rijksuniversiteit groningen / campus fryslân

Towards Automatic Speech Genre Synthesis

Proposing a Framework for Speech Function Recognition and Speech Register Synthesis

Hubert Matuszewski

University of Groningen

Master's Thesis

To fulfill the requirements for the degree of
Master of Science in Voice Technology
at University of Groningen under the supervision of
dr. V. Verkhodanova

Hubert Matuszewski (s3992756)

November 19, 2025

Contents

Abstract

Recent progress in speech synthesis have enabled the generation of synthesised speech which has a more natural, human-like sound. Expressive speech synthesis has made progress in adding emotion and speaker specficic characteristics to synthesised speech. However, there has been no research done on the addition of prosody to speech synthesis based on the kind of text being used for synthesis, and the kind of speech patterns which would correspond to a given category of text.

To that end, this paper seeks to synthesise Speech Genres through the combination of speech synthesis and text classification architectures. Certain text is assumed to have a specific Speech Function, which entails having to speak the text with a particular Speech Register. The combination of Speech Function and Speech Register is a Speech Genre. This paper defines 4 examples of speech Genres: Documentary, News Report, TEDTalk, and Comedy Stand-Up.

An RCNN was used as a Speech Function Classifier, with text data being gathered for each genre by means of web scraping relevant sources. Speech Register synthesis was split into text-to-speech synthesis, which is executed by FastSpeech 2, and Speech Register synthesis, which is executed by kNN Voice Conversion. The audio data was also gathered by means of web scraping various sources.

The Speech Genre output was assessed through human evaluation, with the main experiments being whether the kNN Voice Conversion output was preferred over the standard FastSpeech 2 output, and whether the Speech Genres were discernible by the prosodic characteristics of the Speech Register. There was a general tendency towards preferring the kNN voice conversion (average MOS = 2.728) output over the FastSpeech 2 output (average MOS = 2.487). The speech genres showed a higher average discernability accuracy (35.17%) than random chance (25% with 4 genres), but the results were also not convincing. However, evaluator MOS scores indicate a slight preference for audio samples with a matching Speech Register as opposed to samples with a mismatching Speech Register (average MOS = 3.152 and 3.015 respectively). Additional findings include that RCNN accuracy is improved by using training data which is shorter than testing data (a maximum F1-score of 0.94 compared to the highest score with test and train data of equal length, which is 0.901) and that evaluators tended to prefer samples with a male voice as opposed to a female voice (55.4% preference for male samples vs. 28.6% for female samples).

Additionally, the paper proposes a metric of "discernibility" as a means of testing whether human evaluators are able to distinguish between Speech Registers, offering different approaches of implementation. This research paper is limited by means of a low sample size of human evaluators, using a self-created text and audio dataset, limited comparable studies for result interpretation, the use of a novel theoretical framework which has not seen broader application, and the subjective definition of each of the genres devised for this research paper. The concepts proposed in this paper can also be applied in other research areas, particularly the use of text classification for emotion synthesis, accent synthesis, and potentially sarcasm detection.

The demonstrator (architecture) and data used in this research paper are available at:
`https://github.com/585hubert/Genre_Synthesis`

Glossary

Speech Genre - A categorisation of speech based on the type of function the text holds, and the manner of speech which is associated with the function. A Speech Genre consists of two entities, a Speech Function, and a Speech Register. This is a definition specific to this paper, but stems from essays and works by Mikhail Bakthin (Bakhtin, 2011).

Speech Function - A piece of text from which the characteristics are derived from a specific context and application, which usually bear socially agreed upon conventions. It is generally consistent on the level of thematic content, but may differ in terms of individual aesthetic preferences. This is a definition specific to this paper, but is largely based on how the term is used in sociolinguistics (e.g. Holmes, 2013).

Speech Register - A manner of speech which is governed by its functional communicative context, taking into account both the function of the given speech, alongside any stylistic choices made by the speaker. This definition is a specification of what is discussed by Egbert & Mahlberg (Egbert and Mahlberg, 2020).

Text Classification - The task of determining the category of a text within a predefined set of categories (Minaee et al., 2021). This set of categories can come from pre-established frameworks, or can be devised heuristically. A Text Classifier is subsequently an entity, usually a machine learning architecture, which executes the task of Text Classification.

Speech Synthesis - The task of synthesising a speech signal (Taylor, 2009). Within this thesis, all Speech Synthesis tasks will be Text To Speech tasks, as opposed to alternatives such as Synthetic Speech Generation. As such, Speech Synthesis intakes a text input, and outputs an audio signal. A Text Classifier is subsequently a machine learning architecture which executes the task of Speech Synthesis.

Synthesised Speech - Speech which is generated by artificial means, predominantly through Speech Synthesis. This is in contrast to **human speech**, which refers exclusively to natural speech produced by humans.

1 Introduction

Speech Synthesis (SS) has become fairly ubiquitous in daily life in multiple environments. Whether it be through the use of personal voice assistants such as Alexa, or intercom systems on public transportation, or an automatic response system in a customer services branch of a firm; one is likely to have some interaction with a synthetic speech system.

Aside from broad applications, SS can yield accessibility benefits to many different groups. People with visual impairments can benefit from a system where their surroundings are communicated to them by means of a synthetic voice (Freitas and Kouroupetroglou, 2008). People with reading disabilities such as dyslexia can benefit from having text read to them while trying to parse through the text (Dawson et al., 2019). People with voice pathologies of varying severity can benefit from either having their voice augmented (if one can only muster a whisper for example) or entirely synthesised (the famous example being the KlattTalk system (Klatt, 1982) behind Stephen Hawking's voice). Recent research has focused towards contextual appropriateness; the notion of incorporating user needs, expectations, and preferences when devising a given SS system (Wagner et al., 2019).

Consider two examples of synthetic voices. First, a generic voice assistant tasked with responding to fairly basic commands, such as playing music, making a quick online search, appending an item to a shopping list, and so on. The convention is to have such an SS system be "clear and pleasant" (Wagner et al., 2019). It is responding to commands, so it makes sense that the demeanour with which it speaks is one which is not irritating, doesn't show excessive emotion, and prioritises being understandable to the user which made the initial request. Contrast such a voice assistant with a video game voice, which can be assumed to require expressiveness and personality (Wagner et al., 2019). This again makes sense as otherwise, voice actors within video games or animated television shows would be redundant.

Certain studies have also looked into directly implementing SS with contextual appropriateness, such as SS systems for people with vision loss (Podsiadło and Chahar, 2016). The point here, is that across a multitude of contexts, different features have different levels of importance, thus requiring the field of SS to be subsequently robust to different modes of speech output to accommodate this multitude of context.

A research direction which could accommodate the requirement of expressiveness and personality is Expressive Speech synthesis (ESS), which concerns itself not only with the synthesis of speech from text (as in, the simple carryover of linguistic information), but also in conveying additional para-linguistic information such as emotion, speaking style, attitudes (Zhu and Xue, 2020), and potentially other aspects such as accents, speech disfluencies and many others. One specific application of ESS is Emotional Speech Synthesis (K. Zhou et al., 2022).

The emergence of new applications of ESS forms the basis of the topic of this thesis. Instead of looking at emotional speech, this thesis seeks to look into what will be referred to as "Speech Genres" (see chapter 2), which pertains to specific types of speech which are spoken within a specific context. Examples of functional speech are given below:

- A news reporter, when delivering the news, has a very specific way of delivering the news. While a part of conventional read speech (read from a teleprompter), there are particular cues (aside from linguistic content) which point to the read speech specifically being a news report. Cotter, 1993 mentions specific features such as accentuated pauses, pitch salience (identifiable pitch traits), but also broad objectives such as speaking with warmth, ease and authority.

- A sports commentator doesn't merely communicate what is visible to someone watching the sport, but also adds emotion to their speech to either get viewers excited, or out of the commentator's own excitement. Not only that, but the extent of this varies wildly between sports (consider a golf commentator vs. an F1 racing commentator). Kern (Kern, 2010) identifies particular commentary instances of "building up suspense" and "presenting a climax" within football commentary, both of which have separate prosodic features to instances when commentators give broader analytical information about a game.
- Documentary narration not only has the job of communicating the linguistic information of the topic being presented, but must also do it in a captivating manner which is pedagogically and expressively robust. Consider the success of David Attenborough in narrating various wildlife documentaries. In theory, anyone could read the same script, but it is specifically the execution which makes Attenborough so revered.

Early SS systems, such as KlattTalk (Klatt, 1982) had to focus entirely on the accurate pronunciation of words, lacking the resources to address other factors such as intonation, emotion, and other such paralinguistic features. This left such systems with an output that sounds robotic (Schreibelmayr and Mara, 2022). If the aforementioned examples were to be synthesised, they would stand to benefit from sounding like their human speech equivalent, rather than having a robotic delivery. Robotic sounding synthesised speech has been shown to be less preferred than synthesised speech which sounds more human. From this, I motivate the topic of synthesising contextual speech by means of identifying applying a specific manner of speaking. This entails two tasks. The first is to identify a Speech Function given a piece of text. For example, a text such as *"2 police officers were shot yesterday near the convenience store"* would be classified as a news report. Apply the appropriate expression so that the above text is read "like a news reporter would", as opposed to a flat delivery, an emotional speech, or some other form of speech.

The task of identifying a speech function and applying the correct expression is accomplished by a deep learning architecture, which is subsequently the demonstrator of this thesis project. This results in two general aims of this paper, to develop an adequate text classifier and an adequate expressive speech synthesis system. Both of these aims, and what constitutes sufficient adequacy, is elaborated upon by the research questions in section 2.3.

Thesis Outline

This paper first briefly examines the relevant literature (2). From this, the research questions and hypothesis are stated. Subsequently, the theoretical framework and architecture (3) section describes the problem addressed by the paper, alongside the architecture used to tackle it. The methods section (4) describes the data acquisition and how the trainable architecture was trained. The experimental setup section (5) goes over how the samples were created, and how each of the research questions were tackled. Finally the results section (6) gives the relevant results, and the discussion (7) provides a brief discussion and conclusion regarding the results and further directions of research.

2 Literature review

The introduction alluded to speech genres and as such, the literature review begins with disambiguating the concept further; looking into terms such as "genre", "function", "register", and "style". Since we are looking to perform the tasks of text classification and speech synthesis, the literature review examines these two elements accordingly. The research question and hypotheses follow thereafter.

2.1 Speech Register & Function

Here, we analyse the pre-existing definitions of speech genre, function and register (alongside other aliases) and motivate the specific definitions put forward in the glossary. When looking at functional speech literature, there are a few terms that bear similar meanings and are used interchangeably.

The first term to be examine is "Speech Styles". This term appears within sociolinguistics, with the term being coined at least as far back as 1975 (Giles and Powesland, 1975). One particular mention of speech styles (Erickson et al., 1978) frames the term as a variation of natural speech. Specific parameters and descriptors are given, such as use of intensifiers (a word which strengthens the meaning of another word), hesitation forms (words, utterances and disfluencies which arise when a person is either thinking or searching for a specific word) alongside other linguistic features (Erickson et al., 1978). The paper goes on to make a categorisation of "powerless" and "powerful" speech styles, which goes on to be investigated for effectiveness in court settings (Erickson et al., 1978). While this paper does not use any of those defined speech styles, it provides a useful example or attempting to categorise forms of speech by certain linguistic properties. A more recent use of Speech Styles comes from a study examining speech between a parent and child (Ramírez-Esparza et al., 2014); with a distinction between standard and "parentese" speech (the kind of speech a child would hear from their parent) from "standard speech" (Ramírez-Esparza et al., 2014), and a distinction of social context (1 to 1 versus a group setting). While this paper also uses linguistic differences in speech to distinguish styles, it also works with the assumption that speech styles can be influenced by a social context. In speech synthesis literature, speech styles are described (e.g Wang et al. (2018)) as a type of speech rich in information, influenced by a given speaker's choice in intonation and flow (Y. Wang, Stanton, Zhang, Skerry-Ryan, et al., 2018a), alongside referring to affective prosody (which are variations in pitch, loudness, rate and rhythm). In this case, a larger importance is given to prosody (features outside of direct linguistic context which give additional meaning) than the specific linguistic features or social contexts. However, this term is not compatible with the aim of this paper. While these papers have different nuances and applications, they all share a commonality in defining speech styles to be person-specific. A speech style is not something which is taught or mandated, but rather acquired naturally, and is a set of linguistic characteristics specific to a given person. Since the introduction alludes to specific types of functional speech spoken within a specific context, we require a term which is independent of a person's habits or preferences. For example, we would distinguish the style of David Attenborough delivering a documentary narration from the general concept of documentary narration.

A similar term is "Speech Register", which is defined by Egbert & Mahlberg as a type of text which carries a particular situational context, alongside particular linguistic features (Egbert and Mahlberg, 2020). An earlier sociolinguistic definition is provided by Halliday et al. (Halliday et al., 1964) which describes registers as different varieties of speech which people explicitly choose from and use. A comparison is given through dialects, which are framed as a variety according to the user; and registers which are frames as a variety according to usage. A subsequent paper (Weeks, 1971) provides some examples of different registers, such as whisper, clarification (degree

of slowness and enunciation), and grammatical modification (changes in the form and/or structure of sentences). A key distinction between speech styles and speech registers is that speech registers are intentional, they are a deliberate alteration of language to suit a particular context. With the examples of news reporter and sports commentator given in the introduction, speech registers are a more reasonable term to use as both of these formats entail speech by a commentator or reporter which is explicitly, and intentionally, distinct from their own private speech. Referring to speech styles and speech register, Kortmann (Kortmann, 2020) makes an attempt of distinguishing the two:

> *"...register choices are primarily determined by the functional-communicative context, while stylistic variation is more determined by individual choices and aesthetic preferences and thus is less predictable"* (Kortmann, 2020, p. 203).

Another term to distinguish types of speech is "Speech Genre", which is formulated by Bakthin (Bakhtin, 2011) as two entities; unmediated speech, and complex speech such as academic writing, novels, dramas, and so on. Genres correspond to a particular situation of speech communication, where genres consist of utterances, which are subsequently denoted by their thematic content, style and compositional structure (Bakhtin, 2011). Dementyev (Dementyev, 2016) elaborates on the concept further by framing it as a theory of "practice patterns of verbal communication" (Dementyev, 2016, p. 103). It is explicitly framed as a form of communication separate of the communication, but also as being a standardised speech form (Dementyev, 2016). Specific descriptors of speech genre provided by Dementyev are intentional factor, style manners of beginning and ending a speech, and strategies and tactics of communication (Dementyev, 2016). This framing elaborates on speech registers by acknowledging a fundamental structure and approach to speech. From the introductory example of a news reporter, the genre of a news report would, under this framing, be guided by the strategies and tactics (to inform by means of a specific medium), style (formal), presence of beginning and conclusion (headline of a news segment, structure of discussing a news story, etc.), among other categories mentioned. A speech genre is thus a fitting description for both the initial text or draft of a speech, and the realisation of said text by a given person. Recalling the discussion of speech register, we can assign this realisation of text as a speech register, as it is an intentional means of speaking to suit a particular context.

Finally, a "Speech Function", from a sociolinguistic perspective, refers to the variety of utterances used in discussion, used to either convey information or express social relationship to others (Holmes, 2013). It is also described as a communication technique to transfer ideas (Van Thao, 2022). Since this paper deals with speech which almost always carries a text counterpart to it (such as a teleprompter for a news broadcast, a script for a documentary, etc.), it follows that the choices of utterances and general communication technique are also contained within the text itself, whilst also being devoid of either speech variety (a text can be spoken through different registers) and usually does not inherently contain style aspects as described through variations of natural speech (Erickson et al., 1978). The text does also contain the thematic content, a specific beginning and end, and tactics of speech, which would present it as a speech genre. A speech function would thus materialise through the initial text to be read, as it is separate from the spoken element.

In this paper, Speech Register (*SR*) is treated as the realisation of Speech Function (*SF*). Using the example of news reporting, the text read on the teleprompter would have a speech function of "news report", whereas the manner by which the news reporter reads the text would be the speech register applied to the speech function. To simulate Speech Register, the synthesis of speech is required alongside the synthesis of the specific prosodic variation that a person would use when applying a register to their speech. This necessitates choosing a Speech Synthesis architecture, and an architecture which is able to learn and generate speech registers. The Speech Function is

defined to be a category of text from which a technique, context and structure is found; the identification of which requires a Text Classification architecture. In order to distinguish the synthesis of speech genres from speech styles, the text and audio used for training should come from multiple sources/persons, to avoid overfitting a speech genre to one specific individual.

2.2 Speech Synthesis

Speech Synthesis (SS) refers broadly to any production of speech which is artificial; produced electronically as opposed to the human vocal mechanism. It is often described as a decompressing problem; where you take a compressed form of data (text in this case) and try to decompress it through adding various parameters which are not explicitly within the compressed form (e.g. adding prosody, selecting the fundamental frequency, choosing the volume, and so on). Modern implementations of SS are rooted within machine learning; deep learning specifically. As opposed to defining dictionaries which dictate how a certain word ought to be articulated; modern SS relies on modern computational power to create Neural Network based models which learn how to generate speech from studying large amounts of data.

One example worth considering in this regard is Tacotron (Y. Wang et al., 2017). Tacotron sought to overcome the issue of the pipelined structure of older SS methods. Beforehand, an SS pipeline required various components to handle different parts of speech; such as how long different sounds (or silences) need to be, how to accurately map text into the correct phoneme (whilst trying to handle contextual variations of word pronunciation), how to actually produce speech from these analyses (vocoding), among many other components (see Paul Taylor's book on Speech Synthesis for an in-depth look (Taylor, 2009)). Regardless of whether the chosen manner was concatenative synthesis (bridging every sound fragment together sequentially), statistical parametric synthesis (speech through the generation of averaged acoustic parameters), or any other method; a common problem was that each module was domain specific and required independent training. This not only led to reduced interoperability between different modules, but also meant that any error generated by one of the modules would be propagated across the entire pipeline.

One of the modern types of Speech Synthesis (SS) architectures are encoder-decoder/transducer architectures (see Bataev et al., 2025, Nallabala et al., 2025, Adibian and Zeinali, 2025). In essence, an encoder-decoder seeks to take some input, encode it into some compressed, latent representation, only to be decompressed by the decoder into some desired output. In SS, the input would be text (or some processed form thereof) and the output is either the direct audio of synthesised speech, or some intermediate output which can be further processed into speech. Current implementations of speech synthesis usually come within either specific contexts, or architecture optimisation (examples: Meng et al., 2025, Y. A. Li et al., 2025, H. Wang et al., 2025, Ye et al., 2025, Chen et al., 2025). Most of these approaches make improvements through either an improved means of data augmentation and architecture evolution (Chen et al., 2025), optimisation of integration with large language models (Ye et al., 2025, H. Wang et al., 2025), incorporation of style vectors for prosody prediction (Y. A. Li et al., 2025), and improved tokenisation techniques (Meng et al., 2025). While there are improvements within architecture composition, there isn't necessarily a large amount of progress in trying to expand the scope of speech synthesis input. An exception ot this trend is a paper by Niu et al. (Niu et al., 2025) which uses a "face based synthesis" system; which effectively guesses the vocal characteristics of a given synthesised speaker based on the image of a face, alongside "text prompt synthesis", which can be interpreted as a set of arguments given to guide the architecture in determining the vocal characteristics of a synthesised speech sample (e.g. if a voice is supposed to sound male/female, exhibit emotion, etc.). This paper aims to do something similar, but instead of providing characteristics based on a prospective speaker, the type of text (the speech function) is the characteristic which determines the characteristics to be synthesised. The gap in research to be derived, is that none of these (or any other) papers

have attempted to use the classification of the text input as a parameter for applying prosody to a synthesised speech sample. Whereas most papers are concerned with matching the characteristics of a given person or voice, this paper is concerned with matching the characteristics of a group of people which have a shared Speech Register (for example, they are all news broadcasters).

In almost all of these cases, the papers behind speech synthesis architectures report higher mean opinions scores (MOS) or other similar variants thereof, lower word error rate (WER) through applying automatic speech recognition on the synthesised output, higher speaker similarity scores and higher PESQ (Perceptual Evaluation of Speech Quality) scores compared to previous papers. This shows that speech synthesis technology is constantly improving by means of architectural upgrades, implying that any new architecture may be rendered outdated within a year or two. This subsequently motivates the foregoing of this paper to committing to a specific architecture. Rather, this paper presesnts a general architectural framework by which speech synthesis systems can be used in the production of speech genres (see the Theoretical Framework, section 3). Given the specifications behind speech genres and registers as described in section 2.1, we can consider focusing on speech synthesis systems which are specifically focused on improving the prosody behind synthesised speech.

Expressive Speech Synthesis

From the Speech Style and Function section, many of the terms and definitions highlight the importance of features which go beyond semantic meaning, such as prosody, intonation, flow, etc. Expressive Speech Synthesis (ESS) is a domain of SS which serves "... to colour [speech] with inflections that cover the same range of affective expressions that humans are capable of" (Triantafyllopoulos and Schuller, 2025, p. 1). As such, ESS is chosen as an adequate research path with respect to synthesising speech registers, as the goals of ESS align with the goals of Speech Genre synthesis (instil specific expressions onto synthesised speech). Though ESS has seen large improvements, there are still notable shortcomings when applied to emotive speech. While emotional speech can be produced, it is done from a small amount of basic categories. The output thus lacks the subtlety and complexity seen in human speech (Zhu and Xue, 2020). Additionally, when designing ESS, the initial training is done on emotionally neutral speech, only to add emotion adaptation afterwards. This results in models which have a poor choice in emotions, and an output that fails to adequately output the rich, expressive information found in human speech (Zhu and Xue, 2020). For emotional speech synthesis, the proposed solution is to implement a continuous variable control for emotional strength (Zhu and Xue, 2020). Such advice can also be adapted to controlling Speech Registers (in lieu of emotional strength) by using the Speech Function as the control variable.

Skerry-Ryan et al. (Skerry-Ryan et al., 2018) propose a "reference encoder" to capture a prosody embedding, and an "embedding lookup" to handle speaker specific embeddings. A reference encoder, in ESS, refers to an encoder which is tasked with compressing the prosodic information of a speaker into a compact, latent form, which can be passed onto a decoder along with other embeddings to be decoded into speech. After training an SS model, the reference encoder would learn various ways of applying prosody to a given utterance, which could be adjusted in a controllable manner (e.g. through labels) (Skerry-Ryan et al., 2018). While the notion of a reference encoder is a useful one, the focus, as with previous papers, is on the emulation of a specific speaker rather than a speech genre. Thus while it is not directly applicable to the objectives of Speech Genre synthesis, the premise of a reference encoder is a useful one, as it is possible to encode the pattern of a given genre and then apply that encoding onto a synthesised sample. Wang et al. (Y. Wang, Stanton, Zhang, Skerry-Ryan, et al., 2018b) build on the "reference encoder" of the previous paper through the use of "global style tokens", which can either be explicit labels given to certain data (for example, an "upset" token with accordingly upset sounding training data) or learned labels

where the model learns to classify different styles in an unsupervised fashion. The unique characteristics of each token are represented through a variation of frequency and energy distribution on the synthesised speech samples based on a given token (Y. Wang, Stanton, Zhang, Skerry-Ryan, et al., 2018b). An important point raised in this paper is the notion that objective metrics do not necessarily reflect the user perception of synthesised speech (Theis et al., 2016). This concept forms a solid basis for the objectives of this paper; each genre can be represented as a kind of token to be applied to a speech sample (where "style" would somewhat confusingly be not the individual style of a speaker, but the collective register of a group of speakers) This paper accordingly elects not to use any objective metrics when analysing speech synthesis output. Li et al. (X. Li et al., 2021) propose using a multi-scale reference encoder. Multi-scale refers to the reference encodings being performed not only for one time interval, but rather a few (in this case there are two scales, a global level spanning a whole utterance, and local level, which are closer to the phonetic level of prosody). The local level representation is paired with text input in an attention mechanism, the result of which is fed into an attention based decoder alongside the global prosody vector. This is also a nice addition to speech synthesis systems as it enables a wider range of expression between different speech styles. Variations in prosody between a news broadcast and a TEDTalk may not be as large as variations in prosody between different emotional speech samples (e.g. angry yelling versus a sad cry). Speech Genre synthesis could thus potentially stand to benefit from a multi-scale reference encoder approach.

Most of these improvements lead to higher Mean Opinion Scores (MOS), which can be interpreted as a more pleasant response to synthesised speech with the improvements compared to architectures without them. When examining the mel-spectrograms of synthesis using reference encoders and/or style tokens, they tend to match the mel-spctrograms of emotive and/or expressive speech better than architectures without such modifications. The pursuit of more robust ESS is very active, and newer and better architectures are constantly being developed. Such advancement would mean that the proposition of a fixed architecture to synthesise speech registers would potentially succumb to the issue of becoming outdated by superior future architectures.

The speech synthesiser chosen for this paper is the k-Nearest Neighbour Voice Converter (kNN-VC, Baas et al., 2023). A detailed breakdown of the architecture (including structure, hyperparameters and so on) is given in the methodology section (section 4). The premise of the architecture is to generate output of a particular person by means of storing audio samples of that person as a series of vectors, which transform an input audio sequence into synthesised speech which resembles the speech to be synthesised. Specifically, the parameters are derived as an average of the n closest (by means of cosine similarity) vectors to the input audio vector. We can manipulate this architecture by, instead of focusing on a single voice, including a series of speakers of a given speech genre. The average value of an n speakers from such a pool of vectors would now represent the average Speech Register rather than the style or vocal characteristics of a given speaker, which is aligned with the research objectives of this paper.

The main focus of this paper is to harness the expressive power of the most recent ESS architectures; but instead of deriving the "style" labels from a lookup table, audio, or explicit tokens, the synthesis of different registers is done by deriving a label from the actual text itself. For this, we need to examine Text Classification.

Text Classification

Text Classification (TC) refers to the task of labelling a given piece of input text. This task can be performed at various scales, ranging from sub-sentence structures all the way up to entire documents. There are many ways in which pieces of text can be classified. One common example is spam detection, where the task is to determine whether a given email constitutes spam (unwanted

content which may be a nuisance and/or security risk). Other tasks within Text Classification include Sentiment Analysis (ascribing an emotion/opinion towards a text), Categorisation (assign a topic/category based on labels or user preference) among others (Minaee et al., 2021). Similarly to SS, modern TC is rooted in deep learning methods, such as Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), Transformer Architectures (e.g. RoBERTa, Cai and Ye, 2023), LLMs (Large Language Models, e.g. Zhang et al., 2025) among others. It is not uncommon for accuracies of over 90% to be reported; and often, even simple architectures such as Naive-Bayes can achieve similar accuracies to even the most state-of-the-art models (Hunter et al., 2023). For more complex tasks within TC such as Sentiment Analysis, the results are usually lower, between 60-85% depending on the method and data (e.g. Hartmann et al., 2023). Therefore, the specific choice of Text Classifier is not especially important, as most methods can be deemed to have an acceptable accuracy (Taha et al., 2024 (although it fails to quantify what terms like "acceptable", "good", and "satisfactory" mean), Q. Li et al., 2022).

The Text Classifier chosen for the task of Speech Function identification is the RCNN as presented in Lai et al. (Lai et al., 2015). A detailed breakdown of the architecture (including structure, hyperparameters and so on) is given in the methodology section (section 4). It is chosen as it the RCNN performed well (95% accuracy) on the Fudan University dataset[1], which is a dataset of Chinese documents divided into 20 different categories. This categorisation is quite similar to the kind that would be performed when distinguishing Speech Functions. While there is a difference in language between the implementation of Lai et al. (Chinese) and the implementation of this paper (English), it is reasonable to assume that with the ubiquity of English text data, such an architecture could achieve similar, if not better, accuracy. One issue not addressed by Lai et al. is whether the character length of the dataset has any effect on model accuracy. Accordingly, varying character length entries are tested in this paper (see the Text Classification Pre-Processing (section 4.3.1) and RCNN Training & Testing Setup (section 5.1) sections on the specifics). Word length information was not included, thus it is not investigated. As was mentioned, it is possible to include other RCNN architectures for this task which could also potentially have equally high accuracy. The RCNN is simply a fitting choice.

2.3 Research question and hypothesis

Putting the literature review together, we can conclude that speech genres should be considered as 2 parts, the Speech Function which is derived from the text, and the Speech Register which is the manner by which the text is subsequently spoken. Since Speech Registers are assumed to require specific prosodic features, it was concluded that an Expressive Speech Synthesis approach would be required to produce Speech Registers. The task of identifying Speech Functions was concluded to require a Text Classifier. The use of Text Classification on text to be synthesised to enhance the output of Speech Synthesis is the novelty proposed in this paper.

Having discussed the general topic and relevant literature, the following research questions are put forward:

> **Research Question 1:** To what extent, as quantified by accuracy, F1-scores, and MOS, can the use of Text Classification through an RCNN (Lai et al., 2015) and Voice Conversion through kNN Regression (Baas et al., 2023) improve the output of Fast-Speech 2?

> **Research Question 2:** Can Text Classification by means of an RCNN (Lai et al., 2015) produce Speech Genres that are distinguishable by human evaluators?

[1]The link provided in Lai et al. no longer appears to work, and the dataset no longer appears to be publically retrievable.

Research Question 3: Are human evaluators able to distinguish the synthesised audio (using Baas et al., 2023) of those Speech Genres?

Hypotheses

Text Classification

H1.1 (RCNN F1 Score): Given a set of speech functions *SF* the RCNN should achieve similar accuracies as seen in Lai et al. (≥ 0.95).

H1.2 (RCNN Character Length): An RCNN trained on a particular character length range of text input will perform better on test data from this range than on test data which lies mostly outside of this range.

H1.3 (RCNN Performance based on Dataset): Due to the difficulty of obtaining text data for the Documentary, and the nature by which it was acquired, the RCNN is expected to perform the worst (have the lowest F1 score) on the Documentary genre under all conditions.

Speech Synthesis

H2.1 (kNN Voice Conversion vs. Base FastSpeech 2): The mean opinion scores (MOS) of the audio samples generated by the kNN Voice Conversion (Baas et al., 2023) should be higher than the counterpart audio samples generated by FastSpeech 2 (Ren et al., 2022) alone, regardless of the Speech Register (*SR*) applied to it, due to the additional prosody applied by kNN Voice Conversion. Samples with the appropriate *SR* will have a higher MOS score than samples with a conflicting *SR*.

H2.2 (Speech Register Discernibility): Human evaluators will be able to identify the Speech Register of a sample without the sample containing the accompanying Speech Function. The rate of correct identification should be higher than through guessing (in this case, with 4 genres, is 25%). This hypothesis is based on the assumption that speech registers are inherently distinguishable.

H2.3 (Gender Preference): In line with previous studies (e.g. Mullennix et al., 2003; Hengst, 2021), the average MOS for male samples will be higher than for female samples (only considering the kNN Voice Conversion samples). No genre specific prediction is made.

H2.4 (k Value of kNN Voice Conversion): In accordance with the assumptions of Baas et al., 2023, the samples using k=20 should have the highest MOS. The synthesis process is the same in both papers, thus the results should also be the same.

H2.5 (Discernability vs. Time Spent Listening to a given Genre): The discernability of a genre should be higher for evaluators which spend more time listening to a given genre, as evaluators with more experience of a genre are expected to be more familiar with the prosodic nuances of that genre.

3 Theoretical Framework

Since no previous study has tried to synthesise Speech Genres (especially within the specific framing of this paper), a development of theory is required to translate how speech genres, functions and registers would translate into workable data and architectural structures. This section explains the general problem being tackled; that being the combination of Text Classification and Speech Synthesis, while defining the problem space tackled in this paper specifically. Finally, this section discusses the means of evaluation, and why a standard metric would not suffice. A table explaining the various symbols used throughout the section can be found in appendix A. Additionally, the specific architecture employed to tackle this problem is also explained. The methodology section (section 4) deals with the specific architecture employed to tackle the specified problem space, alongside practical aspects of data collection, data processing, and training the architecture (hyperparameter choice). The discussion (section 7.2) motviates an approach without the selection of a specific architecture and other considerations which may arise.

3.1 General Problem Space

The task at hand is to generate a synthetic speech sample A given an initial text input T. This describes a conventional Speech Synthesis (SS) problem. We distinguish from the conventional problem by wishing to synthesize different types of A given different types of T, specifically different *genres* of speech (as informed by Bakhtin, 2011, Dementyev, 2016). A and T are sets of synthesised speech and text respectively. If t_i is the input text (for example: *"Let me tell you, that was quite the view"*), then a_i is the corresponding synthesised speech.

Genre can be broken into 2 fragments; the *Speech Register* (*SR*) which is deducible from the speech (by characteristics similar to those presented in Weeks, 1971, Egbert and Mahlberg, 2020, and Kortmann, 2020) and the *Speech Function* (*SF*), which is deducible from the text (based on understandings presented in Van Thao, 2022 and Holmes, 2013). We say that each a_i in A has a specific Speech Register sr_i, and each t_i in T has a specific Speech Function sf_i. For now, we assumes that the number of elements in the group *SF* is equivalent to the number of elements in *SR*, and that each sf_i maps onto one, and only one sr_i ($f : SR \rightarrow SF$). The discussion section expands on potential problem spaces with a one-to-many or many-to-one mapping (particularly, a given sr_i being composed of multiple speakers/dialects/etc.).

For a given t_i, a prediction needs to be made regarding to which sf_i it belongs. This is done by the task of Text Classification (TC). For each sf_i we wish to distinguish, we need a set T_i, which is a database of text which is determined to belong to sf_i. TC usually does not handle raw input, and requires preprocessing before being usable (the preprocessing can depend on the type of text, and the context of application). We will refer to preprocessed text as $t_{i(tc)}$. TC subsequently takes a $t_{i(tc)}$ as input and outputs an *SF* prediction.[2]

A final consideration is whether one wishes for the TC to also classify unknown or uncategorised text. The latter would entail having the TC learn to distinguish a separate class, which could be called "neutral", "none", or "other". We will call such a set $sf_\varnothing$. Any $t_{i(tc)}$ classified as $sf_\varnothing$ is ideally[3] said to not belong to any of the other *SF* classes either due to not wishing to classify it within the specific problem space, or because the text is deemed to not belong to any group/deemed to be neutral. Consider a sentence like "The person jumped.", which can be deemed either insufficient to classify, could exist in multiple different *SF* groups, or may belong to a category (such as children's novels) which we may not wish to classify.

[2] In actuality, TC would output a probability distribution across all sf_i in *SF*. An *argmax* function would be applied onto this probability distribution, and by definition of the *argmax*, the most probable sf_i is the output.

[3] This assumes an accurate classifier. A prediction of $sf_\varnothing$ could also be a misclassification of course.

3.1.1 Speech Functions & Speech Registers

Having identified what the TC and SS modules do, we can go about asking how to define a set of *SF* and *SR*.

Speech Function and Speech Register Sets

Before we do, we may motivate the two sets being different. Since both *SF* and *SR* refer to the same genre, we may ask why they're considered as separate sets. In a case where $n(SF) \neq n(SR)$, the reason may be more clear. In particular, a case where $n(SF) < n(SR)$, we would have at least one sf_i which maps onto more than one sr_i, which we would thus wish to distinguish.

But even when $n(SF) = n(SR)$, the distinction may be non-trivial. Each element of *SF* contains the information of T_{tc} (which is the text to be spoken, and any contextual cues around the text), whereas each element *SR* contains the information of T_{ss} (only the text to be spoken) and the audio representation for each *SR* (A_{sr}; which contains important information regarding prosody).

The *SF* has no reference on how a text should sound like after being synthesised. However it has information on the meta characteristics of the text; it can contain text which would not be spoken (thus not synthesised). The *SR* is the opposite; it contains the necessary audio information (such as prosody) for a given synthesised audio, but is only provided the text to be spoken (as the rest of the text is necessarily redundant).

Let's have a specific example. Suppose we choose to synthesise (among others) a documentary genre. A documentary may entail audio of not only the relevant speech, but also background noise, background music, speech of other individuals being interviewed, reenactments, and many others. In a documentary script, these events may be explicitly delineated:

```
*Background music - Sad Violin*

*Visual: Still shot of ruined city*

Narrator: As can be seen, this event had a major impact on the local community.

*Fade Background Music*

John Doe: Indeed, it set me back thousands, it made me lose a lot of my
savings. It's been very hard on the family.

Jay Doe: The house was destroyed beyond recognition. I can still picture it
like it was yesterday. It was awful.
```

Not only this, but a documentary script could contain information about the visuals and timestamps. This can be interpreted as meta information to the text, and thus useful in classification of the *SF* group. If the above were used in training, our $t_{i(tc)}$ could include all of the above (minus certain preprocesses such as stopword removal), where the $t_{i(ss)}$ would only include the narrator text, alongside the audio of the narrator.

Defining a Set

Referring back to the example of a documentary, it may be pointed out that there are different types of documentaries. They may vary by topic, such as nature, history, scientific, biographical, and so on. But there are also different genres of documentaries, such as poetic, expository, observational; among others. When faced with the task of forming *SF* and *SR* groups, one is at liberty to categorise as they wish, whether it be per topic, by genre, or lumping them all into one category.

Extending what was said in 3.1.2, the *SF* and *SR* sets can be a supervised or unsupervised task. At the extreme end of being unsupervised, a set T of text can be passed, by which the *SF* and *SR* are both abstract spaces, whose size can be any n. Assuming an adequately large and processed dataset, this may aid the accuracy of output, but may compromise controllability and usability (an abstract space is necessarily more difficult to control than one which is explicitly defined).

3.2 Evaluation Metrics

Since the two main tasks are Text Classification and Speech Synthesis, there are at least two tasks to identify in the simplest case. The first evaluation is to determine how well the Text Classifier is able to correctly predict *SF*. The second evaluation is of the extent to which the output of the synthesiser is liked or disliked.

3.2.1 Text Classification

The task of TC is, given an input t_i, to predict which sf_i it belongs to.

We may begin with the simplest problem space; where we have two elements of *SF*: sf_x and sf_y. Let's assume that all elements of T are known to belong to either group, and that there is no issue of subjectivity (each element is absolutely certain to belong to its group regardless of who is ascribing the group). In such a case, we are merely interested in maximising the accuracy of predictions. Thus we are looking to maximise the amount of correct predictions divided by all predictions.

Suppose we now expand the T set to include unknown texts, or random text. We know that some elements t_i belong to either class of *SF*, and that some do not belong to either. We thus classify a new, third group $sf_\varnothing$, as described earlier. However, the assumption remains that each of the groups have no issues of subjectivity. We can make two accuracy measures here; the conventional accuracy metric between the three groups (let's call is Acc_1 for simplicity), and also accuracy of distinguishing between an sf_i from $sf_\varnothing$ (Acc_2). The second task is a binary classification task of either assigning a $sf_\varnothing$, or **not** $sf_\varnothing$, where **not** $sf_\varnothing$ is the rest of the *SF* group. In short, Acc_1 determines how well the classes are classified, whereas Acc_2 describes how well the model performs in recognising the desirable classes at all. Acc_2 may be useful in cases where the distinction between different sf_i isn't as important as distinguishing that it is not a part of $t_\varnothing$. We can also expand the amount of sf_i in *SF* to any number desired. Acc_2 remains the same, whereas Acc_1 can be an unweighted, or weighted accuracy depending on the use case.

We may finally drop the assumption of no subjectivity. It may be the case that certain genres have very similar speech functions. Consider for example, radio news broadcasting (let's call it sf_{radio}) against television news broadcasting (sf_{tv}). While they can be acknowledged as being separate on account of having to be communicated across different media (perhaps having to accommodate different limitations) and potentially targeted to different demographics, it's reasonable to assume that, since both contexts are tasked with delivering information verbally across a medium, that they would share a lot of characteristics. In such a case, a lower Acc_1 may not be indicative of a poor model, but rather of two sf_i being very similar. This may not be an issue if the corresponding elements of *SF* are also similar; we will discuss this in the evaluation of the speech synthesis.

A similarity between two sf_i may also inadvertently impact Acc_2. Consider that we pass a t_i which we know to belong to (sf_{tv}). Suppose we first cluster both radio and television into (sf_{news}). Let's say that the probability distribution of the Text Classifier says 60% for sf_{news} and 40% for $sf_\varnothing$. In such a case, the *argmax* yields a prediction of sf_{news}, which would then go on to produce some type of sr_{news}. However, if broken back down to sf_{radio} and sf_{tv}, we may instead have a prediction of 35% for sf_{tv}, 25% for sf_{radio}, and 40% for $sf_\varnothing$. Now, the *argmax* would force the synthesiser to synthesise audio with an $sr_\varnothing$ label. In most cases, it would have been preferable to synthesise sr_{radio}, even if it's the "incorrect" *SF* classification. This example is perhaps extreme for the sake

of demonstration, but if we imagine a much larger SF set, we can imagine this error becoming worse.

We may consider circumventing the *argmax* by implementing minimum thresholds by which we determine to use a certain element of SF. In our earlier example, we could avoid the generation of $sr_{\varnothing}$ by implementing a threshold of 35% to generate sr_{tv}. To make a generalisation, we can say that for every sf_i, there is a probability threshold of $P(sf_i) \geq x; (0 < x < 1)$ by which we generate the corresponding sr_i. This may solve the issue of generating an undesirable $sr_{\varnothing}$, but may yield an additional issue. What if more than one sf_i crosses its threshold for generation? In such a case, we may simply determine the priority of each by means of ranking which threshold is more important. Determination thereof is necessarily a subjective and contextual criterion. This, and the previous metrics, are visualised below in fig. 1.

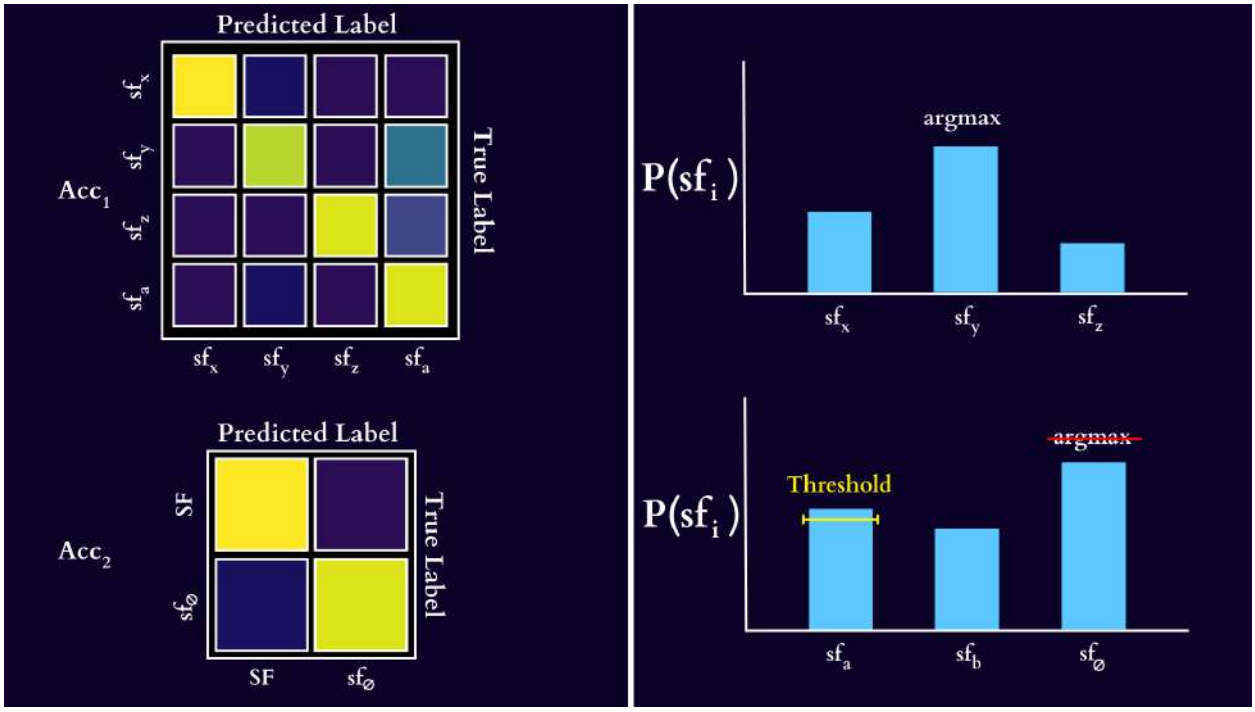

Figure 1: To the left, a depiction of Acc_1 and Acc_2. To the right, two graphs showing the predictions of each (sf_i) by the Text Classifier. In the upper right diagram, we simply choose the highest probability using the *argmax* function. In the lower right, we set a threshold for one of the (sf_i) such that, when crossed, the TC predicts that particular (sf_i) even if another class has a higher probability.

To summarise, when training a model to predict the sf_i that a given t_i belongs to, we may go from a simple problem space of simply maximising the accuracy, or we may have to compromise accuracy with other factors specific to the application of the architecture. We can either use a "none" class $sf_{\varnothing}$ to separate texts of interest from the rest, combine or merge different elements of SF to aid a more desirable SR output, or place thresholds on certain elements of SF by which we generate the corresponding sr_i.

3.2.2 Speech Synthesis

The task of the synthesiser is to generate a_i from a t_i input given an sr_i token.

The evaluation of the synthesised samples in A can be broken down into two categories; the discernibility of the various sr_i, and the preferability of the synthesis.

Discernibility

The following is a foundational assumption:

> The distinction between elements of *SR* are trivial if, given a task of classification between them, they cannot be distinguished by a human listener.

The idea behind making distinctions between different elements of *SR* is that they are supposed to be perceived as fundamentally different manners of speaking which are specific to a given context. If we have a certain sr_i, we say that it carries characteristics which separate them both from a "neutral" manner of speech, as well as other manners of speech. If, upon evaluation, the listeners cannot hear these supposed characteristics being manifested in the samples of a_i, then the effort put into parameterising the corresponding sr_i, alongside gathering the necessary data, doing the necessary preprocessing, and training the necessary architectures, is rendered redundant. If a history documentary register cannot be told apart from a nature documentary register, then there is no point in defining them as separate classes.

Subsequently, we define the task of discernibility as an accuracy evaluation, much in a similar way to how the accuracy worked for TC. There are different methods by which this can be done[4] (see fig. 2)

The simplest option is to present a prompt of a_i and ask a listener to classify between different elements of *SR* which are explicitly presented in text form (including an $sr_\varnothing$ option). This can be done either pair-wise (especially if it may be more important to distinguish between two specific elements of *SR*) or in broader groups. In this case, you allow the listener to carry forward their own interpretation of each of the *SR* elements.

Alternatively, we may again have a prompt of a_i and ask a listener to classify it. However, instead of classifying between written options such as "News Broadcast" or "Children's Bedtime Story", we instead use real audio clips for their options; they listen to an actual recording of the aforementioned options, and choose the one they think fits a_i the best. In such a scenario, the listener has to deal in terms of the researcher's interpretation of the various elements of *SR* rather than their own.

We may alter either of the aforementioned options by having the users ascribe a similarity percentage to each class offered, rather than having to explicitly choose one of the classes. For example, using our example of a nature and history documentary, an a_i sample may be determined to have a similarity of 85% to a nature documentary and 55% to a history documentary. Note that the sum of percentages need not be 100%; if an a_i is deemed to be similar to two classes, a high percentage can be ascribed to both. If two classes are consistently given the same percentage for various a_i, it may be the case that these two classes are indiscernible, and thus it may be redundant to distinguish between them. An sr_i which shows high similarity values for $sr_\varnothing$ may be indicative of it being insufficiently distinguishable from neutral speech, inadequate training audio data for that particular sr_i, or an architecture which cannot adequately capture the prosodic nuance of the given sr_i.

Perhaps the most difficult option to parametrise and quantify; we may give an a_i and ask the listener to describe the kind of sr_i that they are listening to.[5] The answers can be as open as

[4]The list and graphic below are not an exhaustive set. Other methods could include choosing between pictures, drawing the face behind the voice, asking the participant to mimic the voice, alongside other interpretive methods. Evaluators can also be given a set of descriptors & characteristics which they can select (tick all that apply).

[5]The manner by which this is asked can have an impact on how the respondents answer. For example, asking "Describe the manner of speech that you hear in the following audio extract" could yield different answer if phrased as "Describe the kind of speaker you envision based on the following audio extract". It would be important to research the nuances of how such open questions are framed.

desirable; for example certain prompts could include "A local radio station host announcing the afternoon news", "A parent reading a fable to their child" or any other open ended interpretation. While this method would be the most difficult to quantify, and determine the extent of success of the assembled architecture, it would yield the most nuanced view of how neutral listeners perceive a given sr_i (or misperceive it, or if they perceive it at all).

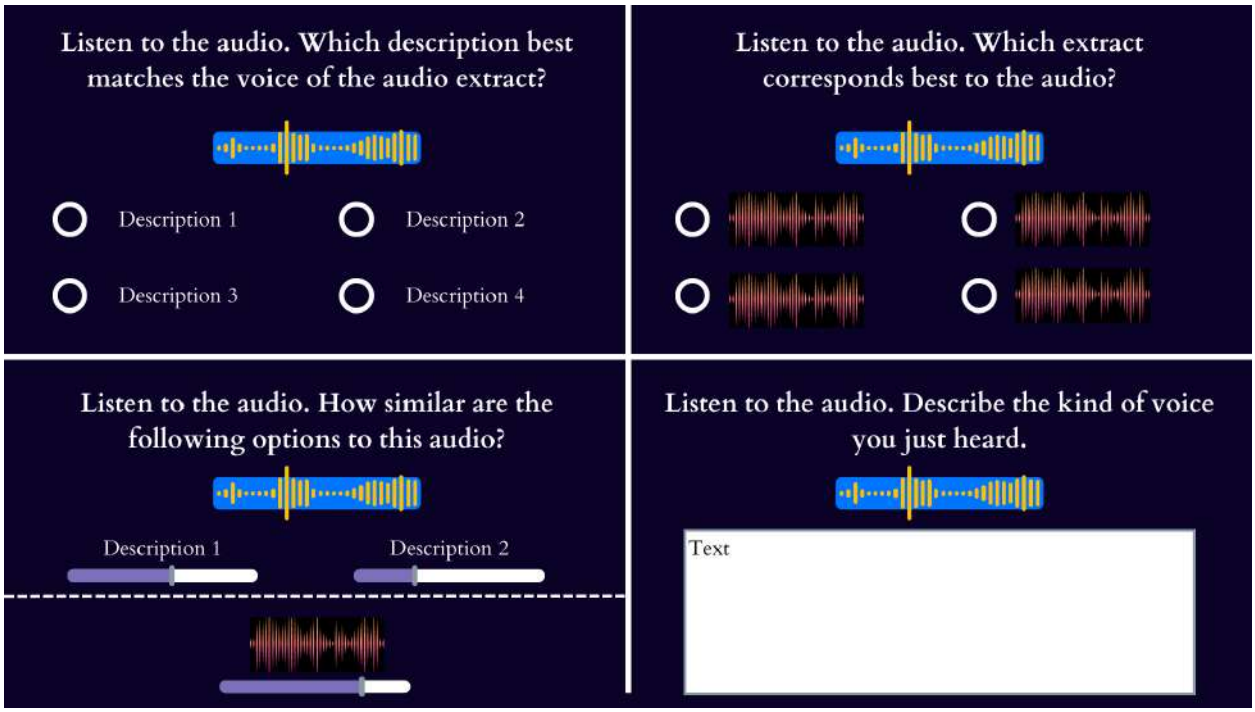

Figure 2: The aforementioned 4 methods of evaluation. We can ask evaluators to choose between labels (Point 1), or choosing the closest matching audio extract of real speech (Point 2). We can also choose to let the evaluator assign a similarity/probability score for either one of Point 1 or 2. Finally, we can ask the user to simply describe the kind of voice they're hearing.

An important point to consider in all samples is the nature of discernment. When discerning an a_i, we use both the prosodic content, and the linguistic content to determine the type of speech we listen to. Consider an audio prompt with the following text:

> *"A local community centre has suffered extensive damage following local thunderstorms over the last three days. Damages are expected to be in between five to seven million dollars."*

This would be classified as a news report on account of the linguistic content of the prompt. More importantly, the prosodic element, or indeed the information conveyed by the corresponding sr_i, is made redundant as the linguistic content is a very strong indicator. In essence, the listener would be classifying the sf_i more so than the sr_i. A high accuracy of discernment in that case tells us nothing about how well the model conveyed the sr_i onto the original t_i passed through it, making it a useless evaluator. To evaluate the sr_i, and be sure that they are being recognised by their prosodic elements, it is advisable that the prompts have a neutral linguistic content; the text cannot hint at any particular genre. A potential exception to this could be an evaluation in the genre of Point 3 where one tests a given sr_i on text which is linguistically congruent, neutral, or conflicting to said sr_i. In such a scenario, the difference in the similarity score can be measured across the three variables to determine the relevant of the prosody against the linguistic content.

To summarise, it's important that each sr_i is discernible, otherwise it is a waste of effort to dedicate time to training them as separate classes. There are different means to have listeners classify

the classes, either by choosing an option among many, or give a qualitative description thereof. It's also important to ensure that it's the prosodic content that's being used for discernment as opposing to the linguistic content.

Preferability of Synthesis

The degree to which a synthesised speech sample is viewed favourably or unfavourably can vary regardless of the accuracy of discernment of the various sr_i. While listeners may recognise a comedy stand-up register every time, they may also agree that the delivery is poor, uncanny, robotic, or any other descriptor.

This paper does not propose any new method of SS Evaluation, rather instead encouraging any application of the architecture to consult the current state-of-the-art methods (both for objective, subjective and psychological criteria). As of writing, most SS Evaluations either use Mean Opinion Scores (MOS) or the Comparative MOS (CMOS) for subjective evaluation. Additionally, surveys can be used if the context of application is known, and a subsequent estimation of listener demands can be derived. See Table 1 of (Wagner et al., 2019) for an in-depth example. While there is evidence to suggest that MOS is a issues of reliability and interpretability (Le Maguer et al., 2024), MOS is used in this study as literature necessary for comparison of results also uses it.

Evaluators

To ask the question of what makes a good radio broadcaster, a good news anchor, a good narrator, etc, is necessarily a subjective question of personal preference and taste. While there may be a large agreement on what a news anchor is, there may not necessarily be agreement on what makes a "good" news anchor.

For each given sr_i, different evaluators may have different experiences with the real life equivalents. For example, if testing a documentary register, the evaluators may have watched completely different documentaries in the past, with different narrators, different production companies, different visuals to go along with the narration, different production quality (depending on the time of production of documentary); which are factors which fundamentally change their perception of the documentary register. Additionally, different listeners have different levels of experience with different registers. If we are testing between the registers of a radio broadcaster, a stand-up routine, and a podcast; we may have evaluators which listen to the radio daily but don't watch comedy stand-up, others which listen to podcasts while studying without ever listening to the radio, and so on.

There are many other factors which may contribute to the perception of a register. The country of origin, or country of residence, of where a listener has encountered a given register may impact evaluation to a certain extent even if a register may share characteristics across different cultures and countries. The relative time at which an evaluator interacted with a register within their lifetimes may affect the perception of genre; a person who only listened to a register of childhood storytelling in their early years may have a different perception than a parent who has a child and also has to listen to childhood storytelling. Similarly, the general time at which an evaluator has or had listened to a register may provide differing perceptions of a register; someone who listened to the radio in the 1970's may have a different perception of the radio broadcaster register than someone who has listened to the radio in the 2000's. There can also be aesthetic preferences like the sentimental value held for a particular speaker of a genre. Certain people may associate a specific speaker with a particular register, and subsequently view any other speaker as inferior. The classic example is David Attenborough being known for his nature documentaries. However the sentimental value can also stem from a more personal point, such as having one's mother read a bedtime story (we could consider this the linguistic equivalent of "Nothing beats the cooking of the

mother"). In such a case, a particular person can be explicitly or implicitly deemed to be the gold standard of a register, and any other alternative could be scored lower in evaluation, not necessarily reflecting the ability of the synthesiser to synthesise registers. Or conversely, a particular delivery may receive an excessively high ration if the listener is reminded of someone, or have a positive memory from said delivery. Finally, there can also be prejudices and biasses regarding the type of voice that should be associated with a given register. Certain listeners may have an implicit or explicit assumption regarding the kind of voice that should deliver a certain register, such as the gender of speaker, age of speaker, accent of speaker, and so on. If a generated sample matches one's prejudices, they may give it a higher rating than if there were to be a mismatch.

The point here, is that the context of the listener is important to consider when evaluating a given set of registers, as said context may have a noticeable effect on the results of subjective evaluation. In idealistic conditions, a researcher may be able to ask the evaluator, aside from the questions directly related to evaluation, about some background information of the evaluator to have a better understanding of the kinds of ratings that the evaluator gives. If the application is intended towards a specific group of people, then indeed one may merely use people from said group for the evaluation, with a bit of additional background information if needed. In cases of a broader audience (such as a voice assistant), it may be desirable to segment the audience into different groups (the means of segmentation is entirely up to the application) and note the different trends across the different groups.

Idealistically, this sounds reasonable. However, a logistical problem arises. On the one hand, we seek to ask as many questions as possible to get as much information out of the evaluation. On the other, the listener wishes to answer as little questions as possible. It's relatively safe to assume that one would prefer to answer five questions as opposed to a hundred. As the number of answers increase, the willingness to answer is very likely to decrease (either directly by annoyance, or indirectly through fatigue), which could yield less accurate/reliable answers. As such, a prospective researcher must contend with compromising depth of information against the patience and willingness of the evaluator.

To summarise, evaluation of different registers can be majorly influenced by a given evaluator's previous experience with the real world equivalent to the register. This may impact the evaluations given, and may cast doubt over how much the results reflect the performance of the architecture. As such, obtaining the context of the evaluator can be useful to do in conjunction with the evaluation, so long as this doesn't reduce the willingness of the evaluator to evaluate.

4 Methodology & Architecture

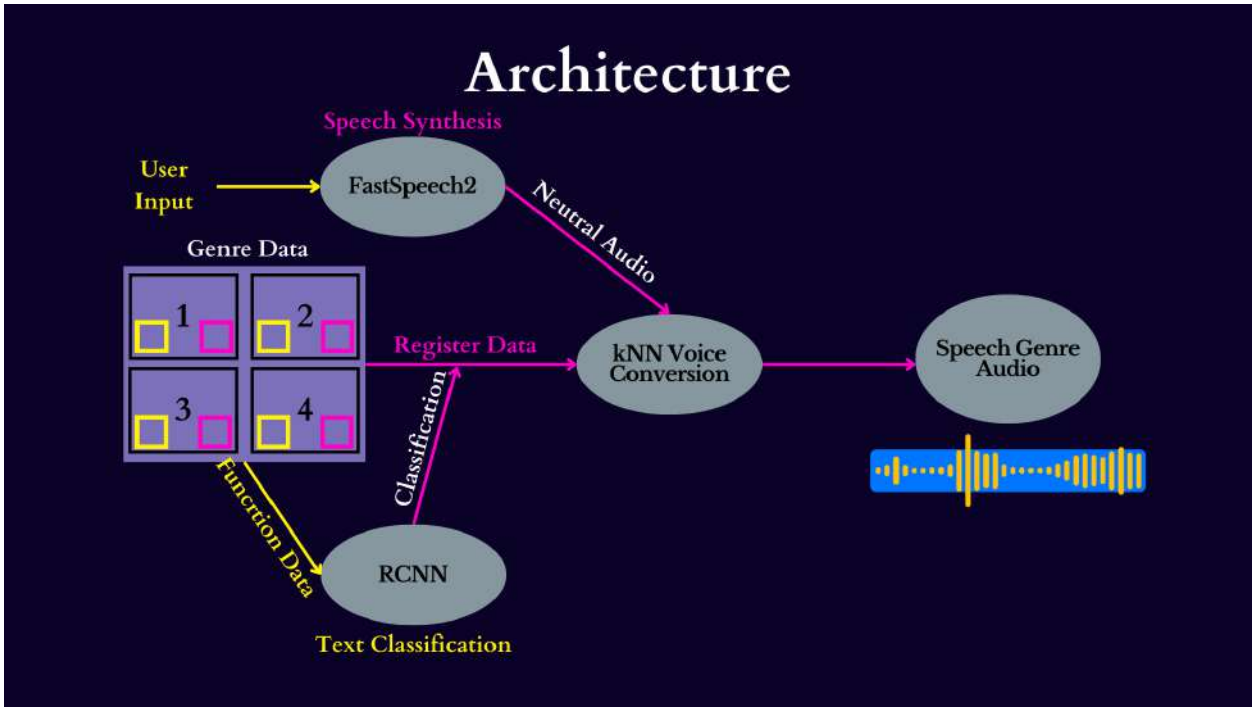

Figure 3: An overview of the architecture assembled to synthesise speech genres. User input goes into FastSpeech 2 (Ren et al., 2022) to undergo speech synthesis. Simultaneously, the the input text goes into an RCNN (Lai et al., 2015) to undergo text classification. Once a prediction is made, it can be used to find the specific dataset needed to synthesise the speech genre. This is done through kNN Voice Conversion (Baas et al., 2023) which performs a kNN regression between the FastSpeech 2 output and the chosen dataset, which yields the final output.

4.1 Defining The Specific Problem Space

Having established the framework, we use it to define the specific problem space of this paper, comprising the genres, architectures and data to be chosen (fig. 3).

For this paper, four speech genres were defined:

(x_{news}). The genre is meant to capture the manner of speaking seen in dedicated news channels such as CNN, CNBC, and other such broadcasting channels. It is primarily dominated by news presented on televisions in the United States of America. It also does not capture amateur or third party news sources, such as Youtube channels (unless the Youtube channel content is from the same entity which broadcasts on television, such as a CNN Youtube channel). Catenaccio et al. (Catenaccio et al., 2011) provides an overview of the analysis of news as a genre.

Documentary (x_{docu}). The type of documentaries within this genre are broadly expository; it relies on telling the story of an individual, inform about a subject, or make persuasions regarding a specific issue. Documentaries can be made by either professional entities, such as National Geographic or Discovery; or they can be made by amateur entities, such as Youtube channels. An analysis of expository documentaries, and the role of the narrator, is provided by Wolfe (Wolfe, 1997).

Comedy Stand-Up (x_{comedy}). The genre is meant to encompass a manner of speech seen when comedians give a stand-up routine in front of a large audience. It includes only the stand-up that is televised, and in explicit stand-up shows (as opposed to stand-up routines done in other establishments as an attraction). The major reference points were comedy specials by famous

comedians; broadcasts on Comedy Central, and stand-up skits on Saturday Night Live. An in-depth analysis of the comedy genre is provided by Aladhami (Aladhami, 2024).

TED-Talk (x_{TED}). A manner of speech which is known to the TED brand; wherein a speaker gives a presentation to an audience about a specific topic. TED offers training in order for its speakers to have a particular set of mannerisms when speaking. The genre captured here can include both official TED events, as well as TED-x events (where TED events are meant for a global audience, and TED-x for a more local one).

These speech genres are defined by subjective means, but with an assumption that each of them have a distinct manner of speech from each other. This paper doesn't define a $\varnothing$ genre on account of not dealing with unknown text. All of the text comes from known sources, and is known to belong to any of the defined genres. To ensure speaker independence of the various sr_i (and thus capturing the genre, rather than style), the data for each genre is collected from multiple speakers. The problem space does not distinguish an $x_{\varnothing}$ class at any time. Only English language data is used due to the mass availability of English data, and the ability of the author to navigate the language.

4.2 Data Collection

The data required for the problem space would be text transcripts and audio recordings of the aforementioned speech genres (the genre data as shown in fig. 3). For example, the x_{comedy} class would require transcripts and audio recordings of various stand-up routines.

Unfortunately, as of the time when the thesis was done, there were no databases available freely which contained any of the desired speech genres. As such, the data had to be gathered by means of web scraping. An overview of the sources of data is given here; and further details surrounding the general web scraping process are given in Appendix B and the subsequent pre-processing sections for both Text Classification and Speech Synthesis. The data is available in Appendix C.

4.2.1 Text Data Acquisition

This section outlines how the various groups of T were collected (the Function Data as depicted in fig. 3). For each of these, the amount of data collected was as large as possible. None of the categories had a specific limit on either the time at which is was created, nor the length of the text. An illustration of the process is shown in fig. 5.

The News Report genre was collected from **https://transcripts.cnn.com/**, which is a website that uploads the transcripts of broadcasts seen on the Cable News Network media company (CNN). The following categories of broadcast were used[6]:

- Anderson Cooper 360°
- CNN Special Reports
- The Source with Kaitlan Collins
- Fareed Zakaria GPS

Each of these underwent web scraping as outlined in Appendix B.

The Documentary class genre was collected from various different Youtube Channels. Documentaries were considered from the following sources (among others):

- ABC News

[6]While in theory, these could all go on to form separate *SF* and *SR* classes, here they are treated as one, "News Report" class in order to avoid overfitting to a particular presenter's method of speaking, such as only modelling Anderson Cooper's delivery.

- Channel 4
- Best Documentary
- NOVA PBS
- Wondody
- TV - Quantum Universe
- DW Documentary
- The People Profiles

The criteria of selection were centered around balancing professional and amateur documentary production. The documentaries tended to follow a generally expository genre (see Zabetie Jahromi and Qaneifard, 2018 for an in-depth explanation) of a narrator talking over a video production. The documentaries were somewhat varied by topic, including science, history, crime, conspiracy theories, among others. The actual content of the documentaries was not considered relevant as there is no need for the architecture to understand the content, but merely the genre of speech. In other words, it did not matter if a documentary contained flawed information (though documentaries that were incorrect were not explicitly sought after either). The text was derived from downloading the captions of the documentaries using the `youtube-transcript-api`. This means that the quality of transcription is contingent on the quality of the captions, depending on if they are automatically generated or manually uploaded with the YouTube video. However, additional information such as music indicators and general background noise indicators are also included in these captions, which aid the classification process. The automatic captioning tended to have a poor quality output compared to the 3 other genres, but the output was intelligible enough and contained enough information to be used for analysis.

The Comedy Stand-Up genre was gathered from two sources:

- **https://scrapsfromtheloft.com/stand-up-comedy-scripts/**, which is a site storing the transcripts of various stand-up routines. It contains different venues, different comedians, and has a time range of 60 years (1963-2023. not evenly distributed). The transcript also contains information about the audience, background music, delivery of speech (mumbling, accents, etc.) and laughter, which all aid in *SF* classification.
- **https://snltranscripts.jt.org/tag/stand-up** which contains a series of stand-up monologues of various different comedians on Saturday Night Live (SNL).

The TED-Talk genre was directly scraped from **https://www.ted.com/talks** by scraping the transcript of each talk which is provided. As stated earlier, both official TED events and TED-x events were considered. Shorter educational videos were also included on account of also being both informative, and generally similar in speech genre, whereas podcasts and debates were rejected on account of the speech genre being much more conversational.

4.3 Data Pre-Processing

Here is the outline of how our initial groups of T were processed into $T_{(tc)}$ and $T_{(ss)}$, with an illustration of the process shown by fig. 5.

4.3.1 Text Classification Pre-Processing

All of the data gathered for training the Text Classifier was scraped from web pages. As such, not only is the retrieved text usually wrapped in HTML and CSS code, but also not readily available to

be used for machine learning tasks. Further details on the libraries mentioned here are discussed in Appendix B.

Using Python, each of the web pages were accessed using the urllib Python library. A combination of BeautifulSoup and re (regex) were used to extract the text from the retrieved web page. This yielded one large body of text per webpage, yielding a t_i sample. These t_i samples are too large for machine learning purposes. Examples of t_i would be the transcript of a 15 minute TED-talk, a 90 minute documentary, a 60 minute news segment, and so on. The t_i had to be broken down.

Every element of T was merged into one string by storing T as a list, and using the `' '.join(list)` command. This string was then divided into individual sentences. This was by adding substrings of the string to a new, empty string. With the sentence "The boy jumped.", T,h,e,(space),b,o, and so on, would be added until the full stop, at which point the sentence was added to a new list, the empty string was cleared again, and the process would repeat. These sentences were recombined into new samples, depending on a specific range of character length. Since Lai et al., 2015 did not specify an optimal range for the length of the texts to be input, different ranges were tested in determining which range of length yielded the highest accuracy. Three ranges of character length were used; 600-1000; 800-1200; and 1000-1400. An additional 200-600 range was made afterwards to test shorter samples.

To do this, sentences were added to an empty string. If the character length was below the lower limit of the range, another was added until the length crossed the lower limit. If the sentence was within range, it was added to a new list, and the process started again. If the character length exceeded the upper limit, the last sentence was removed, and the previous iteration of the string was added. This means that while most of the samples within the dataset are within the range, some samples were below the range. If a single sentence exceeded the upper range, it was discarded. A workaround had to be devised for subtitles automatically generated from Youtube. The yielded subtitles usually generated excessively long strings of words without any full stops, particularly when items such as [Applause], [Music], or other such indicators were present. Such subtitles had full stops artificially placed after every back square bracket ("]"). This process yielded the $T_{(tc)}$ group. Figure 4 shows an example output, showcasing the character length distribution. An example of the character length distribution of T_{tc} is shown in fig. 4.

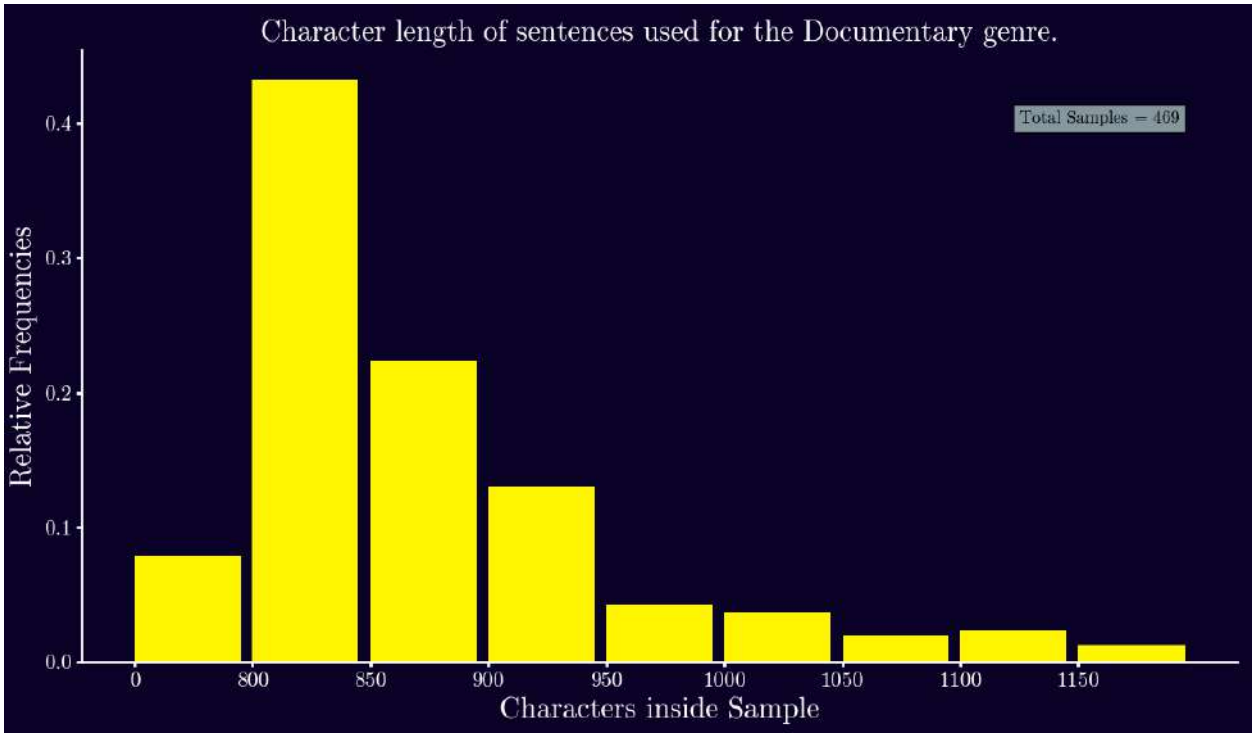

Figure 4: Character length distribution from a subsample of the T_{docu} text, using the 800-1200 range. As discussed previously, the majority of samples lie within the designated range, and a small subset lies below the range.

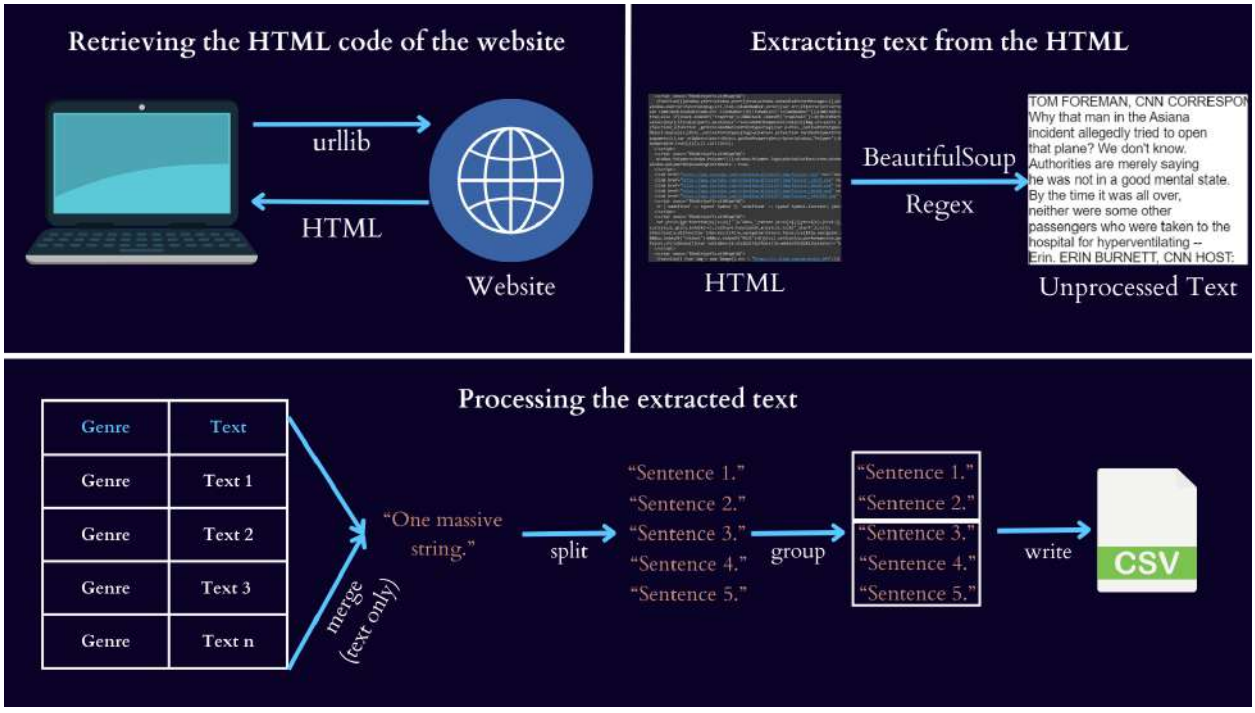

Figure 5: An illustration of the Web Scraping process as discussed above. The HTML code of the website is yielded, from which the text is extracted. The text is then processed by means of regrouping into smaller chunks of texts which can be used for machine learning tasks.

Each of the groups of $T_{(tc)}$ were merged into one large database, and split into a train and test set using the sklearn Python library. The test set was set to 15% of the database. This yielded the train and test sets which were going to be used for training the Text Classifier.

4.4 Audio Data Acquisition

This section outlines how the various groups of *A* were collected (the Register Data as depicted in fig. 3). Upon initial testing with kNN-VC, it was deemed necessary to split each of the genres into different pitch groups. There was a noticeable difference in the fundamental frequency of the male and female speakers, and the kNN-VC output contained very prominent and sudden changes in fundamental frequency, as if altering between a male and female speaker mid sentence. Following convention, the dataset was split into male and female sounding speakers.

The News genre was collected from the CNN website and various subsidiaries of MSNBC. The audio data for CNN was readily available from the CNN website whereas the MSNBC audio data was derived from YouTube. It contains 24.240 minutes of data, split over 11.371 minutes for male speech and 12.869 minutes for female speech. The average sample length is 11.027 seconds, with a minimum of 3.259 and a maximum of 44.606 seconds. This dataset was split across 39 Breaking News segments (CNN) and 11 MSNBC broadcasts.

The TEDTalk genre was collected from the YouTube Shorts section of the official TED channel. It contains 30.787 minutes of data, split over 14.653 minutes for male speech and 16.134 minutes for female speech. The average sample length is 6.796 seconds, with a minimum of 3.221 seconds and a maximum of 11.473 seconds. The dataset is composed of 20 male and 20 female speakers.

The Comedy genre was collected from various fragments across YouTube, ranging from snippets from large comedy companies such as Comedy Central, to snippets of stand up routines posted to a comedian's personal channel. It contains 46.059 minutes of data, split over 22.626 minutes for male speech and 23.433 minutes for female speech. The average sample length is 6.01 seconds, with a minimum of 1.942 seconds and a maximum of 11.624 seconds. The dataset is composed of 20 male and 19 female speakers. The selection of speakers were guided by various online comedian rankings (e.g. "Top 20 Female Comedians of All Time").

The Documentary genre was collected from various fragments across YouTube, alongside audio demos from working narrators. It contains 49.742 minutes of data, split over 25.037 minutes for male speech and 24.705 minutes for female speech. The average sample length is 5.596 seconds, with a minimum of 3.005 seconds and a maximum of 10.613 seconds. The dataset is composed of 17 male and 17 female speakers. The selection of speakers were guided by various online comedian rankings (e.g. "Top 20 Female Documentary Narrators of All Time"). For the female category, the data was largely insufficient. This is consistent with literature which points to a large deficit between male and female documentary narrators Siani et al., 2022. To overcome this shortage, the dataset was augmented with audio samples from various female documentary narrators which had posted their audio onto their personal websites for business reasons.

For each of the aforementioned speech genres, the retrieved audio was cut down into segments between 2-15 seconds of length using the Praat (**https://www.fon.hum.uva.nl/praat/**) software. This was done to remove no speech audio, and to find samples with the lowest amount of background noise (particularly for the Comedy and Documentary genres). Segmentation was done manually through the interface rather than by means of a Praat script, thus no script is available. For each genre, except for the Documentary (female) genre, the retrieved audio was in WAV form. Due to data scarcity, some of the Documentary (female) data were retrieved as MP3 files, which were subsequently converted into WAV form. When analysing their final output of kNN-VC, no noticeable decreases with audio quality were found.

4.5 Architecture - Text Classifier

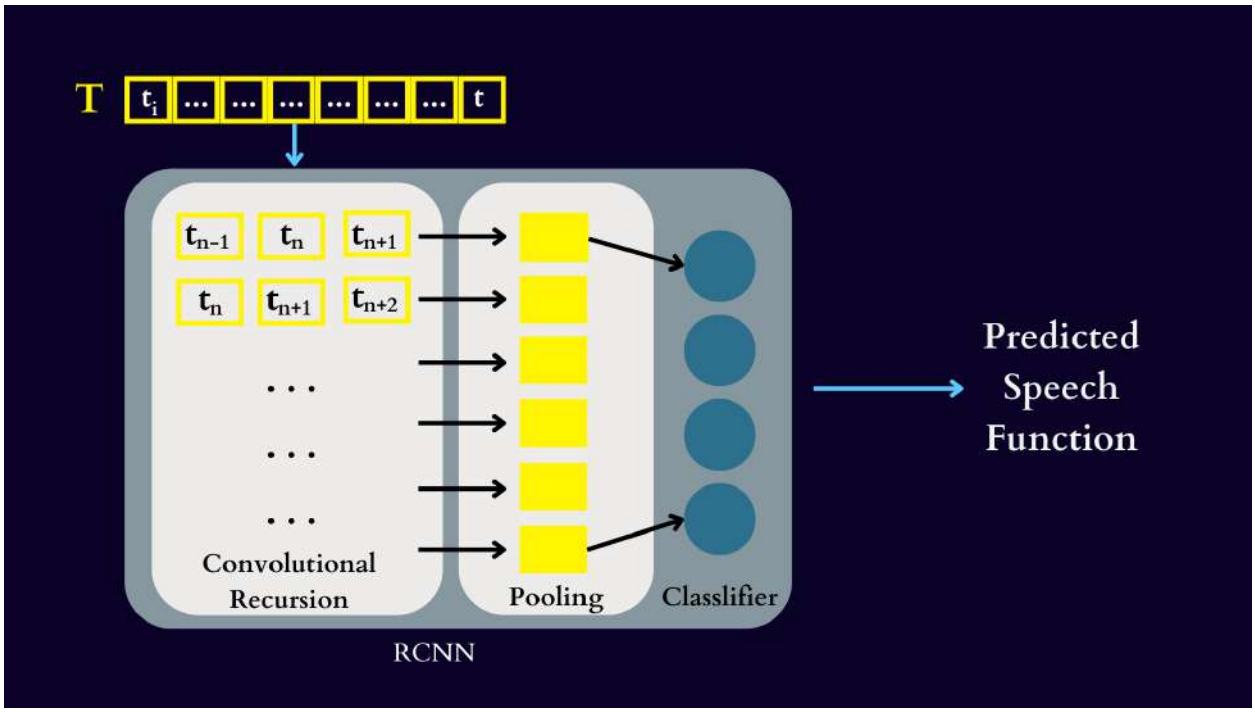

Figure 6: The Recurrent Convolutional Neural Network tasked with Text Classification (determining the Speech Function *SF*) Each *SF* that we wish to distinguish has a corresponding dataset which is used to train the Text Classifier. The prediction is subsequently fed to the kNN Voice Converter.

For the problem of Text Classification (see fig. 3), a Recurrent Convolutional Neural Network is used, as proposed by Lai et a. Lai et al., 2015. The architecture can be broken down by explaining the nature of a Convolutional Neural Network (CNN), and subsequently explaining its Recurrent nature. A CNN is a type of architecture offered as an alternative to Neural Networks which only contain Fully Connected Layers (ones where, for two given layers of a neural network, each neuron connects to every other neuron) within their Hidden Layer set. CNNs employ the use of a kernel (filter) to scan a subset of input in order to extract information about it. An illustration is provided in fig. 6.

The convolution operation ($\circledast$), after which the architecture is named, is an operation which describes how a function f is altered by a function g. Convolution does this through inverting g around the y axis and sliding it across the y axis. The resultant function h is produced by calculating the integral of the area between f and g as g is shifted across the y-axis. Within CNNs, f is represented as a subsection of an input (whether it be a subsection of an image or abstract word embedding), and g is represented by the filter. By altering g, different manipulations of f can be achieved. Different examples of such filters include edge-detection (e.g. the Sobel operator Sobel, 2014), blurring (Tiwari, 2020), and others. This operation forms the convolutional layers within CNNs.

The other type of layer present are Pooling Layers, which take the outputs of a convolutional layer (h from earlier) and reduce its dimensionality. This is also done through a filter, which scans a subset of the input, and extracts a certain value based on the pooling type, whether it be Max Pooling (take the highest value of the subset), Average Pooling (take the average value of the subset), or any other desired method.

A Recurrent Neural Network (RNN) is one which employs the use of a hidden state vector which

is a representation of previous outputs. This hidden state vector acts as a memory unit when processing a given input. RNNs are most useful for sequential data, particularly temporal data. The particular type of RNN employed by Lai et al., 2015 is the Long-Short Term Memory (LSTM) based architecture. This architecture involves the use of a cell, an input gate, an output gate, and a forget cell. The key element is the forget gate, which determines how much of the previous information should be kept or discarded. The forget gate subsequently informs how much memory is used for a given input process.

Lai et al. use the CNN and RNN properties jointly. An input x is constructed using a current word w_i, the word before it w_{i-1}, and the word after it w_{i+1}; hence three words are employed per operation. Each of the words undergoes embedding using the Skip-gram model (Baroni et al., 2014). The left context (c_l) and right context (c_r) are calculated by

$$c_l(w_i) = f(W^{(l)}c_l(w_{i-1}) + W^{(sl)}e(w_{i-1})) \tag{1}$$

$$c_r(w_i) = f(W^{(r)}c_r(w_{i+1}) + W^{(sr)}e(w_{i+1})) \tag{2}$$

where $W^{(l)}$ and $W^{(r)}$ are matrices which transform the hidden layer to the next later, and $W^{(sl)}$ and $W^{(sr)}$ are matrices which combine the semantic information of the current word with the left context of the next word.

The input vector for each word is put together accordingly:

$$x_i = [c_l(w_i); e(w_i); c_r(w_i)] \tag{3}$$

after which it undergoes a linear transformation with tanh activation:

$$y_i^{(2)} = tanh(W^{(2)}x_i + b^{(2)}) \tag{4}$$

and then a max pooling:

$$y_i^{(3)} = \max_{i=1}^{n} y_i^{(2)} \tag{5}$$

With a sequence of $y_i^{(3)}$ for each word Lai et al. then apply a final linear transformation without an activation function, onto which a softmax function is applied to determine the probability of each class.

The parameters, and subsequent dimensionality are the same as in Lai et al. The hyperparameters, which come from Kim, 2019:

- The vocabulary ($|V|$) is set to 80000.
- The vector size of the embedding dimension required for $e(w)$, is set to 300.
- The Hidden Layer size required for $W^{(2)}$ and $W^{(4)}$ is 512.
- The context vector dimension is set to 512.
- Dropout is set to 0.

The batch size is set to 64. For optimisation, the ADAM optimiser (Kingma and Ba, 2017) is used as opposed to Stochastic Gradient Descent from Lai et al.. The learning rate is set to 3×10^{-4}. Most of these hyperparameters are larger than the original publication due to advancements in computational power. While an alternative choice of hyperparameters is possible, it is beyond the scope of this project to further improve them. The RCNN is available in Appendix C.

4.6 Architecture - Speech Synthesiser

The *SS* task is broken down into two parts, text to speech and prosody augmentation. The former is done by a pretrained FastSpeech 2 (Ren et al., 2022) and the latter is done by kNN Voice Converter (Baas et al., 2023). FastSpeech 2 generates "Neutral Audio", which the kNN-VC architecture converts into Speech Genre audio (see fig. 3).

FastSpeech 2 (Ren et al., 2022) is a non-autoregressive text to speech model which looks to tackle the "one-to-many" problem of TTS; that being that many speech sequences are possible from a single text input on account of various variations of speech. The method used in this architecture is to have various predictors assigned for these specific variations. The architecture begins with a text input which undergoes phoneme embedding and encoding into a hidden phoneme sequence. This sequence is subsequently fed into the variance adaptor, which is where multiple parameters are predicted independently. Below is an overview of this variance adaptor:

- Duration Prediction seeks to predict the duration of each phoneme. It uses the Montreal forced alignment (McAuliffe et al., 2017) to achieve this.

- Pitch Prediction seeks to predict the pitch contour of a given extract. This is done through decomposing a ground truth pitch contour into pitch spectrograms through continuous wavelet transform (CWT), and trained with a mean square error (MSE) optimisation (Hirose and Tao, 2015).

- Energy prediction seeks to predict the amplitude of a given sequence. This is done through representing the energy by the L2-norm of the amplitude of each Fourier transformed frame.

The output of variance adaptor then undergoes Mel-spectrogram decoding, which produces the output. Since a pre-trained model was used for evaluation, the specific architecture and hyperparameters are not discussed (refer to Appendix A of Ren et al., 2022). The implementation comes from Chien and Huang, 2020, which discusses any changes to the original FastSpeech 2 model. Most importantly, the Mel-spectrogram is decoded into audio through the use of the HiFiGAN Vocoder (Kong et al., 2020). The model was trained for 900,000 steps.

The kNN Voice Conversion architecture (kNN-VC) works through performing a kNN regression on extracted features from an utterance against a set of extracted features from a pool of references ("bag of vectors"). The intent of kNN-VC is any-to-any voice conversion, where the utterance comes from one speaker, and the references are samples from another speaker. The task is to exchange the voice from the utterance with the voice of the references. This is done through passing both the utterance and references through the WavLM Large Vocoder (Chen et al., 2022). The 6th transformer layer features are extracted, which are represented as 1024 dimensional vectors at 20 millisecond intervals. Each of these vectors undergoes kNN regression wherein the nearest k vectors are found through cosine similarity. A mean value is derived from these nearest k vectors, and this new vector substitutes the original input vector. Each utterance vector is replaced with a derived vector from the kNN regression. This new array of vectors is then put through the HiFiGAN Vocoder (Kong et al., 2020) where the 1024 dimensional vectors undergo dimensional reduction to 128, get converted into Mel-Spectrograms, and subsequently go on to be upscaled into audio form using a hop length of 10ms and a Hann window of 64ms. The architecture is visualised in fig. 7.

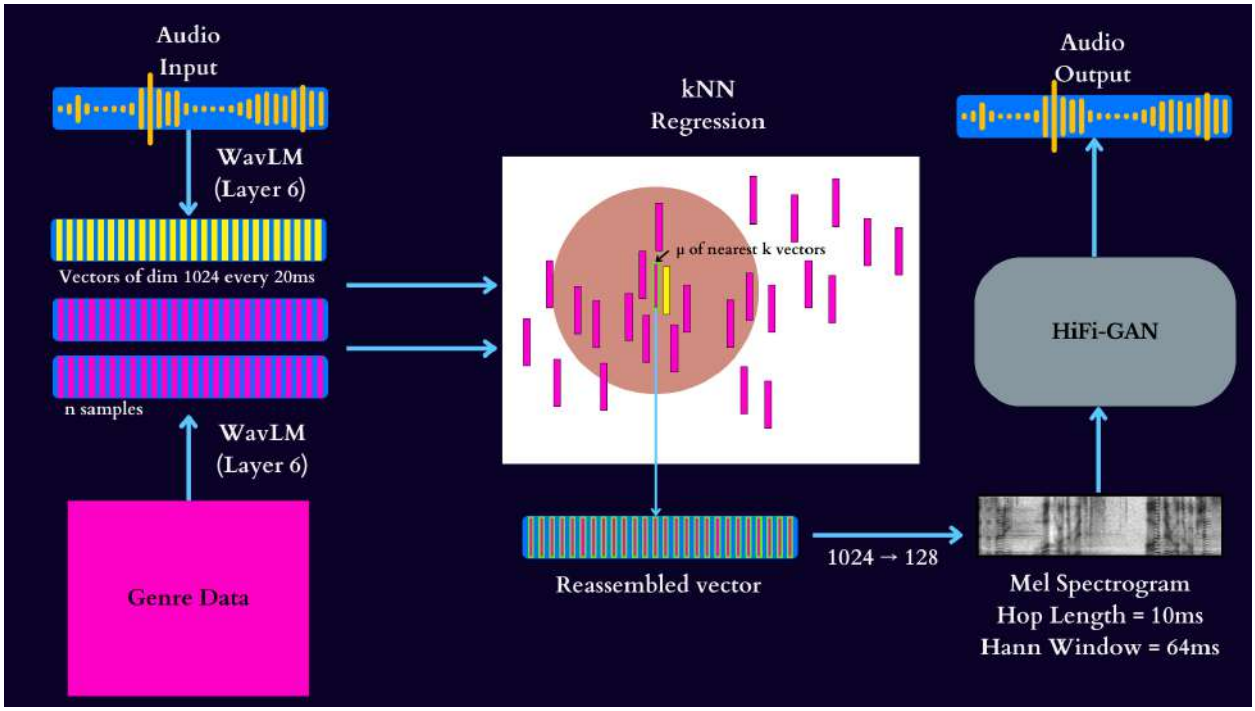

Figure 7: The prosody augmentation unit, the k-Nearest Neighbours Voice Conversion Architecture. Given the output of FastSpeech 2, and the prediction of sf_i by the text classifier, a corresponding sr_i is found. Both the sr_i and FastSpeech 2 samples are preprocessed by the WavLM vocoder, upon which kNN regression is done. The new vectors are subsequently turned back into audio by HiFiGAN.

Both FastSpeech 2, and kNN-VC (alongside the trained PyTorch files) are available in Appendix C.

5 Experimental Setup

5.1 RCNN Training & Testing Setup

The goal of the training was to have the RCNN trained on four different sets of training data, with each training set distinguished by character length. All four models were trained on the same four genres. All four models were trained for 5 epochs and the hyperparameters specified in Section 4.5.

After training, each model was tested on data from its own range, and also the other ranges. So, for example, the model trained on data in the 800-1200 character range would be tested on data from not only the 800-1200 character range, but also the 1000-1400, 600-1000 and the 200-600 ranges also.

Additionally, another test set of data was created using ChatGPT-3. Initially, the use of ChatGPT-3 was intended to be used as a means of data augmentation for genres which had insufficient training data. However, it was later repurposed as a separate dataset to see whether text classification would vary significantly from the data that was collected previously. The prompting was fairly straightforward; for each genre, ChatGPT-3 was asked: "Produce a/an (insert genre) text", with some minor modifications along the way if needed. The samples were determined by the paragraphs generated by ChatGPT-3; each paragraph was one datapoint for testing. A distribution of the character lengths of the ChatGPT-3 test data is shown below:

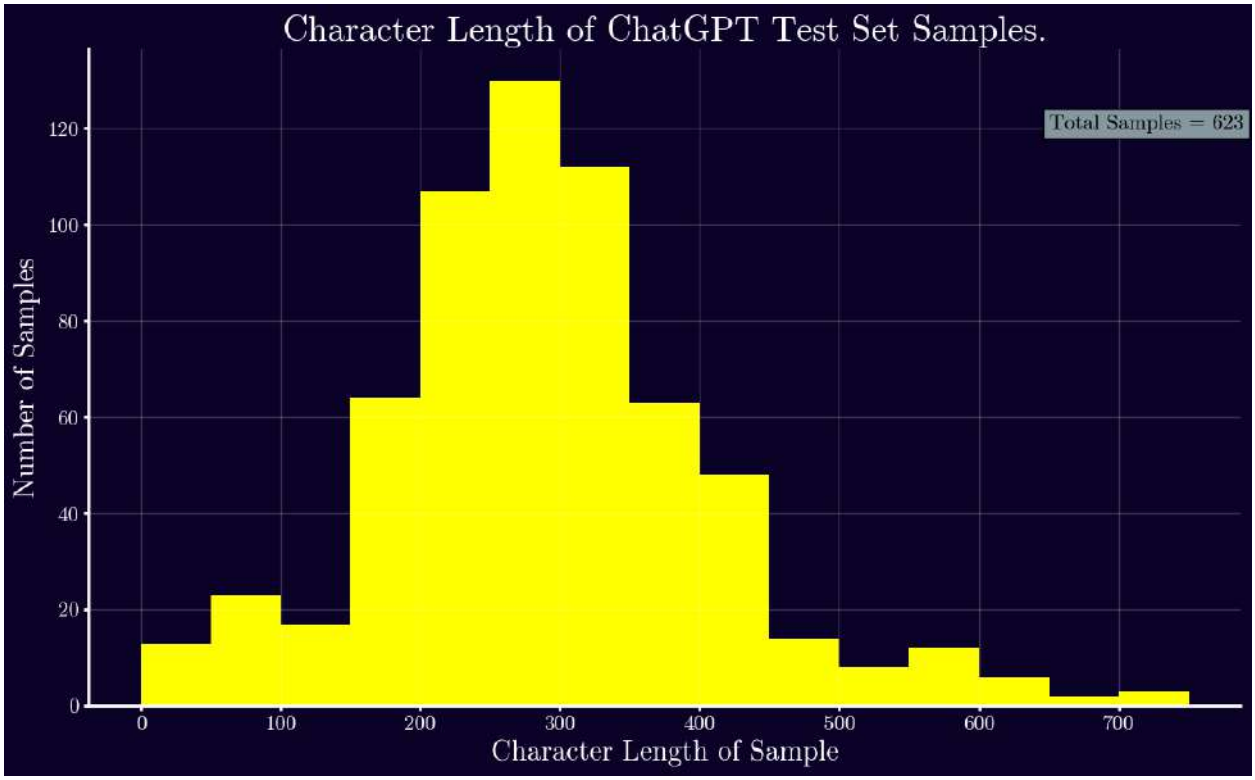

Figure 8: Character length distribution of the ChatGPT-3 test set.

The results for each test are shown as a a 4×4 confusion matrix, depicting accuracy (see section 6.1). From this, the recall and precision are analysed.

Recall is a metric which measures; of all instances where a classification is made (a Positive prediction), how much of them were correct (True Positives). For example, if a TEDTalk is predicted 80 times throughout a test set, recall asks how many of them are actually a TEDTalk. A higher value indicates that a model is successful in avoiding misidentification of a given class. This is a

useful metric to have alongside accuracy as if a model tends to overclassify one class, the accuracy of that class will still remain high as it will still classify said class correctly.

Precision is a metric which measures; of all instances of a class, how many of them were identified correctly. For example, if there are 80 TEDTalk samples within a test set, precision asks how many of these 80 were correctly predicted as being a TEDTalk. This is a useful metric to detect if a particular class is not easily recognisable by a model and whether a model conflates it with another class.

The F1 score is subsequently a harmonic mean of precision and recall. In this experimental setup, there is not necessarily a need to focus on either recall or precision specifically, thus the F1 score is weighted evenly across both factors.

These scores will also be given at genre specific levels to determine any differences (e.g. the precision of the 600-1000 character RCNN on TEDTalk extracts in the 1000-1400 character range).

5.2 Evaluation Setup

The goals of the evaluation is to determine the extent to which evaluators like or dislike the various speech registers produced by kNN regression Voice Converter across different contexts; and to determine whether evaluators are able to distinguish speech registers solely on prosody (that is, a sample without containing the speech function).

The experiment was run online. PsyToolKit was the software of choice (Stoet, 2010). The evaluators were selected from various internet spaces and social circles (such as LinkedIn and Whatsapp) through a link to the experiment. When the evaluator opened this link, they were greeted by an introductory text which explained their role as evaluators, and general background knolwedge of the research topic (see appendix D for the full text). In addition, they had to acknolwedge the following consent form:

```
I confirm that I am over the age of 18 and that I consent to taking
part in this survey. I understand that the results of my survey are
going to be used for scientific research.

I understand that I have the right to refuse the survey, to terminate
participation whenever I want, and to have my results deleted by
contacting the researcher.

I am participating voluntarily. I understand that no personal
information is required from me.
```

None of the participants had sufficient knowledge of the research topic (especially the hypotheses) to answer the survey in a manner which would be favourably biased towards the hypotheses. The participants had the option to terminate participation whenever they desired[7]. The survey also explicitly asked that all users be over the age of 18 to participate. This is not due to persons under the age of 18 being inherently inadequate evaluators, but rather to prevent the possession of data given by persons under the age of 18. Naturally, this is impossible to verify in practice. It is also assumed that all participants had an adequate comprehension of English to understand the experiment and all of the text and audio contained therein.

Prior to the experiment, the evaluators are asked about how long they spend listening to each of the four tested genres per week. This is collected to determine whether the time spent listening to

[7]Indeed the sample size goes down as the experiment progresses, which shows that the evaluators understood the option to terminate whenever they desired

a genre affected the ability of an evaluator to discern the genre, or affected their opinion scores. Evaluators were able to choose integer between “0 hours” and “10 hours or more”. The results are pooled into four groups: “0 hours”, “1-3 hours”,“4-6 hours”, and “7+ hours”. No other data about the evaluators was collected; each response was stored under a unique key which was untracable to the evaluators themselves. The evaluation is divided into two main experiments and three pilot experiments.

Experiment 1a: kNN vs. FastSpeech 2. In this experiment, there are four questions, one for every genre. Each question will pose four samples, which the evaluators have to rate on a ten point Likert Scale between one to five. Each of the four samples contains the same prompt; the text of which contains the speech function of the genre. The four samples are spoken by FastSpeech 2, the corresponding male speech register, the corresponding female speech register, and a dummy (different speech register). The point of this experiment is to determine whether there is a difference in the scores given between the FastSpeech 2 sample and the other samples.

Experiment 1b: kNN vs. kNN. In this experiment, the outline is almost identical to Experiment 1a, but there is no FastSpeech 2 sample. Thus, only the kNN samples are being compared against themselves. The point of this experiment is to determine whether there is a difference in the scores given to the kNN samples between Experiment 1 and 2 (whether the presence of a FastSpeech 2 sample significantly skews the user scores of kNN output).

Experiment 2: Discernability. In this experiment, there are eight questions, one for each genre and gender (4×2). For each question, a prompt is played. The prompts have a neutral linguistic content; they do not contain the speech register of any of the tested genres. This was achieved using the Harvard Sentences (IEEE, 1969). The evaluator is asked to identify the genre from the extract alone, with five options being available. The five options are composed of the correct answer, the ”I don’t know” option, and three decoy answers. The decoy answers consist of the other genres, and also additional genres which were not present at all (e.g. Poetic Reading, Political Rallying, Museum Exhibition, among others). The additional genres were introduced to reduce the possibility of evaluators guessing the correct genre from a 1-in-4 chance. Since there were five options per questions, guessing by chance would yield a discernability of 20%. Since this experiment was investigative, no threshold was set regarding a “desirable” discernability percentage.

The aforementioned experiments were the main experiments as they sought to investigate the main questions posed: whether people prefer speech with an applied speech register, and whether people could tell these registers apart. The pilot experiments are as follows, alongside their respective motivations:

Experiment 3: k value of kNN. Baas et al., 2023 proposed that with small amounts of audio (less than ten minutes), they found $k = 4$ to yield the best results, and that a setting $k = 20$ improved the quality of output when there was more than ten minutes of audio available. Since the intended application of Baas et al., 2023 was to have a bag of vectors of one single speaker, it was worth investigating whether these assumptions held true. As such, four questions were posed, one for each genre. Each question contained three instances of each sample, but the samples differed in k number (4, 20 and 50 respectively). Evaluators were again asked to rate each prompt on a ten point Likert Scale. Since each genre had more than ten minutes of audio, it is expected that $k = 20$ will yield a higher average user score than $k = 4$.

Experiment 4: Genre gender. An interesting query when investigating the perception of speech genres is whether the gender of the speaker entails any preferences. This section consisted of four questions. Each question had three prompts, the female register speaker, the male register speaker, and FastSpeech 2. All three extracts had the same speech function, and matched the

speech registers. Instead of a Likert Scale, the evaluators were asked to choose a favourite out of the three.

Experiment 5: Genre preference. This experiment is almost identical to Experiment 1b, but instead of the speech function being held constant across the four samples, it was the speech register which was held constant. There were four questions, one for each speech register. Each question had four prompts which were composed of the speech functions of all of the four tested genres. Evaluators were again asked to rate each prompt on a ten point Likert Scale. The point of this experiment was to see whether the inversion of register and function would have any significant differences in evaluator preference.

The experiment text file, and the audio samples used, are available in Appendix C

6 Results

The results will be broken down by first discussing the RCNN experiment results, followed by the evaluation results.

6.1 RCNN Accuracy

This section will present the training data first. Subsequently, the testing with equal length train & test data will follow; then testing with unequal data length, and then the ChatGPT-3 test set results. Finally, each result will be broken down per genre.

6.1.1 Training Results

The Validation Loss of all 4 models increased after the 3rd epoch. In general, the 200-600 model showed the highest Validation Loss throughout, whereas the 1000-1400 model showed the least. Since the original paper had used 10 epochs for training, and due to resource limitations, experiments were not performed to see whether Validation Loss would improve over a higher amount of epochs. The Validation loss at each epoch is depicted below:

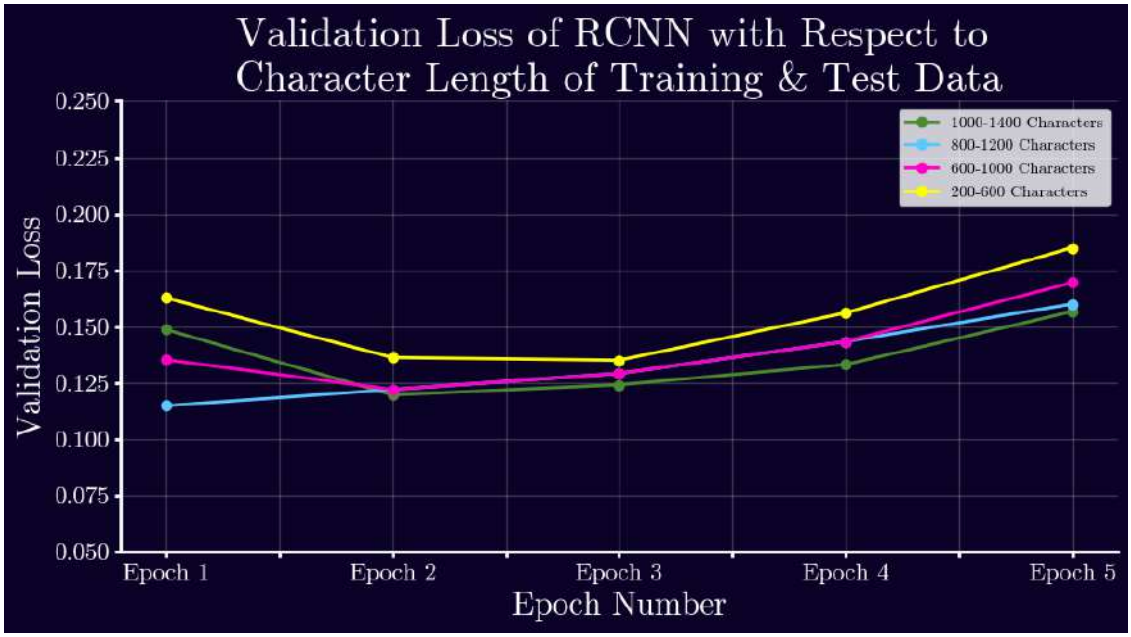

Figure 9: Validation Loss of each model across the 5 epochs.

The Accuracies of the 4 models were somewhat homogeneous, with all 4 models lying within the 94%-96% range across all epochs. In general, the 800-1200 model showed the highest accuracy, whereas the 200-600 model showed the lowest accuracy. The Accuracy at each epoch is depicted below:

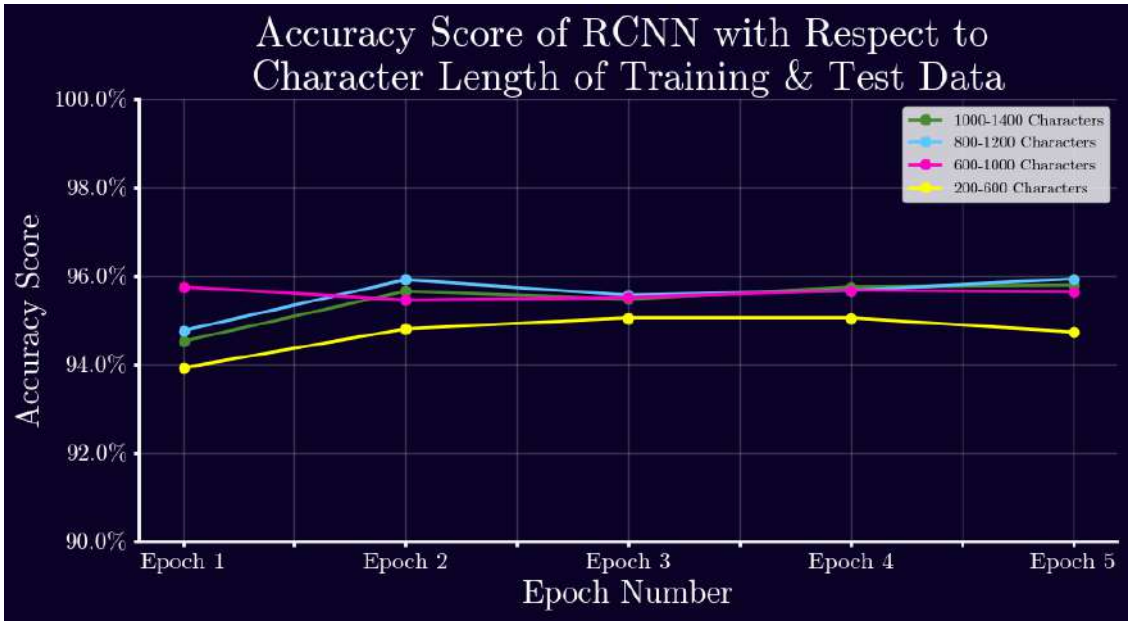

Figure 10: Accuracy of each model across the 5 epochs.

The F1 scores of the models are not as homogeneous as the Accuracies. Generally speaking, the lower character architectures tended to have lower F1 scores throughout training, with the 600-1000 model having an anomalously high F1 in the first Epoch. The F1 score at each epoch is depicted below:

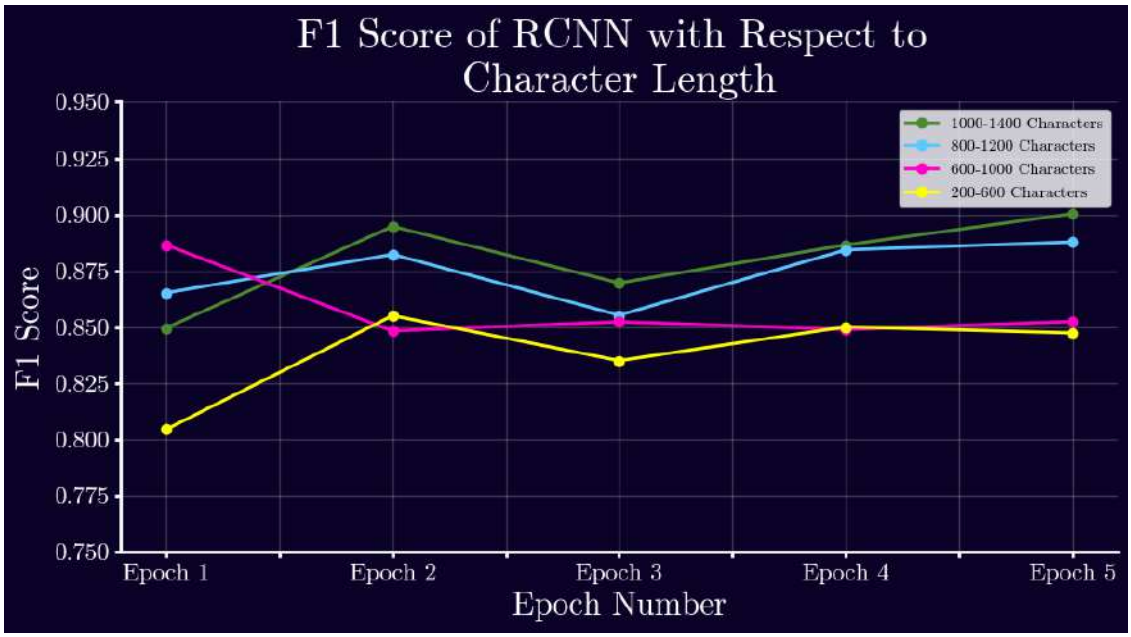

Figure 11: F1 score of each model across the 5 epochs.

Now, the confusion matrices of each model will be shown according to how well it performed across the four genres.

6.1.2 Equal Length Data

The 200-600 model performed well on the News and Comedy genres, achieving accuracies of 97.1% and 95.9%. However, it performed poorly on the Documentary genre, achieving an accuracy of 47.2%. It tended to conflate Documentary samples with a TEDTalk around 28.4% of the time, and with News 17.6% of the time. The full confusion matrix is plotted below:

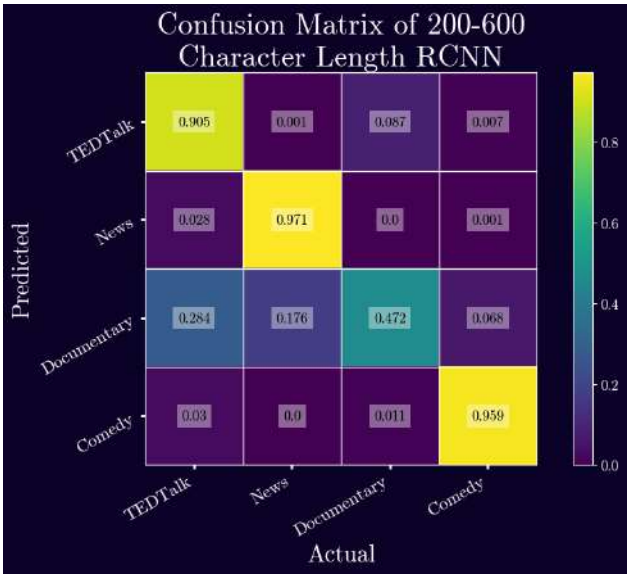

Figure 12: Confusion Matrix of the 200-600 Model across the 4 genres.

The 600-1000 model performed well across all but the Documentary genre, wherein it only achieved an accuracy of 55.9%. It performed better overall compared to the 200-600 model. The full confusion matrix is plotted below:

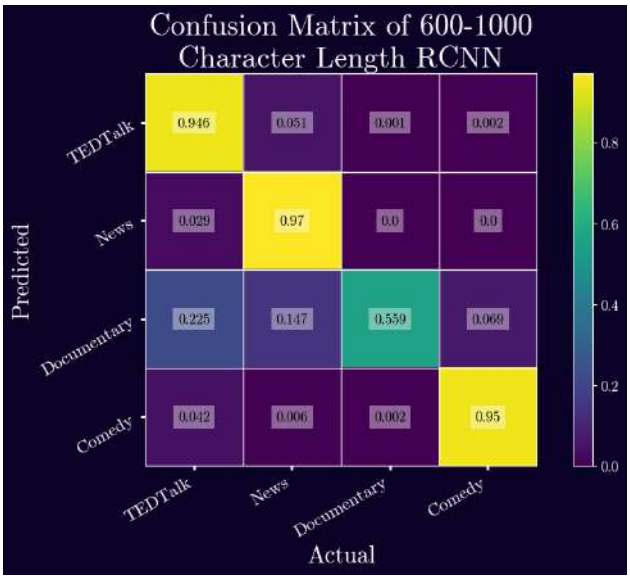

Figure 13: Confusion Matrix of the 600-1000 Model across the 4 genres.

The 800-1200 model performed similarly to the 600-1000 model, but with a higher accuracy for the Documentary genre (67%). This comes from a reduced ambiguity between the Documentary and News samples (14.7% conflation for the 600-1000 model as opposed to 7.7% for the 800-1200 model). The full confusion matrix is plotted below:

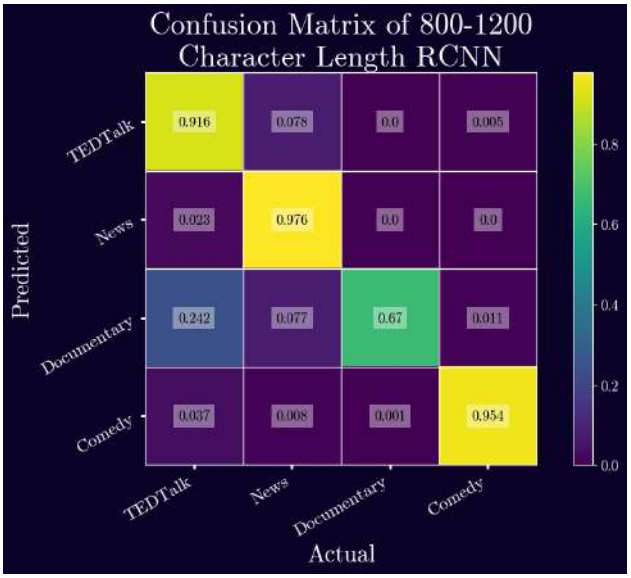

Figure 14: Confusion Matrix of the 800-1200 Model across the 4 genres.

The 1000-1400 model performed the best on average, but not by a large amount; with similar scores to the 800-1200 model.The full confusion matrix is plotted below:

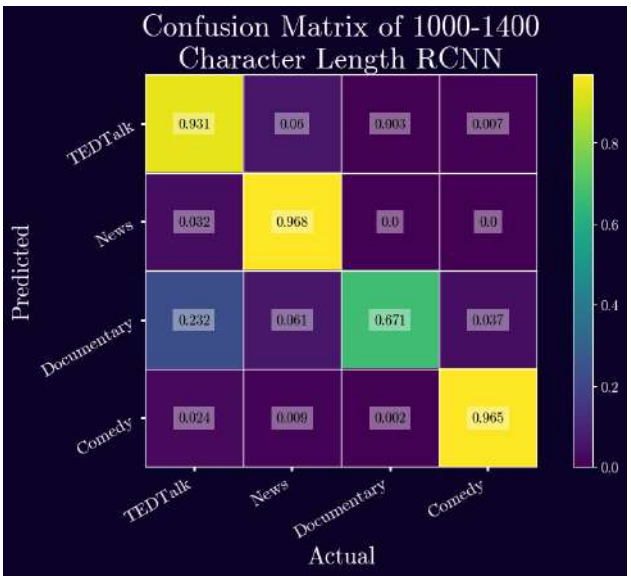

Figure 15: Confusion Matrix of the 1000-1400 Model across the 4 genres.

6.1.3 Unequal Length Results

Having tested each model on test data within its own range, the next set of tests examined how the models would perform on data from longer or shorter ranges.

In terms of accuracy, the 1000-1400 model tended to show the lowest accuracy across the 4 ranges whereas the 200-600 model showed the highest accuracy. All test scores are outlined below:

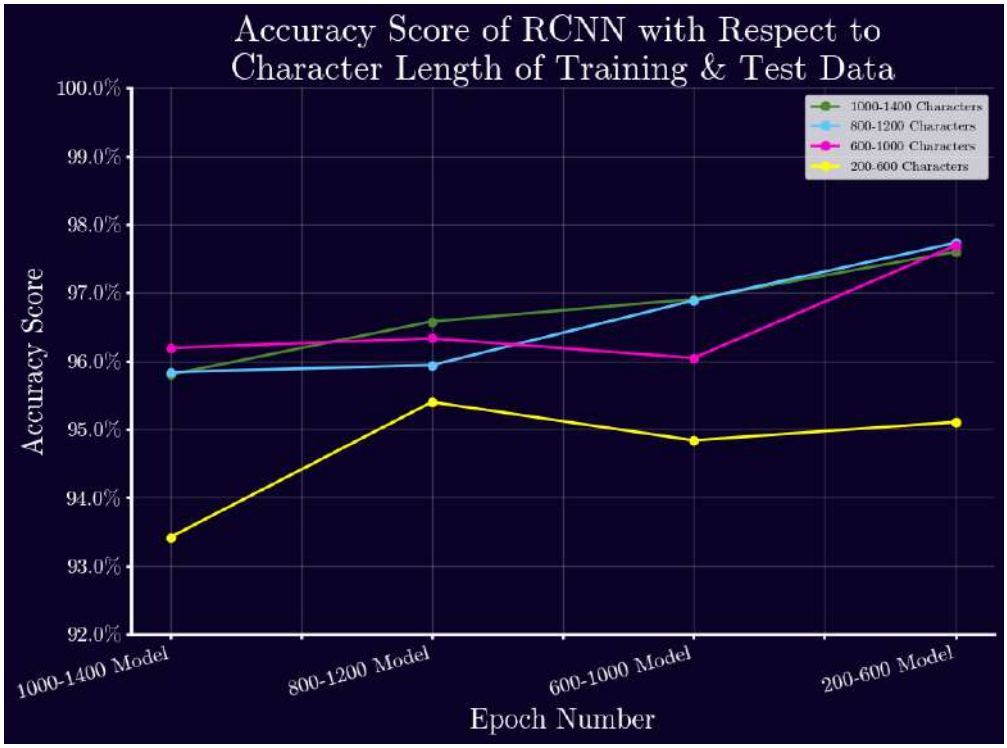

Figure 16: Accuracy scores of all 4 RCNN models across all 4 character length ranges.

The average recall showed more variance across models than the average accuracy. The lowest recall was shown by the 200-600 model tested on its own range (0.826), whereas the highest was shown by the 600-1000 model being tested on the 800-1200 range. All the figures are shown below:

Figure 17: The recall score of all four models across all 4 character lengths.

The average precision showed both a higher variance than the recall, but also a more noticeable pattern. The lowest recall was shown by the 1000-1400 model tested on the 200-600 range (0.765), whereas the highest was shown by the 200-600 model being tested on the 1000-1400 range (0.978). In general, models which were tested on higher character length samples than they were trained on performed better than on shorter character length samples. All of the figures are shown below:

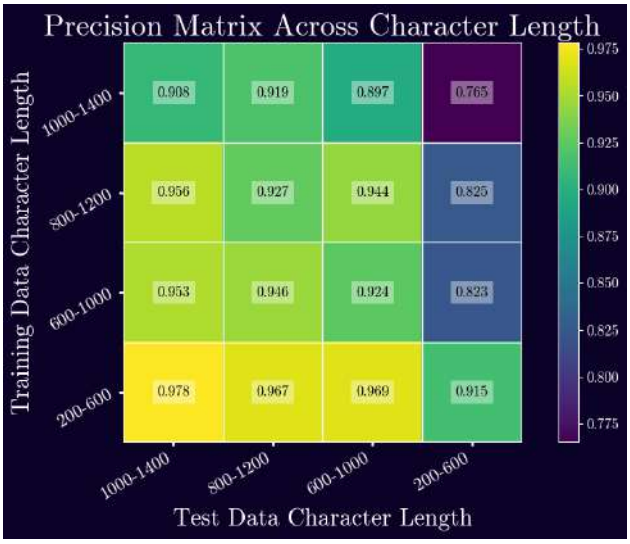

Figure 18: The precision score of all four models across all 4 character lengths.

Regarding the F1 score, we see that models trained on longer samples perform the worst on shorter samples, whereas models trained on shorter samples perform the best on longer samples. The lowest score was achieved by the 1000-1400 model tested on 200-600 character data (0.797) whereas the 200-600 model got the highest score on the 1000-1400 data (0.940). The full result matrix is shown below:

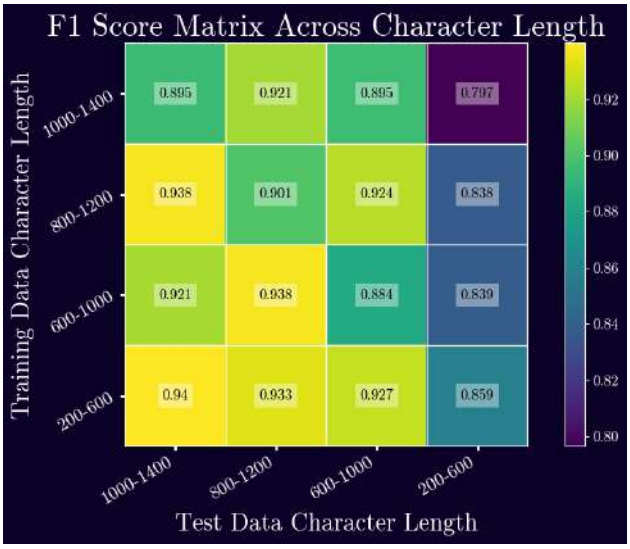

Figure 19: F1 scores of all 4 RCNN models across all 4 character length ranges.

If we take the upper quadrant (where training data character length is greater than testing data character length), the diagonal where both sets of data match, and the lower quadrant, we get 3 averages of 0.9267, 0.8847 and 0.8689 respectively. When performing a t-test to determine if the means of the upper and lower quadrants are significant, a p-value of 0.0305 is obtained.

6.1.4 Results per Genre

With the overall results disclosed, we can examine the results per genre specifically.

TEDTalk

The TEDTalk genre showed a high recall across all models and tests, with the lowest (200-600) recall achieivng a score of 0.905. All the results are shown below:

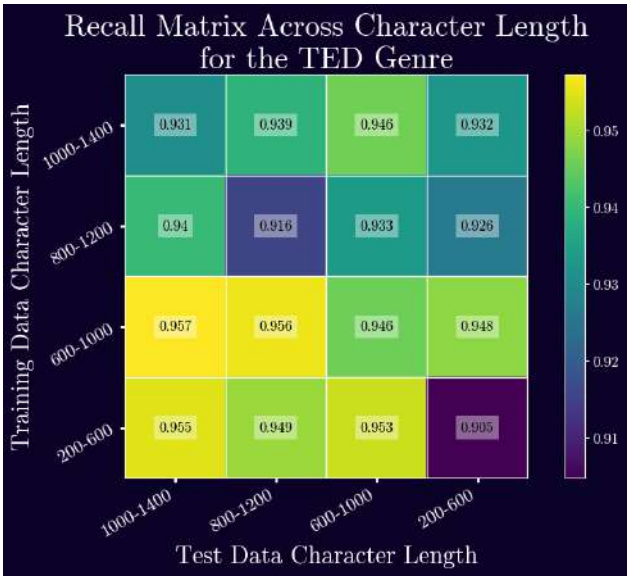

Figure 20: Recall scores of all models for the TEDTalk genre.

Regarding precision, the 200-600 model showed the highest ability, especially in the longer character length test sets. Simultaneously, the 200-600 test set proved difficult for the longer length models. The 1000-1400 model performed particularly worse with a score of 0.868. All the results are shown below:

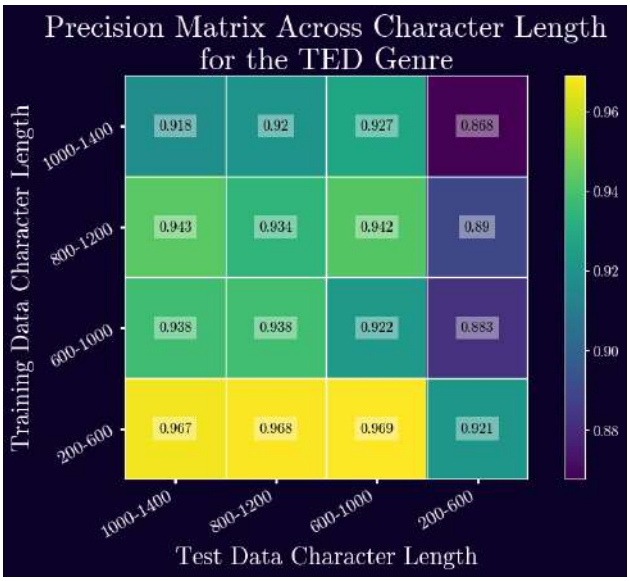

Figure 21: Precision scores of all models for the TEDTalk genre.

News

The News genre showed a consistently high recall, with the lowest score being 0.939 by the 1000-1400 model being tested on the 200-600 test set. The 200-600 performed best, achieving a recall of 0.988 across all higher length test sets. The full result matrix is shown below:

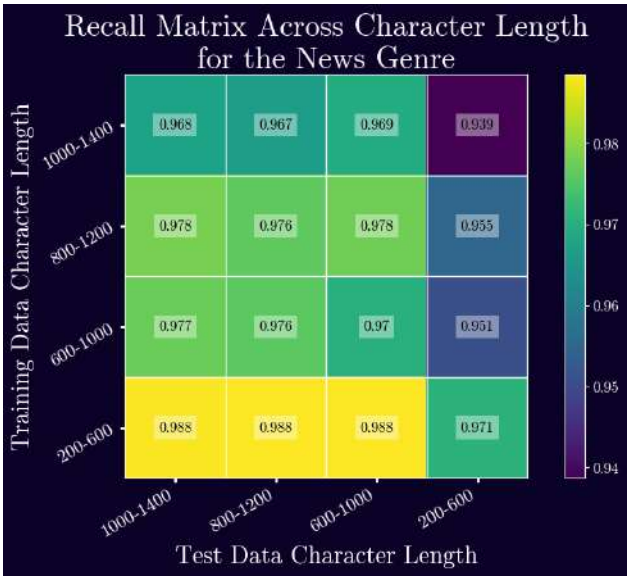

Figure 22: Recall scores of all models for the News genre.

The precision was also consistently high, with the lowest score being 0.961 by the 200-600 model on its own test set. The full result matrix is shown below:

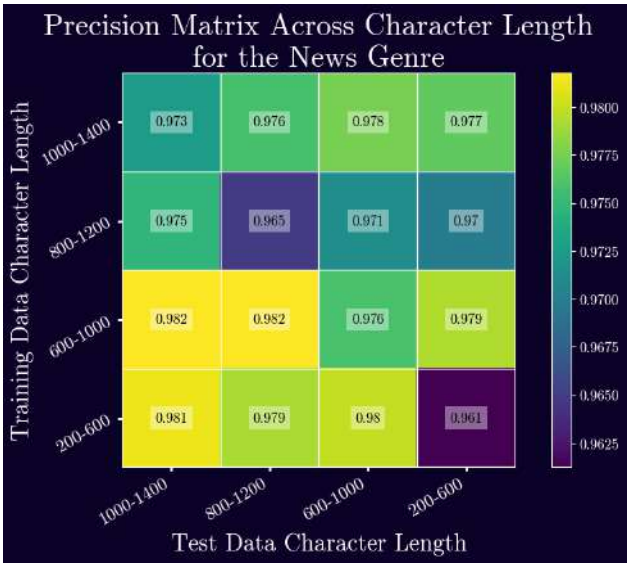

Figure 23: Precision scores of all models for the News genre.

Documentary

The Documentary genre showed the lowest average recall, and the greatest variance across models. The highest score was ahieved by the 600-1000 being tested on the 800-1200 sample (0.824) whereas the lowest was achieved by the 200-600 model on its own test set (0.472). The result matrix is shown below:

Figure 24: Recall scores of all models for the Documentary genre.

The precision is also low for the Documentary genre on average, but with significant variance. The highest score was achieved by the 200-600 model on the 1000-1400 test set (0.983) whereas the lowest was achieved by the 1000-1400 model on the 200-600 length test set (0.266). The 200-600 test set shows 3 distinct performance levels, with the 600-1000 and 800-1200 models achieving precision scores of 0.465 and 0.453 respectively. Contrastingly, the 200-600 model itself managed 0.814. The result matrix is shown below:

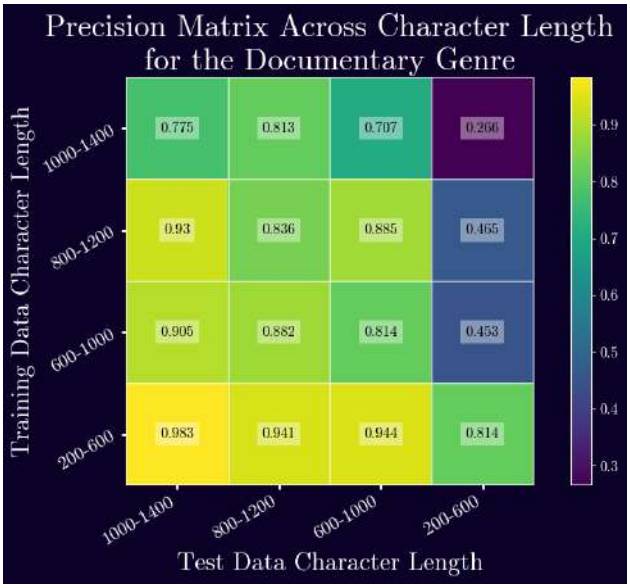

Figure 25: Precision scores of all models for the Documentary genre.

Comedy

The Comedy genre showed the highest average recall, with the lowest score being 0.94 achieved by the 800-1200 model on the 200-600 test set. The highest score was achieved by the 200-600 model on the 800-1200 test set. The result matrix is shown below:

Figure 26: Recall scores of all models for the Comedy genre.

The precision is similar to the News and TEDTalk genres, with a lowest score of 0.951 (1000-1400 model on the 200-600 test set) and a highest score of 0.986 (600-1000 model on the 1000-1400 test set). The result matrix is shown below:

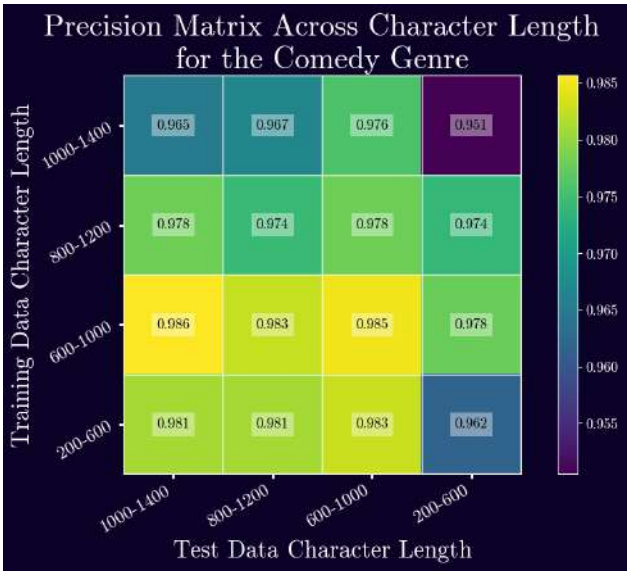

Figure 27: Precision scores of all models for the Comedy genre.

6.1.5 ChatGPT-3 Results

Finally, all models were tested with the ChatGPT-3 generated test set. Each model showed an extreme tendency to label almost every sample as a TEDTalk. The confusion matrix for the 600-1000 model is shown below.

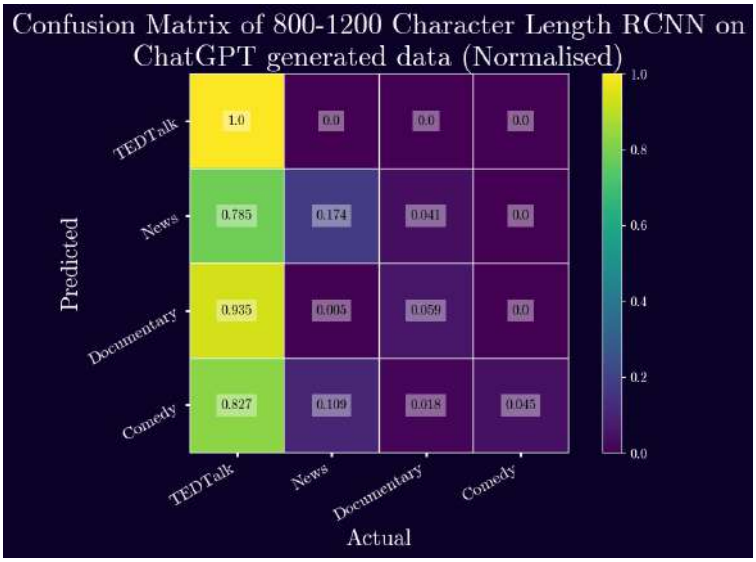

Figure 28: Confusion Matrix of the 600-1000 model being tested on the ChatGPT-3 generated dataset. The other models showed similar or worse performance.

6.2 Evaluation Results

This section will be presented in the same order as section 5.2, and additional results will be presented thereafter.

Experiment 1a

Experiment 1a tested the perception of genre speech against FastSpeech 2. With a sample size of $N = 19$, FastSpeech 2 achieved an average rating of 2.487 ± 0.284. The genre speech samples (male and female) achieved a rating of 2.783 ± 0.391 and 2.664 ± 0.248 respectively (with a combined average of 2.728). For each question, a dummy sample was included where the speech register did not match the speech function. The dummy samples achieved an average rating of 2.737 ± 0.245.

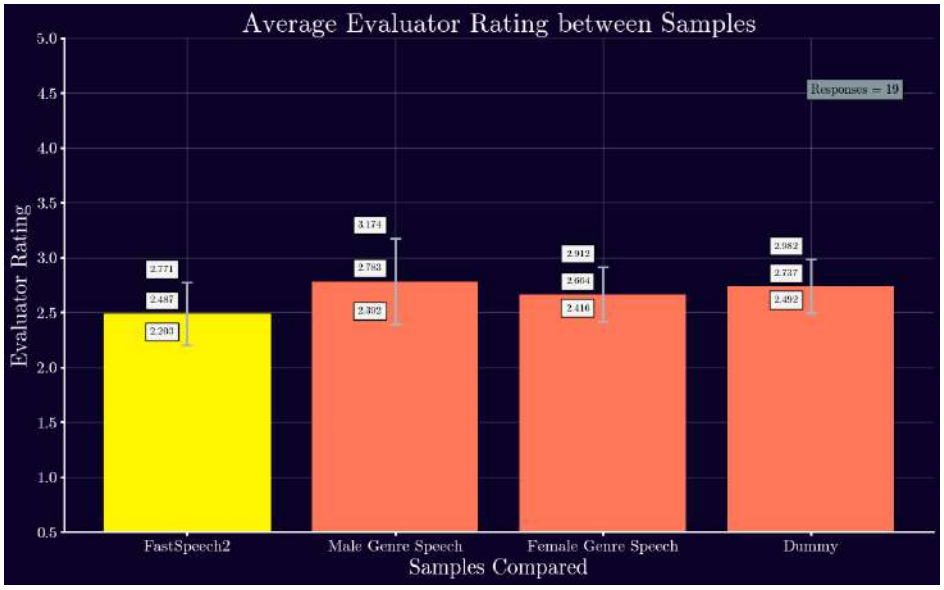

Figure 29: The results for Experiment 1a. The y axis shows the user rating from 1 to 5. Each of the four categories are shown, with the mean and standard deviation values included.

Experiment 1b

Experiment 1b was similar to Experiment 1a; with the only major exception being that FastSpeech 2 was not included within the samples. The analysis was split into two groups; one where the Speech Function matched the Register, and one where it did not match. With a sample size of $N = 16$, the samples where the Register and Function matched achieved an average rating of 2.992 ± 0.382; whereas the other group achieved an average rating of 2.864 ± 0.333.

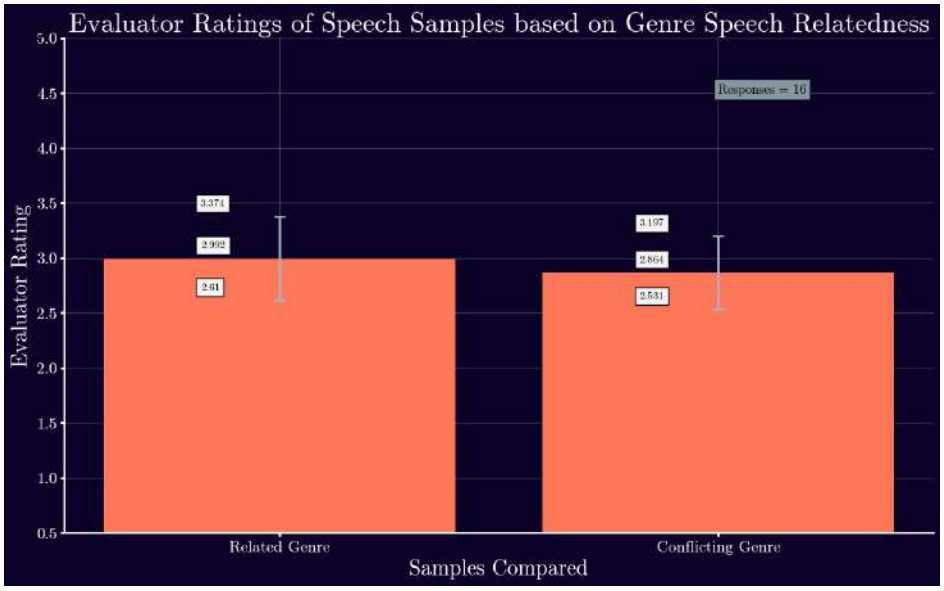

Figure 30: The results for Experiment 1b. The y axis depicts the evaluator rating (1-5). The four samples were grouped into two categories; Related Genre (Speech Function matches Speech Register) and Conflicting Genre (where they do not match).

Experiment 2

Experiment 2 tested whether evaluators were able to determine the Speech Register of a sample without the Speech Function. Since each of the questions included five options, random chance is set at 20% (or 25% depending on how the "Don't Know" option is considered). The average discernability across all 8 questions was 35.17%, with TEDTalk (Female) achieving the lowest discernability at 12.5%, and Comedy (Male) achieving the highest discernability at 62.5%

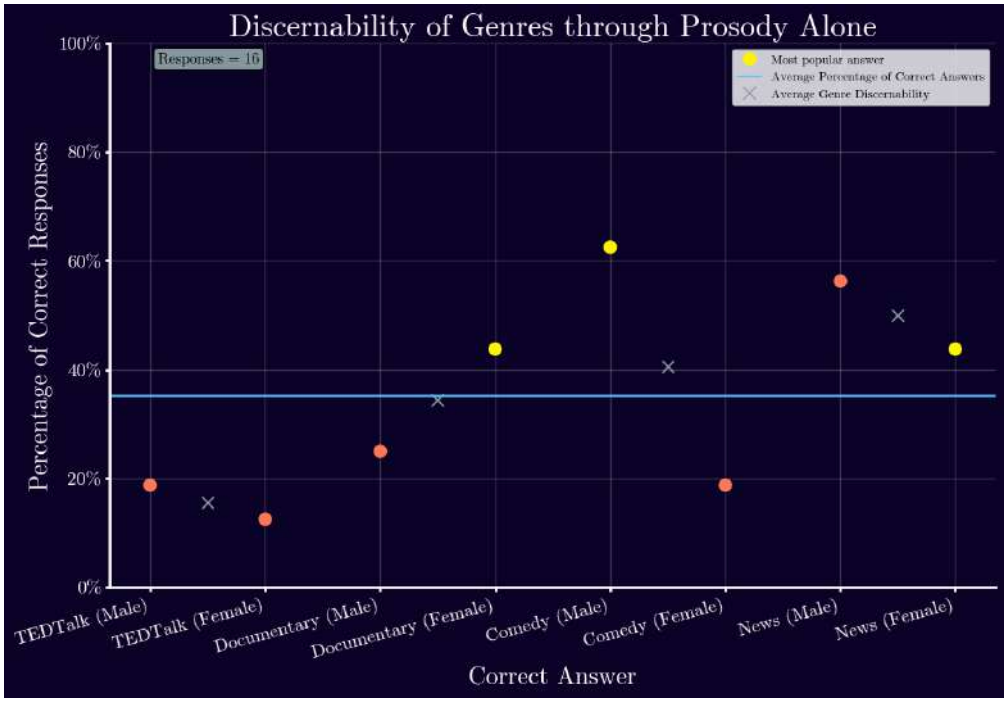

Figure 31: Discernability results for each of the tested genres, per gender. The y axis depicts the percentage of respondents who guessed the correct answer. The results are sorted per genre, and the grey x markings depict the average discernability of a genre. The blue line shows the overall average discernability score across all genres.

For the TEDTalk genre, the correct answer was only chosen 12.5% of the time, the lowest of all questions. 68.75% went to other genres which were seen previously, and 18.75% went to a genre the evaluators had not come across before. ”Don’t Know” was chosen 0% of the time. For the Documentary genre, while 43.75% of evaluators chose the correct answer, the majority (56.25%) chose a genre (Museum Exhibition) that was not even included within the sample list. TEDTalk (included), Political Rallying (not included) and ”Don’t Know” were chosen 0% of the time. The Comedy genre has an even larger disparity, with 18.75% choosing the correct answer as opposed to 75% choosing one of three genres (Motivational Speaking, Political Rallying and Weather Forecasting) which weren’t included. 6.25% chose the ”Don’t Know” option. The News Genre had 43.75% of evaluators choose the correct answer, with 50% choosing genres which weren’t included (Museum Exhibition and Instruction Guide) and 6.25% choosing the ”Don’t Know” option. Stand Up Routine (included within the set of sampled genres) was chosen 0% of the time.

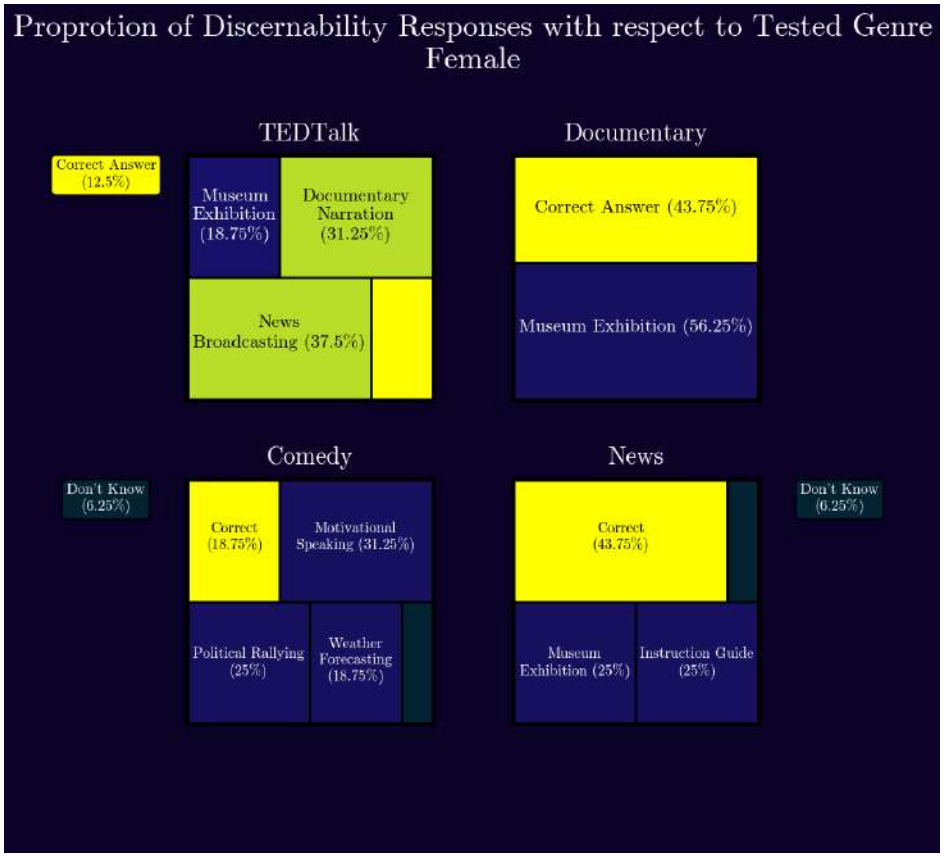

Figure 32: The answers given by evaluators for each of the genres, for samples containing the female genre speaker. Yellow indicates the correct answer, light green indicates another genre which was synthesised, dark blue indicates newly introduced genres and dark green indicates the ”Don’t Know” option. The options which were not chosen are not included in the graph.

For the TEDTalk genre, the correct answer was chosen 18.75% of the time. 50% of evaluators chose a genre which wasn’t included (Weather Forecasting) and 31.25% chose the ”Don’t Know” option, the highest of all questions. Sports Commentary and Political Rallying (both not included with the set of sampled genres) were chosen 0% of the time. For the Documentary genre, 25% of evaluators chose the correct answer. 25% majority chose a different genre from the included set of genres (News Broadcasting) and 43.75% chose a genre which wasn’t included (Poetic Reading and Public Speaking). 6.25% chose the ”Don’t Know” option. The Comedy genre showed the highest percentage of correct responses at 62.5%, with 37.5% choosing a genre which wasn’t included

(Public Speaking)[8]. News Broadcasting (included) and "Don't Know" were chosen 0% of the time. The News Genre had 56.25% of the evaluators choose a correct answer, 12.5% choose a different genre from the included set (Documentary Narration) and 31.25% choose a genre which wasn't included (Instruction Guide and Poetic Reading). "Don't Know" was chosen 0% of the time.

Figure 33: The answers given by evaluators for each of the genres, for samples containing the female genre speaker. Yellow indicates the correct answer, light green indicates another genre which was synthesised, dark blue indicates newly introduced genres and dark green indicates the "Don't Know" option. The options which were not chosen (0%) are not included in the graph.

Experiment 3

Experiment 3 tested whether the k value in the kNN regression had any significant impact on evaluator ratings. With a sample size of $N = 14$, the samples where $k = 4$ achieved an average rating of 3.000 ± 0.886; the samples where $k = 20$ achieved an average rating of 2.946 ± 0.800; and the samples where $k = 50$ achieved an average rating of 3.045 ± 0.847.[9]

[8]This option was included twice in the answer set erroneously.

[9]The standard deviation for Experiment 3 is significantly different from Experiments 1a and 1b. When using Levene's test to determine the variances of the answers were significantly different between Experiment 1a and 5, and Experiment 1b and 5; the p-values were 2.27×10^{-5} and 7.74×10^{-3} respectively.

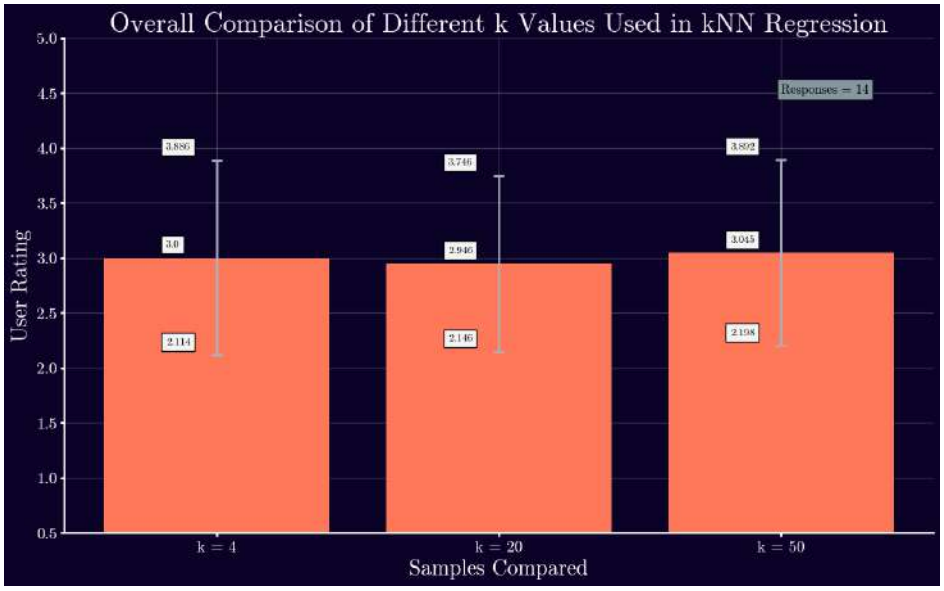

Figure 34: The average ratings between the three different *k* groups. The y axis depicts the the average evaluator rating from 1 to 5.

Experiment 4

Experiment 4 tested whether people had a preference for either the Male Genre speaker, the Female Genre speaker, or the FastSpeech 2 speaker. The Male Genre speaker was the most preferred sample with 55.4% of evaluators choosing the male speaker across the four genres. The Female Genre speaker was chosen 28.6% of the time whereas FastSpeech 2 was chosen 16.0% of the time.

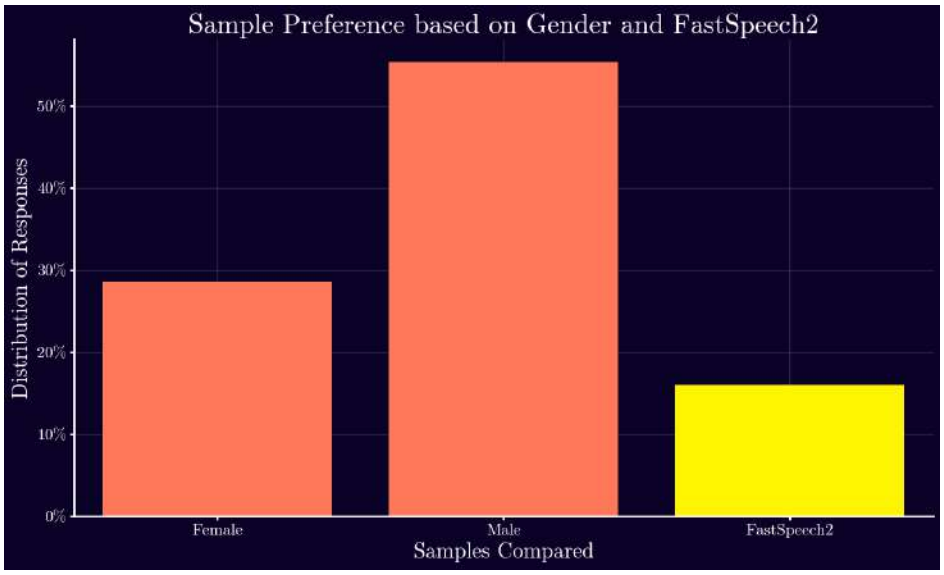

Figure 35: Sample preference across the three categories. The y axis depicts the three categories.

This pattern was not constant across all genres. For example, FastSpeech 2 was chosen more often for the Comedy genre than the Female Genre speaker; whereas in the Documentary genre, the Female Genre speaker was chosen more often than the Male Genre speaker. A graph showing the results per genre is shown below:

Figure 36: Gender preference results broken down per genre. The y axis shows the preference percentage per given sample.

Experiment 5

Experiment 5 tested whether evaluators had a preference for when a given Speech Register read a text with the matching Speech Function. Samples where the Speech Function and Speech Register matched achieved an average rating of 3.152 ± 0.949. Samples which had a mismatch achieved a rating of 3.015 ± 0.987.

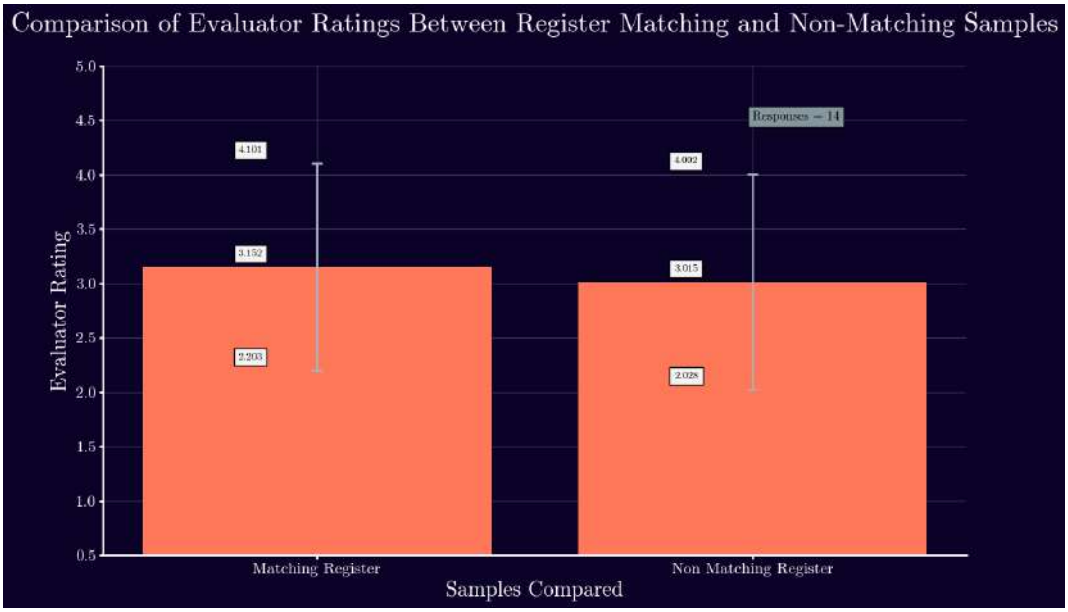

Figure 37: Evaluator ratings between register matching and non-matching samples. The non-matching bin consists of three options out of four, where the fourth is the matching register. The y axis depicts the evaluator scores from 1 to 5.

Other observations

It was mentioned that the evaluators were asked to disclose the extent to which they listened to each genre. This data can be used to test whether both the evaluator scores and discernability. First, the evaluator scores.

Across 64 responses (16 participants answering for 4 genres), 23 of them indicated listening 0 hours to a given genre; 34 indicated listening to between 1-3 hours, 5 indicated listening to 4-6

hours and 2 indicated listening to 7-10 (nobody indicated 10 or more). The results were divided between genre speakers (where the Speech Function matches the Speech Register) and "dummy" speakers. Genre speakers were further divided per gender.

The average rating for people who listened to 0 hours of a genre was 2.710 ± 0.961. The Dummy speakers tended to receive the highest rating (average = 2.891 ± 0.944), whereas the male genre speaker tended to get the lowest score (average = 2.522 ± 0.866). The average rating for people who listened to 1-3 hours of a genre was 2.716 ± 0.994. The scores across the three groups were almost identical. The average rating for people who listened to 4-6 hours of a genre was 3.167 ± 0.907. In contrast to the 0 hour group, the dummy speaker received the lowest average score (2.800 ± 1.030) whereas the female genre speaker received the highest score of (3.700 ± 0.510). The average rating for people who listened to 7+ hours of a genre was 2.083 ± 1.096. The scores across the three groups were almost identical, with the female genre speaker receiving a slightly higher average of (2.250 ± 1.250).

Figure 38: Evaluator ratings of samples with respect to time spent listening to said genre. The y axis depicts the evaluator score from 1-5. All four genres are pooled together. The results are split between the male genre speaker, the female genre speaker, and a dummy. The N value is per category, so the total N value per group is 3 times the stated N value. FastSpeech 2 is not included.

The average discernability of a genre for people who listened to 0 hours of said genre was 47.826%. The Male News sample showed the highest discernability at 80% whereas the Female Comedy sample showed the lowest discernability at 14.286%.

The average discernability of a genre for people who listened to 1-3 hours of said genre was 41.176%. The Male Comedy sample showed the highest discernability at 71.429% whereas the Female Comedy sample showed the lowest discernability at 14.286%. When analysing the different between the 1-3 hour group and the 0 hour group, the Male TED sample showed the largest decrease in discernability (-32.143%) whereas the Female Documentary sample showed the high-

est increase (+30.556%)

The average discernability of a genre for people who listened to 4-6 hours of said genre was 30.0%. The sample size is too small to make comparisons per genre. Likewise, the sample size for people that listened to more than 7 hours of a genre is also too small to include. An illustration of the average discernability is shown below.

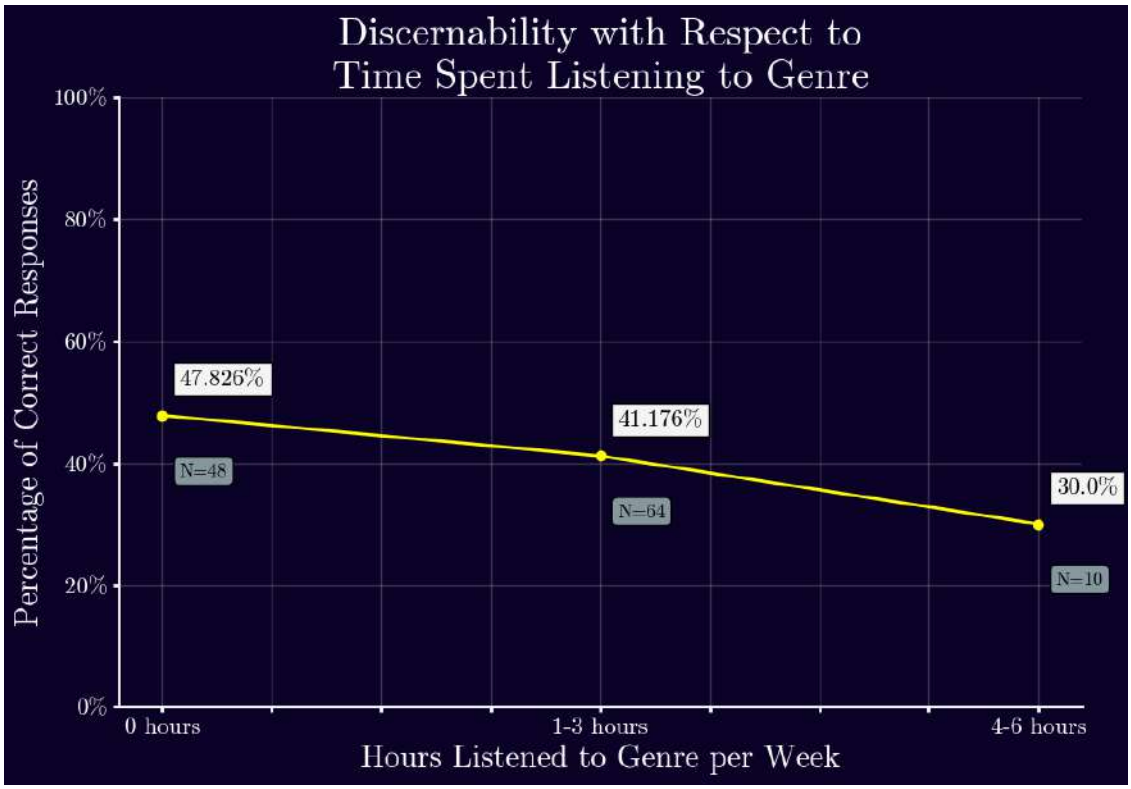

Figure 39: The average discernability of a genre with respect to time spent listening to a given genre.

7 Discussion & Conclusion

7.1 Addressing the Research Question & Hypotheses

This section will be addressed in the same order that the hypotheses were given in Section 2.

RCNN Accuracy & F1 Score

The accuracy for each RCNN model where the character lengths of the train and test data was homogeneous, was between 0.94 and 0.96, which is on par with the value of 0.95 stated in the hypothesis, meaning that the results are consistent with Lai et al. (Lai et al., 2015). However, for experiments where the test length was higher than the train length, the average accuracy was higher than 0.95, the highest being achieved by the 200-600 model on the 800-1200 dataset (0.976). Regarding F1- scores, an RCNN will perform near or above an F1 score of 0.9 when trained on data between 800-1200 characters or lower, or if the RCNN is used on data which has a higher average character length than its training data. The results are thus aligned with the hypothesis. This result was also expected as the literature review (section 2) showed that most text classification tasks report accuracies and F1-scores around 0.9.

Since the dataset used in the experiment has not been used in any other experiment, it is difficult to determine whether the RCNN would perform equally well across other datasets. It is possible that higher quality datasets could be made that could correspond to higher F1 scores. In contrast to Lai et al., the RCNN in this paper only had to distinguish between 4 classes rather than 20, which could exaggerate accuracy scores.

RCNN Character Length

Whereas the F1 score for homogeneous train and test data was 0.8847, the average F1 score for experiments with a longer test character length was 0.9267, contrary to the hypothesis. Experiments with a shorter test character length indeed performed worse than experiments with homogeneous train and test data. The results are thus considered to be misaligned with the hypothesis. This type of experiment has not been done in previous literature, thus no direct comparisons are possible.

One potential reason for this is that an input with a higher character count represents an input which contains more clues of a given genre. In the same way that a certain problem can seem easier if one trains on more difficult versions thereof, the RCNN may have an easier time identifying a longer length input having been trained to extract data out of fewer words.

RCNN Performance based on Genre

The confusion matrices for experiments with homogeneous train and test data indicated that the RCNNs struggled with the Documentary genre accuracy. The recall scores for all the experiments are also noticeably lower for the Documentary genre than the other 3 genres. The precision figures were also lower but not to the same extent; the lower quadrant of the Precision Matrix for the Documentary genre (fig. 25) was still typically above 0.9. The results are thus considered to be generally aligned with the hypothesis.

It is possible that, with the improvement of auto-captioning technology of video sharing websites such as YouTube, that the quality of text data retrieved from videos may improve to a point where it may be as usable as text which is scraped from pre-existing text. The lower accuracy and F1-score of the Documentary genre can be almost entirely attributed to the lack of a pre-existing text dataset of documentary scripts. The dataset created for the Documentary genre had the lowest quality of all datasets used in this paper (misidentification of words, improper full stop placements, and so on).

kNN Voice Conversion vs. Base FastSpeech 2

The average MOS score of FastSpeech 2 samples was 2.487, whereas the average MOS score of all varieties of kNN Voice Conversion samples was 2.728 (fig. 29). For comparison, the lowest reported MOS score for any of the FastSpeech 2 implementations from the original FastSpeech 2 paper was 3.68 ± 0.09 (Ren et al., 2022). When evaluators were asked to select a preference between FastSpeech 2 and 2 kNN Voice Conversion samples, FastSpeech 2 was chosen 16.0% of the time. While the results generally aligned with the hypothesis, the sample size is insufficient to claim that kNN Voice Conversion is preferred over FastSpeech 2 samples. In order to draw more definitive conclusions on this question, the evaluation would have to be repeated with more evaluators. A larger sample size could also allow for more specific demographic analysis (whether age, sex, country of origin, or any such metric has any significant impact on the results). Similarly, a higher sample size would be required to make any conclusions regarding which genres tend to perform better or worse when synthesised.

A potential reason for why the FastSpeech 2 scores were in this experiment than the scores from the original FastSpeech 2 paper could lie with the progression and development of speech synthesis technology since the original paper was published. If we assume newer systems to be better (which tends to be the case as seen in section 2), then it makes sense that the scores of FastSpeech 2 will decrease over time. The limitations of MOS pointed out by Le Maguer et al. (Le Maguer et al., 2024) are also applicable here.

Speech Register Discernability

The average Genre Discernability was 35.17%, which is higher than if evaluators guessed randomly for each question (25%). While the result is higher than the hypothesis threshold, the results aren't sufficient to determine whether human evaluators can determine the Genre of an extract through the Register alone. Since no other piece of literature has pursued Speech Genre identification through Speech Registers alone, there is no other result that can be used for comparison.

On the one hand, the result also shows that 64.83% of the time, the evaluators were unable to guess the genre of a sample; which could be reasonably considered a poor result. On the other hand, Experiment 2 explicitly introduced additional genres in order to avoid an inflated discernability based on having listened to the 4 genres in Experiment 1. The best experiment to shed light on these results is to repeat it with human speech; which is to say that non-genre text should be read out (by people who are trained to speak a specific genre) applying the various Speech Registers. If such results would be equally poor, then it would indicate a general inability to discern a genre through prosody alone. If the results would be better, it would point to an inadequacy of the synthesised samples (whether it be the data or the architecture).

Gender Preference

In Experiment 1a, the average MOS for male samples was 2.783, whereas for female samples it was 2.664. In experiment 4, evaluators tended to prefer the male sample 55.4% of the time, whereas the female sample was preferred 28.6% of the time. While the results generally aligned with the hypothesis, the evidence is not sufficient to claim that male samples were preferred over female samples (due to sample size). Previous results on the matter are also mixed; with some papers failing to find a significant difference between preference of male and female speakers (e.g. Eadie et al., 2008), and others showing a preference for (male) speakers with a lower pitch and faster speaking tempo (e.g. Quené et al., 2021).

With the perception of gender varying across cultures (see Costa et al., 2001 for example), or an evolving perception of the gender binary (see Hyde et al., 2019 for example). It is difficult to determine to what extent any gendered analysis is applicable to different cultures and contexts,

and whether gendered analysis will remain a worthwhile consideration. Although an interesting parameter to investigate, no broader conclusions can be made on the basis of the results of this experiment.

k Value of kNN Voice Conversion

The average MOS value of samples with k=20 was 2.946, compared to 3.0 for k=4 and 3.045 for k=50.

These results do not align with Baas et al., 2023, which argued that for a large set of speakers, k=20 was the optimal range. The results also don't explicitly disprove this notion either, it merely seems to suggest that the value of k is unimportant for user opinion scores in Speech Registers.

7.2 Framework Motivation

This section will be structured as independent questions that could be levied against the proposed framework.

Why is there a need for Automatic Speech Function Classification?

In many applications of speech synthesis, the kind of input text is already known. For example, if one were to program a TTS function for an audiobook website, there are tags available for the genre of the book, the author, and other metadata that give sufficient information pertaining to the kind of text that is being given. Similarly if a TTS function is being implemented for a news website to read the articles aloud, the information is already there that it is a news article, the genre of news (sports, business, etc.), who wrote it, and so on. This again, is adequate information pertaining to how such a text should be read.

Additionally, for a given body of text, it is very unlikely to be the case that the register would change dramatically, if at all. Thus it may be seen that there is not need for Speech Function detection as the Speech Function is expected to remain homogeneous throughout an entire text. A documentary would not suddenly transform into a continuity announcement, or a poetic reading.

In some cases, such as a voice assistant, the input text may have to come from various different sources. Using the voice assistant example, it may be asked for the weather in the morning, then to have the morning news read aloud, then perhaps for it to have a joke told, then to read a child a story, and so on. All of these come from totally different functions and registers, and would stand to suffer from being delivered in a homogeneous, monotonous delivery. Since the current speech synthesis systems, both in speech synthesis (Ren et al., 2022, Meng et al., 2025, Adibian and Zeinali, 2025) and in expressive speech synthesis (Skerry-Ryan et al., 2018, Y. Wang, Stanton, Zhang, Skerry-Ryan, et al., 2018b, X. Li et al., 2021, Baas et al., 2023) cannot inherently distinguish a joke from a story or any other function of speech, the could not adapt the prosody of synthesised output to accommodate those functions.

Classification of text also removes dependence of speech synthesis systems on providing explicit arguments (such as style tokens from Skerry-Ryan et al., 2018), and can directly synthesise from only the information provided by the text. This is a step forward for speech synthesis as the conventional model asserts that speech synthesis has no idea of the meaning of a text, it merely breaks a text down to phonetic level and by various machine learning methods, figures out the best sound to create given the input text. While Automatic Speech Function Classification does not lead to speech synthesis systems that understand texts (or develop any meaningful semiotic relation), it does bring machine learning models closer to a more human-like understanding of language. While it may not understand why a joke is funny, it may recognise that a joke is not the same entity as a news report (so for example, a news event, and a joke about a news event would be distinguished) as opposed to merely seeing anything and everything as "text" with no further

insight.

It also yields opportunities for hybrid systems of text generators and synthesisers. Speech function detection is an NLP task, and can thus be used for other NLP tasks. One example is text generation; where one may ask to generate a comedy stand-up routine. Another is sentiment analysis, where a text classifier may explicitly forbid the synthesis of a given text extract if it detects that a text extract has a negative sentiment, if it contains profanity, or if it contains hate speech. Speaking of omitting speech, it may sometimes be useful to categorise which text should not be synthesised.The earlier concept of a "Neutral" class $sf_{\varnothing}$ can be made very useful here. If the input text is from a news website, it may contain other text other than the news, such as the headlines of other articles, texts related to advertising, hyperlinks, and other text which isn't important to the news. The ability to classify them as $sf_{\varnothing}$ could be used to explicitly prevent synthesis of this unimportant text, which could improve the user experience of the architecture.

Is the additional effort required to implement such an architecture worth it?

TTS tasks rely on solely having an adequate text pre-processing and an architecture which can produce intelligible and natural speech. The proposed framework would, aside from incorporating a Text Classifier, also require data to be segmented according to different classes, possibly require the collection of new data to synthesise a unique genre, separate training for each of the classes, and a far more robust evaluation system which is not as readily interpretable as more conventional applications.

The additional effort put into the aforementioned tasks allows for assets to be created which are usable in broader contexts and scopes. A dataset which is classified according to genre can be used for architectures which are specifically tailored for one specific genre (in the case where Text Classification is indeed deemed to be redundant). Such classes of data would not only contain known information about prosody and text, but may also be consistent in additional features. For example, a documentary class may contain background music and multiple speakers, and thus a dataset which is categorised to be in a documentary class can also be used to train other architectures such as background noise cancellation (for a paper such as Nikitaras et al., 2022) and speaker diarisation (for a paper such as Kanda et al., 2022). A dataset classified in a storytelling class can be used to train emotional speech synthesis (for a paper such as K. Zhou et al., 2023). With adequate documentation and methodology, the necessary preparations of data and architecture can extend their application beyond the specific use case. Investigations of evaluations metrics are also at the forefront of speech synthesis (Wagner et al., 2019), thus the framework also incentivises a core research area of speech synthesis.

Instead of using Speech Registers, why not just rely on Vocoders to capture speech prosody?

When using readily available speech synthesisers of various celebrities, their speech is already generated with their particular genre of speech. This owes to the fact that the training data is taken from when the person is performing the speech genre that they are known for. Famous documentary narrators have data of them narrating documentaries, news casters have data of them giving the news, poetry readers reading poetry, and so on. With the current state of the art speech synthesis architectures, they are able to adequately capture the prosody of this data. Thus instead of using Functions and Registers, we may simply find data with a desirable voice for a chosen type of text, and train the architecture accordingly.

The aim of is to have a framework that is necessarily speaker independent (as opposed to approaches seen by papers such as Niu et al., 2025, Choi et al., 2022, Huang et al., 2023). One speaker, regardless of how perfect they are deemed to be for a given genre of speech, is necessarily limited by anatomy, training (if applicable), vocabulary, language, and so on. If the architecture is

trained on the basis of functions instead of a speaker, it can be more readily fine tuned to use any speaker. Particularly, if the context of application is within a particular company or website which seeks to have a unique and distinguishable voice, it makes more sense for the starting point to be a generic, customisable architecture that understands the text, rather than an architecture which is trained using the genre of one, unique speaker.

Not only this, but it also aids genre versatility. If the prosody and speaker are separated as two different entities, then the physiological parameters of the speaking voice used for the speech synthesis output can be used to generate any register desired so long as it is adequately trained.

To summarise, the combination of TC and speech synthesis stands to be a beneficial next step for many applications of SS. While such a combination may not be necessary in every conceivable context of SS, the steps taken in preparing such an architecture can also be made useful in other contexts as well, such as the kind of data assembled, and the individual NLP and speech synthesis architectures which are trained. On account of how varied the means of evaluation may be, how differently the speech genres may be defined, and the different potential contexts of application, this framework is necessarily not a specific solution, but rather a general problem space, so that any subsequent research can append architectures, data, and means of evaluation as they shall see fit. This paper acknowledges that the future will likely improve on all 3 aspects, and thus should not necessarily restrict the solution only to what is currently available as of writing.

7.3 Limitations & Future Work

Above all, a major limiting factor in this study and the ability to extrapolate results is that the specific theoretical framework used to guide the experiments is not present elsewhere within the literature space, meaning that these experiments have not been done before, resulting in an absence of previous results. While there are previous results in how well the performance of each of speech synthesis architectures (Ren et al., 2022, Baas et al., 2023) performed against other architectures, there are no results in how well they (or any other architecture) perform in the generation of speech synthesis. Genre discernability is equally a new pursuit within speech synthesis without comparable results (at least none that have been available for the analysis of this paper). Similar points can be raised concerning the effect of k value of kNN-VC speech genre generation, the effect of male vs. female speakers per speech genre, and the effect of previous familiarity of speech genres; on the discernability and preference patterns of speech genre output. These are all points which would benefit from future research.

As stated in the Discernability section of the Theoretical Framework (section 3.2.2), if a human listener cannot tell the difference between different registers (or is apathetic to them), then the pursuit of synthesised registers is trivial. This idea can also be extended to the idea that if there is no meaningful preference between the "correctly" applied register compared to either a "conflicting" register, or no register at all, the the pursuit of register synthesis is also trivial. The results (Experiment 1b, Experiment 2, Experiment 5) show limited, but non-conclusive evidence that there is an ability for human evaluators to distinguish speech registers, and prefer audio samples which have a matching register applied to the function of the text. Aside from the idea that the lack of preference and discernability coming from an inherently flawed approach, a result indicating a lack of preference and/or discernability could also be attributed to either an adequate speech register dataset or insufficient speech synthesis architecture (in either generating the neutral audio, or the speech register audio). Further testing could focus on measuring user satisfaction of a speech function being synthesised with flat delivery as opposed to its appropriate speech register.

The entire framework is contingent on having speech genres which are broadly understood and recognisable. With the introduction and definition of the specific speech genres introduced in this paper, alongside the selection of text and audio which is supposed to represent those defined gen-

res, carries an implicit assumption that the broader public has the same understanding of those speech genres, and agrees that the text and audio used for training the architecture is indeed representative of said genres. The experiments did not confirm this, there is no way of knowing whether the defined genres are even considered sufficiently distinct to constitute a designation of a genre. Future works could direct query human evaluators about their opinions of the genres to be presented, perhaps including a means to provide a qualitative description of those genres (e.g. giving names of people associated with the genre, specific shows/television channels, etc.). If there is any form of variation of the genre, either by means of language, culture, context or otherwise, which is not accounted during data assembly or evaluation, then there can arise a reduction in discernability or preference which is not accounted for.

Finally, while the framework provides a means of using text and audio to synthesise speech genres, it does not provide a means to combat certain challenges of speech production, such as de-noising poor audio; cleaning noisy text data; handling multilingual data; pronunciation of symbols, abbreviations, acronyms and numbers; and many other issues found in speech synthesis (Kuligowska et al., 2018). There are also aspects of the framework itself which would require future clarification, such as what constitutes a sufficient number of speakers to define a speech genre; determining the effect of the inclusion of genre neutral text on text classification; how to handle multilingual data (and how the experiments of this paper would fare in other languages); an analysis of the effectiveness of the different means of evaluation presented in fig. 2, the effectiveness of the accuracy metrics as proposed in fig. 1; and the effects of testing a specific speech function with multiple speech register (e.g. synthesising a news broadcast function with a formal news register, satirical news register, radio vs. television news registers, etc.).

7.4 Generalisation of the Theoretical Framework

This section will first deal with relaxing some of the assumptions outlined in the theoretical framework, and subsequently suggest expanding the problem space to include any categorisation of text and speech, not just speech genres.

Expanding the Framework

To reiterate the theoretical framework; each problem space is assumed to have n speech genres. Each speech genre has one Speech Function and one Speech Register. The architecture tasked with converting the text of the Speech genre into the Synthesised Speech is a combination of a Text Classifier and a Speech Synthesiser. We seek to maximise the accuracy of classifying Speech Functions by the Text Classifier, and aim to maximise both discernability, and the Mean Opinion Scores, of the synthesised speech of the Speech Synthesiser.

Let's first address expanding the one-to-one mapping between SF and SR. A many-to-one ($n(SF) > n(SR)$) system could be used wherein we may recognise different Speech Functions, but may be content in having them share the same Speech Register. An example of this would be the aforementioned similarity between a news broadcast over a radio, and news broadcast over television. We can motivate their different Functions just from the communication media alone, but we may also acknowledge that there is scarcely any acoustic differences in the Speech Registers. A reduction in SR may also be made if some of the previous sr_i fail the discernability criterion. For example, if we have a set of text for Nature Documentaries, and another set of text for Science Documentaries, but their Registers prove indistinguishable in evaluation, they may instead be synthesised under a common "Documentary" register. In such a case, the distinction between the different SF is rendered functionally redundant as the additional effort put into separating them makes no different to the final synthesised output. Thus the many-to-one is a weak form of expansion.

A one-to-many ($n(SR) > n(SF)$) problem space could be used where a Speech Function is ex-

pected to be expressed differently based on different conditions. Consider the following examples:

- A News Speech Function may be realised differently depending on the country of broadcast. The Speech Register may vary due to cultural reasons, but may also be different on account of the accents of the news broadcasters.
- A novel or poem may be spoken differently depending on its genre.
- Characters with synthesised voices in video games will have a different Speech Register depending on their role within the video game.
- Satire and Ironic media will necessarily mimic their true counterparts, but then also deviate in Speech Register to define their role as satire/ironic.

It can be argued, similarly to our many-to-one case, that we could simply modify the SF change to reflect the aforementioned nuances. This would yield a system which would distinguish between accents, genre of text, and so on. However this may be a time-consuming process. What could be done instead, is that an identified sf_i is synthesised into the various sr_i associated with it, and the desired a_i can be decided afterwards either systemically or subjectively.

Referring back to the issue of subjectivity, we may choose to represent SR groups as a spectrum rather than a single, concrete set of parameters for a given SF. For example, a comedy stand-up genre may have a varied set of prosodic parameters depending on the genre of comedy. What could be done, is to include a $sr_\varnothing$ class, even if we do not wish to classify a corresponding $sf_\varnothing$. If we use an sr_i as one end of a spectrum, and $sr_\varnothing$ as the other end, we can establish a spectrum of parameters, which could be interpreted as the extent to which a given text sample corresponds to our genre of choice.

Going beyond functions and registers, there are potential examples where text classification could be used to influence the output of speech synthesis. Firstly, text classification could assist the synthesis of accents and dialects. Given an English text, there are a few different accents which could be used to synthesise the corresponding speech. If we assume that different English speaking regions have sufficiently unique vocabularies then the accent could be applied to the speech synthesis without having to manually apply an accent label. The use of eye dialect (spelling a certain word incorrectly to better match a specific pronunciation) could also benefit automatic accent/dialect detection. Such a text classification can be used within the current research on synthesising speech accents (such as Liu et al., 2024, X. Zhou et al., 2024, among others) by providing accent information directly from the text without requiring audio sample analysis (or being used alongside audio analysis). Secondly, text classification could be used for emotion prediction. Going back to the main focus of expressive speech synthesis (to emulate the range of affective expressions present in natural speech), the emotion of a given piece of text could be determined from the text itself, and then applied to the synthesised speech. The inclusion of exclamation marks, capital letters, extra repeated letters, tone indicators (e.g. /s for sarcasm) could not only guide the emotion, but also the extent of emotion synthesised. Such a text classification can be used within the current research on emotional speech synthesis (such as Lei et al., 2022, Lei et al., 2021, among others) by adding another (text based) predictive parameter for emotion prediction.

With the progress of deep learning, there are possibly many other ways to combine the architectures of text classification and speech synthesis across a wide variety of contexts and applications.

7.5 Ethical Considerations

Since this project seeks to replicate various types of functional speech, it stands to reason that a potential use of this research would be to synthesise speech in areas such as voice acting, news

reporting, radio broadcasting, and other speech based industries. A temptation can arise for entities which own organisations such as news stations or media companies to use Speech Genre synthesis in lieu of human speech, which poses a potential risk of threatening the employment of people working in those industries (news anchors, commentators, presenters, etc.). This paper does not support the mass removal of human workers in any speech related employment by means of artificially generated speech.

The evaluation process is not expected to have brought any experiences which would be a cause of concern from an ethical perspective. For example, it is not expected that anyone was upset, angry, offended; or experienced any adverse effects from the act of evaluation. However it was still important to ensure that the evaluation process was transparent and comfortable. Participants were informed about the nature of the study, that they could cease evaluation at any stage that they wanted, and that their results could be deleted upon request. Explicit permission to process the data of a given evaluator was sought, with further communication information provided should it be necessary.

While no new data was collected (every bit of text and audio was already publicly available elsewhere), it is a fact that the people behind the text and audio may not be necessarily aware of their implicit involvement in the training of architecture used in this paper. While of the data comes from publicly available sources, it is unlikely that every person which encompasses the data could have been aware that their writings or voice would be used in the training of artificial intelligence. In that regard, this research paper avoided the use of data from any parties which could be negatively affected (e.g. children, people with speaking disabilities, etc). Each piece of data comes from a source where the person knew that they would be broadcast in some way, shape or form to a mass audience. This also somewhat motivated the specific kinds of genres chosen in this paper.

The architecture used in this research paper can also be classified as artificial intelligence, which has been shown to have a large carbon footprint (Kirkpatrick, 2023), which contributes to worsening global climate conditions (Intergovernmental Panel on Climate Change, 2023). Training of architectures and production of audio samples were therefore limited to avoid excessive computation time and resource use.

7.6 Conclusion

This paper successfully combined speech synthesis with text classification in the pursuit of Speech Genre synthesis. It shows that Speech Function classification can be accomplished by means of text classification, and that Speech Genres synthesis can be achieved by means of currently available speech synthesis architectures. The main direct use of the framework and architecture presented in this paper is the generation of synthetic speech from a text item which does not have a corresponding human spoken counterpart available, and which also is not contingent on the mannerisms of one given speaker, but rather the congregated average of multiple speakers tasked with delivering a particular kind of speech.

The accuracy of the speech function classification was comparable with previous literature, and the synthesised speech samples using speech registers had more preferable evaluator ratings compared to standard speech synthesis. Evidence is also presented that the character length of the training and test data used in text classification alters the accuracy of text classification, which can be used for future text classification architectures. With a limited number of evaluators, a newly devised dataset, and a new theoretical framework, the strength of these results is limited, requiring further research for context.

The framework and results presented in this paper demonstrate broader applicability of using text data as a prosodic parameter for synthesised speech generation, with potential uses in emotion

prediction and accented speech synthesis. The framework also highlights potential future directions for subjective speech synthesis evaluation and introduces the concept of "discernibility" when testing the ability of evaluators to distinguish between different types of speech. The results also indicate that the use of text data generated by large language models yields poor text classification results.

References

Adibian, M., & Zeinali, H. (2025). End-to-end multi-speaker fastspeech2 with hierarchical decoder. *IEEE Access*, *13*, 127805–127814. https://doi.org/10.1109/ACCESS.2025.3589120

Aladhami, N. A. A. (2024). An overview concerning the monologue and stand-up comedy. *Arab World English J. Transl. Lit. Stud*, *8*(1), 112–121.

Baas, M., van Niekerk, B., & Kamper, H. (2023). Voice conversion with just nearest neighbors. https://arxiv.org/abs/2305.18975

Bakhtin, M. (2011). The problem of speech genres. *THE PROBLEM OF SPEECH GENRES*.

Baroni, M., Dinu, G., & Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In K. Toutanova & H. Wu (Eds.), *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 238–247). Association for Computational Linguistics. https://doi.org/10.3115/v1/P14-1023

Bataev, V., Ghosh, S., Lavrukhin, V., & Li, J. (2025). Tts-transducer: End-to-end speech synthesis with neural transducer. *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. https://doi.org/10.1109/ICASSP49660.2025.10890256

Cai, F., & Ye, H. (2023). Chinese medical text classification with roberta. In S. Wen & C. Yang (Eds.), *Biomedical and computational biology* (pp. 223–236). Springer International Publishing.

Catenaccio, P., Cotter, C., De Smedt, M., Garzone, G., Jacobs, G., Macgilchrist, F., Lams, L., Perrin, D., Richardson, J. E., Van Hout, T., & Van Praet, E. (2011). Towards a linguistics of news production [Discursive Perspectives on News Production]. *Journal of Pragmatics*, *43*(7), 1843–1852. https://doi.org/https://doi.org/10.1016/j.pragma.2010.09.022

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Wu, J., Zeng, M., Yu, X., & Wei, F. (2022). Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, *16*(6), 1505–1518. https://doi.org/10.1109/jstsp.2022.3188113

Chen, S., Wang, C., Wu, Y., Zhang, Z., Zhou, L., Liu, S., Chen, Z., Liu, Y., Wang, H., Li, J., He, L., Zhao, S., & Wei, F. (2025). Neural codec language models are zero-shot text to speech synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, *33*, 705–718. https://doi.org/10.1109/TASLPRO.2025.3530270

Chien, C.-M., & Huang, C.-Y. (2020). An implementation of microsoft's "fastspeech 2: Fast and high-quality end-to-end text to speech". https://github.com/ming024/FastSpeech2

Choi, H.-S., Yang, J., Lee, J., & Kim, H. (2022). Nansy++: Unified voice synthesis with neural analysis and synthesis. https://arxiv.org/abs/2211.09407

Costa, J., Paul, Terracciano, A., & McCrae, R. R. (2001). Gender differences in personality traits across cultures: Robust and surprising findings. *Journal of Personality and Social Psychology*, *81*(2), 322–331. https://doi.org/10.1037//0022-3514.81.2.322

Cotter, C. (1993). Prosodic aspects of broadcast news register. *Proceedings of the Nineteenth Annual Meeting of the Berkeley Linguistics Society: General Session and Parasession on Semantic Typology and Semantic Universals*, *19*. https://doi.org/10.3765/bls.v19i1.1520

Dawson, K., Antonenko, P., Lane, H., & Zhu, J. (2019). Assistive technologies to support students with dyslexia. *TEACHING Exceptional Children*, *51*(3), 226–239. https://doi.org/10.1177/0040059918794027

Dementyev, V. (2016). Speech genres and discourse: Genres study in discourse analysis paradigm. *Russian Journal of Linguistics*, (4), 103–121.

Eadie, T. L., Doyle, P. C., Hansen, K., & Beaudin, P. G. (2008). Influence of speaker gender on listener judgments of tracheoesophageal speech. *Journal of Voice*, *22*(1), 43–57. https://doi.org/https://doi.org/10.1016/j.jvoice.2006.08.008

Egbert, J., & Mahlberg, M. (2020). Fiction – one register or two?: Speech and narration in novels. *Register Studies*, *2*(1), 72–101. https://doi.org/https://doi.org/10.1075/rs.19006.egb

Erickson, B., Lind, E., Johnson, B. C., & O'Barr, W. M. (1978). Speech style and impression formation in a court setting: The effects of "powerful" and "powerless" speech. *Journal of Experimental Social Psychology*, *14*(3), 266–279. https://doi.org/https://doi.org/10.1016/0022-1031(78)90015-X

Freitas, D., & Kouroupetroglou, G. (2008). Speech technologies for blind and low vision persons. *Technology and Disability*, *20*(2), 135–156. https://doi.org/10.3233/tad-2008-20208

Giles, H., & Powesland, P. F. (1975). *Speech style and social evaluation*. Academic Press.

Halliday, M. A. K., McIntosh, A., & Strevens, P. D. (1964). *The linguistic sciences and language teaching*. Longmans, Green.

Hartmann, J., Heitmann, M., Siebert, C., & Schamp, C. (2023). More than a feeling: Accuracy and application of sentiment analysis. *International Journal of Research in Marketing*, *40*(1), 75–87. https://doi.org/https://doi.org/10.1016/j.ijresmar.2022.05.005

Hengst, T. (2021). *The influence of text to speech voice gender on the listening experience for hard and soft news delivery* (Doctoral dissertation). Tilburg University. Communication and Cognition.

Hirose, K., & Tao, J. (2015). *Speech prosody in speech synthesis: Modeling and generation of prosody for high quality and flexible speech synthesis*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-45258-5

Holmes, J. (2013). *An introduction to sociolinguistics* (4th). Routledge. https://doi.org/10.4324/9781315833057

Huang, R., Zhang, C., Wang, Y., Yang, D., Liu, L., Ye, Z., Jiang, Z., Weng, C., Zhao, Z., & Yu, D. (2023). Make-a-voice: Unified voice synthesis with discrete representation. https://arxiv.org/abs/2305.19269

Hunter, S. B., Mathews, F., & Weeds, J. (2023). Using hierarchical text classification to investigate the utility of machine learning in automating online analyses of wildlife exploitation. *Ecological Informatics*, *75*, 102076. https://doi.org/https://doi.org/10.1016/j.ecoinf.2023.102076

Hyde, J. S., Bigler, R. S., Joel, D., Tate, C. C., & van Anders, S. M. (2019). The future of sex and gender in psychology: Five challenges to the gender binary. *American Psychologist*, *74*(2), 171–193. https://doi.org/10.1037/amp0000307

IEEE. (1969). Ieee recommended practice for speech quality measurements. *IEEE No 297-1969*, 1–24. https://doi.org/10.1109/IEEESTD.1969.7405210

Intergovernmental Panel on Climate Change. (2023). *Climate Change 2023: Synthesis Report. Summary for Policymakers* (tech. rep.) [doi: 10.59327/IPCC/AR6-9789291691647.001]. Geneva, Switzerland, IPCC.

Kanda, N., Xiao, X., Gaur, Y., Wang, X., Meng, Z., Chen, Z., & Yoshioka, T. (2022). Transcribe-to-diarize: Neural speaker diarization for unlimited number of speakers using end-to-end speaker-attributed asr. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8082–8086. https://doi.org/10.1109/ICASSP43922.2022.9746225

Kern, F. (2010). Speaking dramatically: The prosody of live radio commentary of football matches. In D. Barth-Weingarten, E. Reber, & M. Selting (Eds.), *Prosody in interaction* (pp. 217–238). John Benjamins Publishing Company. https://doi.org/doi:10.1075/sidag.23.18ker

Kim, J. (2019). Jungwhank/rcnn-text-classification-pytorch: Pytorch implementation of "recurrent convolutional neural network for text classification". https://github.com/jungwhank/rcnn-text-classification-pytorch

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. https://arxiv.org/abs/1412.6980

Kirkpatrick, K. (2023). The carbon footprint of artificial intelligence. *Communications of the ACM*, *66*(8), 17–19.

Klatt, D. (1982). The klattalk text-to-speech conversion system. *ICASSP '82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, *7*, 1589–1592. https://doi.org/10.1109/ICASSP.1982.1171431

Kong, J., Kim, J., & Bae, J. (2020). Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. https://arxiv.org/abs/2010.05646

Kortmann, B. (2020). *English linguistics: Essentials*. J.B. Metzler. https://books.google.nl/books?id=0Bd4zQEACAAJ

Kuligowska, K., Kisielewicz, P., & Włodarz, A. (2018). Speech synthesis systems: Disadvantages and limitations. *Int J Res Eng Technol (UAE)*, *7*, 234–239.

Lai, S., Xu, L., Liu, K., & Zhao, J. (2015). Recurrent convolutional neural networks for text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, *29*(1). https://doi.org/10.1609/aaai.v29i1.9513

Le Maguer, S., King, S., & Harte, N. (2024). The limits of the mean opinion score for speech synthesis evaluation. *Computer Speech Language*, *84*, 101577. https://doi.org/https://doi.org/10.1016/j.csl.2023.101577

Lei, Y., Yang, S., Wang, X., & Xie, L. (2022). Msemotts: Multi-scale emotion transfer, prediction, and control for emotional speech synthesis. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *30*, 853–864. https://doi.org/10.1109/TASLP.2022.3145293

Lei, Y., Yang, S., & Xie, L. (2021). Fine-grained emotion strength transfer, control and prediction for emotional speech synthesis. *2021 IEEE Spoken Language Technology Workshop (SLT)*, 423–430. https://doi.org/10.1109/SLT48900.2021.9383524

Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P. S., & He, L. (2022). A survey on text classification: From traditional to deep learning. *ACM Trans. Intell. Syst. Technol.*, *13*(2). https://doi.org/10.1145/3495162

Li, X., Song, C., Li, J., Wu, Z., Jia, J., & Meng, H. (2021). Towards multi-scale style control for expressive speech synthesis.

Li, Y. A., Han, C., & Mesgarani, N. (2025). Styletts: A style-based generative model for natural and diverse text-to-speech synthesis. *IEEE Journal of Selected Topics in Signal Processing*, *19*(1), 283–296. https://doi.org/10.1109/JSTSP.2025.3530171

Liu, R., Sisman, B., Gao, G., & Li, H. (2024). Controllable accented text-to-speech synthesis with fine and coarse-grained intensity rendering. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *32*, 2188–2201. https://doi.org/10.1109/TASLP.2024.3378110

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., & Sonderegger, M. (2017). Montreal forced aligner: Trainable text-speech alignment using kaldi. *Interspeech 2017*, 498–502. https://doi.org/10.21437/Interspeech.2017-1386

Meng, L., Zhou, L., Liu, S., Chen, S., Han, B., Hu, S., Liu, Y., Li, J., Zhao, S., Wu, X., Meng, H. M., & Wei, F. (2025). Autoregressive speech synthesis without vector quantization. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1287–1300). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.acl-long.65

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning based text classification: A comprehensive review. https://arxiv.org/abs/2004.03705

Mullennix, J. W., Stern, S. E., Wilson, S. J., & Dyson, C.-l. (2003). Social perception of male and female computer synthesized speech. *Computers in Human Behavior*, *19*(4), 407–424. https://doi.org/10.1016/s0747-5632(02)00081-x

Nallabala, N. K., Souprayen, B., Ramasamy, M., Penumarti, S. K., & Navuluri, N. C. (2025). "an efficient speech synthesizer: A hybrid monotonic architecture for text-to-speech via VAE & LPC-Net with independent sentence length". *Circuits, Systems, and Signal Processing*, *44*(8), 5827–5851.

Nikitaras, K., Vamvoukakis, G., Ellinas, N., Klapsas, K., Markopoulos, K., Raptis, S., Sung, J. S., Jho, G., Chalamandaris, A., & Tsiakoulis, P. (2022). Fine-grained noise control for multispeaker speech synthesis. *Interspeech 2022*, 828–832. https://doi.org/10.21437/interspeech.2022-10765

Niu, R., Wu, W., Chen, J., Ma, L., & Wu, Z. (2025). A multi-stage framework for multimodal controllable speech synthesis. https://arxiv.org/abs/2506.20945

Podsiadło, M., & Chahar, S. (2016). Text-to-speech for individuals with vision loss- a user study. *Interspeech 2016*.

Quené, H., Boomsma, G., & van Erning, R. (2021). Attractiveness of male speakers: Effects of pitch and tempo. In B. Weiss, J. Trouvain, M. Barkat-Defradas, & J. J. Ohala (Eds.), *Voice attractiveness: Studies on sexy, likable, and charismatic speakers* (pp. 153–164). Springer Singapore. https://doi.org/10.1007/978-981-15-6627-1_9

Ramírez-Esparza, N., García-Sierra, A., & Kuhl, P. K. (2014). Look who's talking: Speech style and social context in language input to infants are linked to concurrent and future speech development. *Developmental Science*, *17*(6), 880–891. https://doi.org/https://doi.org/10.1111/desc.12172

Ren, Y., Hu, C., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T.-Y. (2022). Fastspeech 2: Fast and high-quality end-to-end text to speech.

Schreibelmayr, S., & Mara, M. (2022). Robot voices in daily life: Vocal human-likeness and application context as determinants of user acceptance. *Frontiers in Psychology*, *Volume 13 - 2022*. https://doi.org/10.3389/fpsyg.2022.787499

Siani, A., Mcarthur, M., Hicks, B., & Dacin, C. (2022). Gender balance and impact of role models in secondary science education. *New Directions in the Teaching of Physical Sciences*, *17*. https://doi.org/10.29311/ndtps.v0i17.3939

Skerry-Ryan, R., Battenberg, E., Xiao, Y., Wang, Y., Stanton, D., Shor, J., Weiss, R., Clark, R., & Saurous, R. A. (2018). Towards end-to-end prosody transfer for expressive speech synthesis with tacotron. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th international conference on machine learning* (pp. 4693–4702). PMLR. https://proceedings.mlr.press/v80/skerry-ryan18a.html

Sobel, I. (2014). An isotropic 3x3 image gradient operator. *Presentation at Stanford A.I. Project 1968*.

Stoet, G. (2010). PsyToolkit: A software package for programming psychological experiments using linux. *Behav. Res. Methods*, *42*(4), 1096–1104.

Taha, K., Yoo, P. D., Yeun, C., Homouz, D., & Taha, A. (2024). A comprehensive survey of text classification techniques and their research applications: Observational and experimental insights. *Computer Science Review*, *54*, 100664. https://doi.org/https://doi.org/10.1016/j.cosrev.2024.100664

Taylor, P. (2009). *Text-to-speech synthesis*. Cambridge University Press.

Theis, L., van den Oord, A., & Bethge, M. (2016). A note on the evaluation of generative models. https://arxiv.org/abs/1511.01844

Tiwari, S. (2020). A Blur Classification Approach Using Deep Convolution Neural Network. *International Journal of Information System Modeling and Design (IJISMD)*, *11*(1), 93–111. https://ideas.repec.org/a/igg/jismd0/v11y2020i1p93-111.html

Triantafyllopoulos, A., & Schuller, B. W. (2025). Expressivity and speech synthesis. https://arxiv.org/abs/2404.19363

Van Thao, N. (2022). Investigating the realization of speech function in a speech through systemic functional linguistics perspective. *Script Journal: Journal of Linguistics and English Teaching*, *7*(1), 31–41. http://dx.doi.org/10.24903/sj.v7i1.917

Wagner, P., Beskow, J., Betz, S., Edlund, J., Gustafson, J., Eje Henter, G., Le Maguer, S., Malisz, Z., Székely, É., Tånnander, C., & Voße, J. (2019). Speech Synthesis Evaluation — State-of-the-Art Assessment and Suggestion for a Novel Research Program. *Proc. 10th ISCA Workshop on Speech Synthesis (SSW 10)*, 105–110. https://doi.org/10.21437/SSW.2019-19

Wang, H., Liu, S., Meng, L., Li, J., Yang, Y., Zhao, S., Sun, H., Liu, Y., Sun, H., Zhou, J., Lu, Y., & Qin, Y. (2025). Felle: Autoregressive speech synthesis with token-wise coarse-to-fine flow matching. *Proceedings of the 33rd ACM International Conference on Multimedia*, 10229–10238. https://doi.org/10.1145/3746027.3755494

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., & et al. (2017). Tacotron: Towards end-to-end speech synthesis. *Interspeech 2017*. https://doi.org/10.21437/interspeech.2017-1452

Wang, Y., Stanton, D., Zhang, Y., Skerry-Ryan, R. J., Battenberg, E., Shor, J., Xiao, Y., Ren, F., Jia, Y., & Saurous, R. A. (2018a). Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *CoRR*, *abs/1803.09017*. http://arxiv.org/abs/1803.09017

Wang, Y., Stanton, D., Zhang, Y., Skerry-Ryan, R., Battenberg, E., Shor, J., Xiao, Y., Ren, F., Jia, Y., & Saurous, R. A. (2018b). Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. https://arxiv.org/abs/1803.09017

Weeks, T. E. (1971). Speech registers in young children. *Child Development*, *42*(4), 1119–1131. Retrieved November 17, 2025, from http://www.jstor.org/stable/1127797

Wolfe, C. (1997). Historicising the 'voice of god': The place of vocal narration in classical documentary. *Film History*, *9*(2), 149–167. Retrieved November 19, 2025, from http://www.jstor.org/stable/3815172

Ye, Z., Zhu, X., Chan, C.-M., Wang, X., Tan, X., Lei, J., Peng, Y., Liu, H., Jin, Y., Dai, Z., Lin, H., Chen, J., Du, X., Xue, L., Chen, Y., Li, Z., Xie, L., Kong, Q., Guo, Y., & Xue, W. (2025). Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. https://arxiv.org/abs/2502.04128

Zabetie Jahromi, A., & Qaneifard, M. s. (2018). A survey on the structure of expository documentary. *Journal of Interdisciplinary Studies in Communication and Media*, *1*(2), 65–88. https://doi.org/10.22034/jiscm.2018.81871

Zhang, Y., Wang, M., Li, Q., Tiwari, P., & Qin, J. (2025). Pushing the limit of llm capacity for text classification. *Companion Proceedings of the ACM on Web Conference 2025*, 1524–1528. https://doi.org/10.1145/3701716.3715528

Zhou, K., Sisman, B., Rana, R., Schuller, B. W., & Li, H. (2022). Speech synthesis with mixed emotions.

Zhou, K., Sisman, B., Rana, R., Schuller, B. W., & Li, H. (2023). Speech synthesis with mixed emotions. *IEEE Transactions on Affective Computing*, *14*(4), 3120–3134. https://doi.org/10.1109/TAFFC.2022.3233324

Zhou, X., Zhang, M., Zhou, Y., Wu, Z., & Li, H. (2024). Accented text-to-speech synthesis with limited data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *32*, 1699–1711. https://doi.org/10.1109/TASLP.2024.3363414

Zhu, X., & Xue, L. (2020). Building a controllable expressive speech synthesis system with multiple emotion strengths. *Cognitive Systems Research*, *59*, 151–159. https://doi.org/https://doi.org/10.1016/j.cogsys.2019.09.009

Appendices

A Notation

Symbol	Description
Any subscript i can refer to either the ith element of a group, or the subscript can denote the class of an element. For example, T_{sr} refers to a group of text which belong to a certain SR class.	
A, a_i	The audio group, and a member of the audio group. Can either refer to training data for the synthesiser, or the final audio output of a synthesiser.
T, t_i	The text group, and a member of the text group.
$t_{i(tc)}$, $t_{i(ss)}$	A member of the text group which has been preprocessed for Text Classification, and a member of the text group pre-processed for Speech Synthesis.
SF, sf_i	The Speech Function group, and a member of the Speech Function Group.
SR, sr_i	The Speech Register group, and a member of the Speech Register Group.
$x_\varnothing$	Neutral class of a group, mostly SF and SR (can also be referred to as "None" or "Other"). These classes can be used when one may expect text input that cannot, or isn't desired to, be categorised within the problem space.
$n(X)$	Total number of elements in group X.

B Web Scraping

Web Scraping was performed using the Python programming language as a means of both obtaining the relevant training text data of each style, and the corresponding training audio data (T and A). The following libraries were used:

- httplib2, version 0.22.0 (according to the `.__version__` command) was used to make http requests. This library, when using the `http.request()` command, returns the status and response of the page.
- Beautiful Soup, version 4.11.2 (according to the `.__version__` command) was used to parse the response of the httplib2 request. Using the `BeautifulSoup(response, 'html.parser')` command, we now have a page with which we can select relevant items from the page. BeautifulSoup can also be used to do this; with commands such as `.find_all(class_="")` finding specific html classes (such as textboxes), or `.has_attr('href')` to find whether a class contains a URL.
- Markdown, version 3.4.4 (according to the `.__version__` command) was used for web pages which were formatted in Markdown (a markup language to add formatting to plain text documents) in order to retrieve the text from a piece of text. If we have found a piece of text in markdown from BeautifulSoup (which we call md), the code to parse it is as follows:

```
def md_to_text(md):
    html = markdown.markdown(md)
    soup = BeautifulSoup(html, features='html.parser')
    return soup.get_text()
```

- re, version 2.2.1 (according to the `.__version__` command) which is the native Python library for handling Regular Expressions (regex). This is a tool which searches for particular patterns of strings. The re library was used primarly for extracting relevant text found from the processing in BeautifulSoup, but was also used as a substitute for BeautifulSoup in certain cases. For example, the `re.findall(r"(?<=href\=\").*(?=\"\>.+\b)","Text")` command was also used to find URLs from the CNN website (where the body of the website is denoted as "Text". Since each of the websites are uniquely designed, they each require specific regex functions. As such, devising the regex functions for each website is a heuristic process.

- pandas, version 1.5.3 (according to the `.__version__` command) was used to convert the lists of found results into dataframes, which would then be exported as csv (comma separated values) files. If we have a list of texts (let's call it "LIST", we can convert them to a pandas dataframe as follows:

```
LIST=list(dict.fromkeys(LIST))
Genre_Tag=["Genre"]*len(LIST)
df=pd.DataFrame(list(zip(Genre_Tag, LIST)),columns =['Genre', 'Text'])
df.to_csv("MyCSV.csv")
```

  The first line turns the list into a dictionary and retrieves the keys only. This is done to prevent duplicate entries, as a dictionary necessarily cannot have more than one key. The second line generates another list of the name of the genre mulitplied by the length of the initial LIST. This is done to label each element. The third line merges both of these lists (Genre_Tag and LIST) into a Pandas dataframe, giving them the names "Genre" and "Text" respectively. The final line exports the dataframe into a CSV, which can then be used as input into machine learning architectures. Below is an example of a generated CSV using this method.

Genre	Text
News Report	TOM FOREMAN, CNN CORRESPONDENT: Why that man in the
News Report	ERIN BURNETT, CNN HOST: The Oath Keepers were a key pres
News Report	I'm so glad her life made this turn for success and the positive sid
News Report	LIEBERMANN: So the question now, how long does this assessm
News Report	ANDERSON COOPER, CNN HOST: 2024 Republican race for pr
News Report	ERIN BURNETT, CNN HOST: But the controversy seemed to fina
News Report	ERIN BURNETT, CNN HOST: They were based on but fundamen
News Report	ANDERSON COOPER, CNN HOST, "ANDERSON COOPER: 36
News Report	COOPER: It's rare to get some good news related to Alzheimer's,
News Report	A. COOPER, CNN HOST: Breaking news now, even as we are le
News Report	When authorities conducted a welfare check on J. J. in Novembe
News Report	ANDERSON COOPER, CNN HOST: Good evening. Before we be

Figure 40: A CSV image generated using the Pandas package. Here, the initial LIST contains the web scraped output of different versions of the Anderson Cooper 360° show. The Genre Tag is "News Report".

C Demonstrator and Data

The folder containing all relevant data is in the corersponding GitHub repository:

`https://github.com/585hubert/Genre_Synthesis`

The contents (as of 19/11/2025) are as follows:

Walkthrough

The process of synthesising Speech Genres is as follows:

1. First, a piece of text needs to be converted into neutral synthesised speech (speech which is devoid of any Speech Register) using a speech synthesiser. In this research paper, FastSpeech2 is used for this task. Inside the FastSpeech2 folder is a notebook file which contains the `!git clone command` for the specific implementation of FastSpeech2 used. There is both a README file inside of this folder, and a README file from the cloned folder. Follow the instructions of those README files (download the pretrained model). To synthesise desired text, the input was changed within the `eval.py` file, specifically within the `get_data()` function. The folder contains both text with the Speech Function of each genre (Reference Audio Text) and function neutral text (Harvard Sentences), both of which were used in the research paper.

2. Secondly, a text classifier is required to identify the Speech Function of a given text. This task is performed by a Recurrent Convolutional Neural Network (RCNN) in this research paper. Inside of the RCNN folder, there is a notebook file which contains the `!git clone` command to the specific RCNN implementation used. Follow the instructions of the README file inside of that folder. The README file contains a link to a zipped folder which contains the necessary csv files to train the RCNN models used in the research paper, alongside the `classes.txt` necessary for classification. You unzip this folder and use the contents to replace the files from the default git clone. The cloned repository has the tools to both train and evaluate the RCNN models.

3. Finally, the k-Nearest Neighbours Voice Conversion (kNN-VC) architecture is used to apply the Speech Register to the output of the speech synthesiser from step 1. Inside of the kNN-VC folder is a notebook file which contains the `torch.hub.load()` call of the specific implementation of the kNN-VC architecture used in the paper, alongside the full implementation of creating the Speech Register embeddings, and applying them to speech samples. The folder containing all of the audio files is too large to be included. Thus, the README file contains a link to a zipped folder which contains the PyTorch tensors for each of the genres, for each of the genders.

With these steps, you should be able to synthesise Speech Genres. The research paper also looked at alternative cases where the Speech Function did not match the Speech Register (e.g. a News Broadcast text being synthesised with a Stand-Up Comedy register) or the synthesis of function neutral text with various registers.

Additional Material

- *Embeddings - Subset* - A folder which contains a small subset of the training data used for the kNN-VC to generate the genre embeddings.

- *PsyToolKit Script* - A folder which contains the survey file used for the evaluation of synthesised speech samples. The Samples folder contains the necessary samples required to run the survey.

- *Result_Analysis* - A folder which contains the notebook files which analysed both the RCNN evaluation, and the human evaluation of synthesised speech samples. The accompanying data.csv file contains the responses of all of the participants, each indexed with a unique but

nontraceable key.

- *Web Scraping Tools* - A folder which contains the notebook files which were used to obtain the text necessary to train the RCNN, one for each genre. Additionally, the Text Splitter notebook is used to split the text for each genre into segments of texts of desired character length.
- *Additional Figures* - Figures of results which couldn't make it into the main document. Most of them are genre specific breakdowns of each question.

D Evaluation Text

```
Hello There.

This evaluation is part of a research project looking into synthesising speech
styles. In short, the project is trying to teach A.I. to recognise different
types of speech (such as a News Broadcast, Sports Commentary, Poetic Reading, etc)
and then speak those different types of speech in the right way (rather
than with a flat delivery). It should imitate a stand up comic when presented
with a stand up text, a documentary narrator when presented with a documentary
script, and so on (more information in the dropdown).

If you're using a mobile device, I recommed switching to "desktop site"
if possible.

You will be asked  to listen to various audio samples. In the first part of
the survey, you give a rating of how much you like each of the voices. In
the second part, you will be asked to try to guess which A.I. voice is
speaking a given extract.

Before the survey begins, there will be a short onboarding question
regarding how much time you spend listening to specific types of
speech.

The survey should take no more than 30 minutes to complete in total.
You have the right to refuse the survey, to stop whenever you'd
like, or to request your responses to be deleted (see contact information).

More information is provided in the dropdown.

Thank you in advance!
```

Additional information:

```
Speech styles are synthesised by means of first converting a chosen piece
of text into a synthesised speech sample, using pre-existing speech synthesis
software. The additional step done here is to then take the input text and
classify which speech style it belongs to. From this, a dataset corresponding
to the identified genre is taken, and a secondary architecture then takes the
generated speech sample and modifies it to sound like the desired speech style.

For example, if we have an input text of "Earlier today, the president
```

addressed the nation regarding recent terrorist attacks within the nation's train stations", the architecture will identify this as a news extract, and then access the dataset containing news anchor speech. After the initial sample, this dataset will be used to create a representation of news speech, and apply it to the generated sample.

Audio is naturally required for this evaluation. A good pair of headphones would be advisable, but not required. If you can hear the audio samples well, you're fine. If you're having issues with audio, either check your audio settings, your audio device (the headphones/speakers you're using) or try to run the evaluation with a different browser.

Your country information and I.P. address will not be stored as they are not needed. No personal data (e,g. your name, age, gender, etc) is required either.

You can stop the survey at any time you like. Partially filled in responses will still be recorded, so if you wish to have it deleted, please contact me.

The provided contact information is my student email address: h.matuszewski@student.rug.nl