay I Was - Sam Jay - Full Special
https://youtube.com/watch?v=dXZb9KlmDow
– I Got Electrocuted Trying to Eat Some Hot Dogs - Mia Jackson - Full Special
https://youtube.com/watch?v=k18vx1ychx0
– I Look Like a Senator's Nephew - Andy Haynes - Full Special
https://youtube.com/watch?v=iB3Q7j4Bbns
– I Smoke Weed and I Watch Nature Shows - Greer Barnes - Full Special
https://youtube.com/watch?v=xNuQzXU2D_A
– I Was Raised by a Pack of Lesbians - Jordan Jensen- Stand-Up Featuring
https://youtube.com/watch?v=8I1bapOz04c
– I'm Aaron from Tennessee a Pathological Liar - Aaron Weber - Stand-Up Featuring
https://youtube.com/watch?v=wE52zn5zqRw
– I'm Just Horny and Crying All the Time - Jenny Zigrino Jen-Z - Full Special
https://youtube.com/watch?v=PY9_iOjf2qk
– I'm Like the Bisexuality of Ability - Tina Friml - Stand-Up Featuring
https://youtube.com/watch?v=LbaK754C4Kk
– I'm Nervous Insecure and Squishy - Mark Normand - Full Special
https://youtube.com/watch?v=E3KN_Mhq3UQ
– I'm a Build-A-Bitch - Pink Foxx - Stand-Up Featuring
https://youtube.com/watch?v=5GvFc3zL8aU
– I'm a Vegan and I'm So Sorry - Julio Torres - Full Special
https://youtube.com/watch?v=UM_cKdv1ZR0
– I've Never Pulled My Jew Card Until Now - Eagle Witt - Stand-Up Featuring
https://youtube.com/watch?v=FoScKrPS1tc
– If I Have to Have Sex on One More Air Mattress... - Liza Treyger - Full Special
https://youtube.com/watch?v=Z8NXcTeiczo

– It's Never Too Early to Go Back to Bed - Martha Kelly - Full Special
https://youtube.com/watch?v=w1k48QoDJUI
– Jessi Klein I Would Like to Get Married Before I Get Herpes - Full Special
https://youtube.com/watch?v=-1rYlS8xys4
– Jim Gaffigan I'm Too Lazy - Full Special
https://youtube.com/watch?v=NaACTpYah1w
– Lying About Your Ethnicity as a Kid - Dauood Naimyar - Stand-Up Featuring
https://youtube.com/watch?v=pgdMBXKwsN8
– Mark Normand Don't Be Yourself - Full Special
https://youtube.com/watch?v=LHSTW5uPXBY
– My Brain Is Like a Radio DJ Who Does Not Take Requests - Emily Heller - Full Special
https://youtube.com/watch?v=xjP40RRc_Lc
– My Favorite 'Wheel of Fortune' Clip - Emmy Blotnick - Full Special
https://youtube.com/watch?v=oqG7KUhMJQo
– Natasha Leggero But You See I Reinvented Myself - Full Special
https://youtube.com/watch?v=OtHK_ilkRU4
– Nick Kroll Thank You Very Cool - Full Special
https://youtube.com/watch?v=kd-K-9faVfM
– Oh You're Building a Wall of Muffins Now - Vanessa Gonzalez - Full Special
https://youtube.com/watch?v=N3OLR6y-RmQ
– Patton Oswalt My Inner Child Doesn't Feel Like Chopping Wood Today - Full Special
https://youtube.com/watch?v=IuzCilEFBgo
– Rachel Feinstein Only Whres Wear Purple - Full Special
https://youtube.com/watch?v=rHaNv0X3iHQ
– Roy Wood Jr No One Loves You - Full Special
https://youtube.com/watch?v=UXUTrKd1syM
– Sam Morril Positive Influence - Full Special
https://youtube.com/watch?v=OUN_f7xKnpM
– Seeing Your Girlfriend's Dildo - Neel Nanda - Stand-Up Featuring
https://youtube.com/watch?v=AFqV-Mh4Ra8
– The Loudest B Slap Ever - Chris Tellez - Stand-Up Featuring
https://youtube.com/watch?v=vSkoSGsH4Oo
– The Number One Reason to Date a Short King - Caitlin Peluffo - Stand-Up Featuring
https://youtube.com/watch?v=Jcamfv9LER4
– Tourette's Is Sort of The Abstract Art of Disabilities - Benny Feldman - Stand-Up Featuring
https://youtube.com/watch?v=U8Rzd3AJUSY

– When Dirty Talk Goes Wrong – Tyler Groce – Stand Up Featuring
https://youtube.com/watch?v=jw2FjzPWqoM
– Which Bugs Are Gay - Jaboukie Young-White - Full Special
https://youtube.com/watch?v=q9LKzA3aBJE
– Why Gen Z Dating Advice Doesn't Work for Millennials - Jenny Zigrino - Stand-Up Featuring
https://youtube.com/watch?v=asY7CQJkUa8
– Why Is Live Theater Still Happening - Devin Field - Full Special
https://youtube.com/watch?v=3KNEWk5IK4A
– Why Ralph Barbosa Gave His Doctor a One-Star Review - Stand-Up Featuring
https://youtube.com/watch?v=Bq7O57JOFAM
– Why the Drama It's Just Pickles - Retta - Full Special
https://youtube.com/watch?v=oC_nGr4ZJCw
– Zach Galifianakis Who's the Boss Now - Full Special
https://youtube.com/watch?v=-a43xLs0AeI

B. Electronic attachments

B.1 Dataset structure description

Dataset files are hosted on a Google Drive `https://drive.google.com/drive/u/1/folders/1qhKsNl1B-yqPx80JqQflLInboVcRHaSV`.

The final dataset, which includes audios, audios after vocal removal, videos, original subtitles, labeled and resegmented subtitles and manually annotated random subset is located in the `dataset` folder. The folder is further divided into Russian and English parts as necessary. The files within each video have the same name for easy association. File `meta_data.json` contains meta information about each video - link to the video, channel name, duration, and technical description of the downloaded media files.

The labeled subtitle files contain a list of phrases, with each phrase accompanied by an audio span in milliseconds and a humorous label. Additionally, the token spans are provided, including their corresponding character spans and audio spans (See B.1).

B.1: Structure of the json item with resegmented subtitle


```
{
    "text": "Hello,  New  Orleans,  how  are  you?",
    "audio_span": [29.96, 31.45],
    "label": 1,
    "token_spans": [
      {
        "text": "Hello,",
        "audio_span": [29.96, 30.62],
        "text_span": [0, 6]
      },
      {
        "text": " New",
        "audio_span": [30.62, 30.68],
        "text_span": [7, 11]
      },
      {
        "text": " Orleans,",
        "audio_span": [30.68, 30.91],
        "text_span": [12, 21]
      },
      {
        "text": " how",
        "audio_span": [30.91, 31.12],
        "text_span": [22, 26]
      },
      {
        "text": " are",
        "audio_span": [31.12, 31.17],
```

```
            "text_span": [27, 31]
        },
        {
            "text": " you?",
            "audio_span": [31.17, 31.45],
            "text_span": [32, 37]
        }
    ]
}
```


Laughter annotations of validation videos (both ELAN files and the exported laughter tables) are in the `annotation` folder.

Forced-aligned subtitles and alignment validation is in `mfa` folder.

Files from the intermediate stages of subtitle preprocessing can be found in the `preprocessed_sub` folder, here is their description:

- `subtitles_faligned` - subtitles with word-level forced alignment and new segmentation (see example B.1 for the format)
- `subtitles_faligned_annotation_labeled` - humour labeled subtitles. Labels are based on the window approach and manually annotated laughter.
- `subtitles_cleaned` - preprocessed subtitles if text preprocessing like whitespace clean up, artifacts removal and auditory markers removal was applied
- `textual_laughter_markers` - folder with extracted laughter markers from subtitles, i.e. '[audience laughs]'

Folder `experiments` contains results on different experiments with laughter detection:

- `detected_peaks` - peak detection results on the subtracted channels audios for each threshold parameter
- `clusterization_experiments` - results of hyper-parameter search for clusterization filtering
  - `clusterization_labels` - clusterization labels
  - `clusterization_logs` - clusterization hyperparameter search results
  - `clusterization_plots` - plots of PCA-reduced clusterizations
  - `peaks_features` - extracted audio features from Audio Spectrogram Transformer for each detected peak
  - `reverse_labels` - manually determined whether to reverse clusterization labels (turn 0 to 1 and 1 to 0)
- `ml_whole_experiments.json` - hyper parameter search for machine learning approach of laughter detection on the whole audio
- `ml_window_experiments.json` - hyper parameter search for machine learning approach of laughter detection on the inter-subtitle window spans in the audio

- `channel_subtraction_results.txt` - list of successful, non-successful and failed audios after channel subtraction, labeled manually

Models checkpoints and cross-validation logs are in the `models` folder.

The extracted features files for the SVM multimodal classification (Section 3.2) are organized in the `features` folder. These files contain the extracted features for the context-utterance splits of subtitle phrases, where each split consists of 4 context phrases and 1 target utterance. The following is a description of the different feature files:

- `bert_features` - This folder contains BERT embeddings. The embeddings are stored in the `embeddings.npy` file, and the size of each feature vector for one split is 768. The video names are sorted, and the features from each video are concatenated. Token information with the offset mappings can be found in the `tokens.jsonl` file.
- `video_features` - This folder contains video features extracted using the VideoMAE model. Each split has an output size of (1568, 768), and there is one file for each video.
- `openface_features` - This folder contains the extracted features from OpenFace. Each video has its own file, following the standard OpenFace output format.
- `facial_mean_context_utterance_features.npy` - This file contains the mean vectors of the OpenFace features for each context-utterance split. The mean values of the context frames are concatenated with the mean values of the utterance frames, resulting in an output vector size of 142. The features from all videos are concatenated.
- `video_mean_features.npy` - This file contains the averaged video features along the first axis. The output vector size is 768, and the features from all videos are concatenated.

Useful plots are in the `plots` folder.

B.2 Scripts description

Scripts are available on GitHub[1]. The scripts are documented with argument details to assist reproduction. Here is the description of their functionality.

B.2.1 Dataset Collection, Preprocessing, and Laughter Detection Labeling

The scripts for dataset collection, preprocessing, and laughter detection labeling can be found in the `dataset` folder:

- `collect_videos.py`: Crawls and downloads videos, audios, and subtitles from YouTube.

[1] `https://github.com/kuzanna2016/multimodal_humour`

- `preprocess_subtitles_text.py`: Preprocesses subtitle text by removing artifacts, cleaning up white space and punctuation, and reconstructing censored words.
- `prepare_for_mfa.py`: Prepares TextGrid files for forced alignment with MFA, including converting numbers and characters into full written form.
- `resegment_and_word_align.py`: Aligns forced aligned words with tokens and resegments subtitles.
- `extract_textual_laughter_markers.py`: Extracts spans of subtitles containing auditory laughter markers like `[audience laughs]`.
- `label_with_textual_laughter_markers.py`: Labels humor based on textual laughter markers.
- `swear_words_rus.py`: Regular expressions to replace censored swear words.
- `replaced_swears.json`: Dictionary with replaced swear words for English.
- `numeric.py`: Auxiliary functions for working with number conversion.
- `laughter_detection`: Laughter detection experiments for the machine learning approach. Should be run inside the `laughter-detection` project[2] after installing its requirements:
  - `label_with_annotation.py`: Labels validation videos with manually annotated laughter.
  - `laughter_detection_model.py`: Sets up laughter detection model.
  - `laughter_detection_experiments.py`: Runs hyperparameter search for the laughter-detection model.
  - `label_with_laughter_detection.py`: Labels videos with laughter-detection results.
  - `vocal_remover.py`: Runs the vocal-remover model. Should be run from inside the `vocal-remover` project[3] after installing its requirements.
- `peak_detection`: Laughter detection experiments for the peak detection approach
  - `subtract_channels.py`: Subtracts channels in audios.
  - `extract_audio_features.py`: Sets up the AudioTransformer model for audio features extraction.
  - `peak_detection_experiments.py`: Runs hyperparameter search for the peak detection threshold.
  - `clusterization_experiments.py`: Runs hyperparameter search for clusterization.

[2] `https://github.com/jrgillick/laughter-detection`
[3] `https://github.com/tsurumeso/vocal-remover`

  - `clusterization_validation.py`: Validates clusterization on annotated videos, needs manual cluster label check.
  - `plot_clustering.py`: Plots clustering results on PCA-reduced points.
  - `const.py`: Contains video names, that were annotated as successful after channel subtraction.

B.2.2 Feature Extraction for Multimodal SVM Model

The feature extraction scripts for the multimodal SVM model are in the `feature_extraction` folder:

- `extract_video_features.py`: Extracts video features using VideoMAE.
- `extract_bert_features.py`: Extracts textual features using BERT models.
- `extract_open_face_features.py`: Combines extracted OpenFace features.

B.2.3 Humour Detection Models and Training Scripts

Humour detection models and their training scripts can be found in the `models` folder:

- `bert.py`: Contains BERT-based models set-up.
- `colbert.py`: Contains ColBERT architecture.
- `text_train_validate.py`: Contains training and validation for the textual models.
- `multimodal_train_validate.py`: Contains training and validation for multimodal setting.

B.2.4 Useful Plotting Scripts

Useful plotting scripts can be found in the `plotting` folder:

- `plot_audio_with_annotated_laughter.py`: Plots audio waveforms with annotated laughter and other spans (detected peaks, subtitle segmentation, word segmentation).

B.2.5 Other Files

Other files include:

- `utils.py`: Contains data processing functions and common functions for validating.
- `requirements.txt`: Lists the requirements to run the code.