DEGREE PROJECT IN COMPUTER SCIENCE AND ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2021

A Comparative Study of the Quality between Formality Style Transfer of Sentences in Swedish and English, leveraging the BERT model

MARIA LINDBLAD

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

A Comparative Study of the Quality between Formality Style Transfer of Sentences in Swedish and English, leveraging the BERT model

MARIA LINDBLAD

Master's Programme, Computer Science, 120 credits
Date: May 27, 2021

Supervisors: Yulia Sidorova, Henrik Svensson
Examiner: Viggo Kann
School of Electrical Engineering and Computer Science
Host company: Hejare AB
Swedish title: En jämförande studie av kvaliteten mellan överföring av formalitetsstil på svenska och engelska meningar, med hjälp av BERT-modellen

Abstract

Formality Style Transfer (FST) is the task of automatically transforming a piece of text from one level of formality to another. Previous research has investigated different methods of performing FST on text in English, but at the time of this project there were to the author's knowledge no previous studies analysing the quality of FST on text in Swedish.

The purpose of this thesis was to investigate how a model trained for FST in Swedish performs. This was done by comparing the quality of a model trained on text in Swedish for FST, to an equivalent model trained on text in English for FST. Both models were implemented as encoder-decoder architectures, warm-started using two pre-existing Bidirectional Encoder Representations from Transformers (BERT) models, pre-trained on Swedish and English text respectively. The two FST models were fine-tuned for both the *informal to formal* task as well as the *formal to informal* task, using the Grammarly's Yahoo Answers Formality Corpus (GYAFC).

The Swedish version of GYAFC was created through automatic machine translation of the original English version. The Swedish corpus was then evaluated on the three criteria *meaning preservation*, *formality preservation* and *fluency preservation*.

The results of the study indicated that the Swedish model had the capacity to match the quality of the English model but was held back by the inferior quality of the Swedish corpus. The study also highlighted the need for task specific corpus in Swedish.

Keywords

Formality Style Transfer, BERT, Natural Language Generation, Swedish Language Models, GYAFC, Encoder-Decoder Models

Sammanfattning

Överföring av formalitetsstil syftar på uppgiften att automatiskt omvandla ett stycke text från en nivå av formalitet till en annan. Tidigare forskning har undersökt olika metoder för att utföra uppgiften på engelsk text men vid tiden för detta projekt fanns det enligt författarens vetskap inga tidigare studier som analyserat kvaliteten för överföring av formalitetsstil på svensk text.

Syftet med detta arbete var att undersöka hur en modell tränad för överföring av formalitetsstil på svensk text presterar. Detta gjordes genom att jämföra kvaliteten på en modell tränad för överföring av formalitetsstil på svensk text, med en motsvarande modell tränad på engelsk text. Båda modellerna implementerades som kodnings-avkodningsmodeller, vars vikter initierats med hjälp av två befintliga Bidirectional Encoder Representations from Transformers (BERT)-modeller, förtränade på svensk respektive engelsk text. De två modellerna finjusterades för omvandling både från *informell* stil till *formell* och från *formell* stil till *informell*. Under finjusteringen användes en svensk och en engelsk version av korpusen Grammarly's Yahoo Answers Formality Corpus (GYAFC).

Den svenska versionen av GYAFC skapades genom automatisk maskinöversättning av den ursprungliga engelska versionen. Den svenska korpusen utvärderades sedan med hjälp av de tre kriterierna *betydelse-bevarande*, *formalitets-bevarande* och *flödes-bevarande*.

Resultaten från studien indikerade att den svenska modellen hade kapaciteten att matcha kvaliteten på den engelska modellen men hölls tillbaka av den svenska korpusens sämre kvalitet. Studien underströk också behovet av uppgiftsspecifika korpusar på svenska.

Nyckelord

Överföring av formalitetsstil, BERT, Generering av naturligt språk, Svenska språkmodeller, GYAFC, Kodnings-avkodningsmodeller

Acknowledgments

I would like to express my deep gratitude and appreciation to my supervisors Yulia Sidorova and Henrik Svensson, for all discussions, feedback and insight. I would like to thank Hejare AB for allowing me to write my thesis at their company and supporting me throughout the process. I would also like to thank my examiner Viggo Kann for his valuable feedback, which improved the final version of the report. Moreover, I would like to thank Yahoo Research for making their data available, and Sudha Rao and Joel Tetreault for sharing their corpus. Finally, I would like to thank Johan Boye for his valuable suggestions during the specification of this thesis project.

Stockholm, May 2021
Maria Lindblad

Contents

List of Figures

List of Tables

List of acronyms and abbreviations

BERT Bidirectional Encoder Representations from Transformers

BLEU Bilingual Evaluation Understudy

E&M Entertainment & Music

F&R Family & Relationships

FST Formality Style Transfer

GPT Generative Pre-trained Transformer

GYAFC Grammarly's Yahoo Answers Formality Corpus

KB-BERT Swedish BERT

LSTM Long Short Term Memory

M-BERT Multilingual BERT

NLG Natural Language Generation

NLP Natural Language Processing

NMT Neural Machine Translation

PBMT Phrase-Based Machine Translation

PINC Paraphrase In N-gram Changes

RNN Recurrent Neural Network

Chapter 1

Introduction

Formality Style Transfer (FST) is a task within the field of Natural Language Generation (NLG) where the goal is to automatically transform a piece of text from one level of formality to another. For example an informal sentence can be transformed into the same sentence but written in a more formal style.

Research in FST has many applications and can lead to several commercial and social benefits. One application is writing assistance tools. For example, such a tool could be used to help people formulate more formally written essays or resumes and by extension help them finish school or get employment. The tool could also be used to transform complicated text into more informal and comprehensible text. This could help people comprehend written material that they have previously been unable to absorb due to its complex writing style.

The research field of FST has developed a lot in recent years and many recent studies utilise transformer based models for the task [6][7]. Despite of the rapid development in the field of FST, no FST models or techniques have been tested for the Swedish language. This is partly because Swedish is a language used by relatively few people but also because there are no parallel informal / formal datasets in Swedish and until recently there were no transformer based models pre-trained on the Swedish language. The lack of research in the field of Swedish FST may lead to an insecurity regarding whether or not the techniques are applicable to Swedish material and how the models perform when applied to text in Swedish.

In 2020, the National Library of Sweden released a version of the transformer based BERT model pre-trained on Swedish text, called Swedish BERT (KB-BERT) [8]. The KB-BERT model was proven to match the performance of the corresponding original English BERT model and also outperformed a

multilingual BERT model. The KB-BERT model hence constitutes a good basis for future studies in Natural Language Processing (NLP) focused on the Swedish language and therefore also future studies in the field of Swedish FST.

The company Hejare AB [9] is a Swedish consulting firm, that are in the process of developing a web platform called Momang [10]. Momang is intended for storing, updating and distributing consultant resumes in an efficient and easy way. The platform already contains writing assistance tools such as spelling correction and translation. Since the customers of consultant firms react differently to different levels of formality in a written resume and Hejare AB want to spare their users the tedious task of rewriting resumes, a welcome addition to the platform would be a FST tool applicable to both the English and Swedish language. Since no such tools for the Swedish language exist it is in the interest of Hejare AB to create their own FST tool.

The intention of this thesis project is to investigate how a FST model specifically trained for text in Swedish, compare to an equivalent model trained for text in English. The study serve as a first step in the development of the new tool for Momang and hopefully the results can be used as guidelines later on when the tool is created.

The project is also interesting from a purely academic point of view, since there is to the author's knowledge no previous research published in the area of FST trained on text in Swedish. There are multiple papers on FST for the English language which makes a comparison of the two languages interesting. This thesis project could serve as a benchmark for future studies in the field of Swedish FST.

1.1 Problem

1.1.1 Original Problem and Definition

This thesis originates from the lack of knowledge in the field of Swedish FST and the lack of models trained for the task. This shortage of knowledge in the academic community also affect the commercial market, since FST can be a useful addition to writing assistance tools [4] and it is hard to develop such tools without any research to base the development on.

The lack of a benchmark dataset in Swedish is also something that slows down progress in the field. Automatically translating an existing corpus from another language is an option, but there is no metric to evaluate the translations with, that also take formality into account.

1.1.2 Research Questions

- How does the quality of an encoder-decoder model, warm-started using KB-BERT and fine-tuned on sentences in Swedish for the task of FST, compare to the quality of an encoder-decoder model, warm-started using BERT_{BASE} and fine-tuned on sentences in English for the task of FST?
- How can the quality of sentences automatically translated from English to Swedish be evaluated with regard to formality?

1.2 Purpose

There are two primary purposes of this thesis. The first is to investigate how well a FST model specifically trained for text in Swedish performs. The baseline to which this Swedish FST model will be tested against, is a corresponding FST model trained for text in English. Assuming the Swedish model proves reliable, this work could be used as a benchmark for future research in the field and also work as a guide for the commercial sector when developing new tools for FST in Swedish. Assuming the Swedish model however proves unsuccessful, the identified limitations of the model could be helpful in future research attempting to improve FST models for Swedish text.

The second purpose is to define a manual evaluation method, specifically designed for evaluation of sentences translated from English to Swedish, with special regard to the *formality preservation* of the sentences. Assuming the method proves reliable, this evaluation could potentially enable new corpus to be created through machine translation. In turn, that could speed up the progress in the field of Swedish FST.

1.3 Goals

The goal of this project is to compare two models trained for FST in English and Swedish respectively. This goal is divided into the following sub-goals:

- Define an method for evaluating sentences translated from English to Swedish, with special regard to the *formality preservation* of the sentences.
- Translate the GYAFC corpus to Swedish.
- Evaluate the translation of the GYAFC corpus.

- Construct a model for Swedish FST that leverages KB-BERT.
- Construct a model for English FST that leverages BERT_{BASE}.
- Train two versions of the Swedish FST model, one converting *formal* sentences to *informal* sentence and the other converting *informal* sentences to *formal* sentences. Train using the Swedish GYAFC corpus.
- Train two versions of the English FST model, one converting *formal* sentences to *informal* sentence and the other converting *informal* sentences to *formal* sentences. Train using the English GYAFC corpus.
- Evaluate the FST models using the BLEU and PINC scores.
- Compare the scores of the English and Swedish FST models.

1.4 Delimitations

Due to time constraints, it is not within the scope of this project to manually correct the translated GYAFC sentences. Instead the quality of the sentences will be assessed. This assessment will be performed manually since no appropriate automatic tool has been found. Time constraints also prevent from assessing more than a subset of sentences.

The two BERT models in this study will not be implemented from scratch or pre-trained in this project. Instead two pre-existing BERT models will be utilised and these models will not be fine-tuned or evaluated on any other task than FST. Nor is it within the scope of the project to investigate the leveraging of any other model for FST, even though a selection of such models are presented in Chapter 2.

Ideally, the models used for FST should be evaluated using as many evaluation metrics as possible. However, due to practical constraints only BLEU and PINC will be used.

Finally, this thesis will not go into depth of the linguistic differences between the two languages Swedish and English. Analyzing these differences from a formality perspective requires advanced linguistic knowledge and in order to gain any valid insight from such an analysis a vast study would have to be conducted within the subject.

1.5 Ethics and Sustainability

There is a risk that the models in this thesis will generate sentences that are biased or incorrect. It is therefore very important that the models are trained with data that does not contain for example prejudice. It is also key that the models are evaluated from a quality point of view, whilst also paying attention to the ethical perspective.

Training NLG models often require a great amount of energy resources. Therefore it is vital not to train unnecessarily, wasting these resources. A good approach is to utilise a model which has already been pre-trained, since pre-training is performed only once and merely a fraction of the resources is required for fine-tuning. From a sustainability perspective, it is therefore advantageous to leverage pre-trained BERT models to warm-start encoder-decoder models, as will be done in this thesis.

The GYAFC dataset is only intended for research. As this was requested by the creators of the corpus, it is very important not to break the trust they gave when deciding to share the corpus. The corpus and trained models will therefore not be shared with anyone or published anywhere without the approval of the creators of GYAFC.

1.6 Structure of the Thesis

Chapter 2 presents relevant background and related work for this thesis. In Chapter 3 the method is described and Chapter 4 covers the practical implementation of the method. The results of the thesis are displayed in Chapter 5, and the method and results are further discusses in Chapter 6. In Chapter 7, conclusions are drawn and directions for future research are proposed.

Chapter 2

Background

The following chapter presents the result of the pre-study which was conducted in the beginning of the project. The chapter introduces the relevant background knowledge which is needed in order to understand the thesis and the choice of method.

First, an introduction to FST is provided together with some related work in the field. Secondly, relevant facts of the GYAFC dataset is presented and then a few evaluation techniques for style transfer are introduced. This is followed by an introduction to word embeddings and then a section regarding deep learning techniques for NLG. Then a section on transfer learning is provided and the chapter ends which a section describing the BERT model and how to leverage the model for NLG.

2.1 Formality Style Transfer (FST)

Formality Style Transfer (FST) is a sequence-to-sequence task where the goal is to automatically convert a piece of text from one level of formality to another. An informal sentence can for example be transformed into a formal version of the same sentence. There are also other types of style transfer, such as sentiment style transfer and gender style transfer, but according to Heylighen and Dewaele, formality is an especially important form of style [11].

Techniques for FST can for example be a useful addition to writing assistance tools [4]. This is partly because the style of a text can communicate information that the literal meaning of the words does not [12] and therefore lead to the text being received in different ways. For instance, a formally written email used in a job application may result in a job offer and an informal version of a new law might be easier for the public to understand.

2.1.1 Identifying Formality

Historically, one way of identifying formality in text has been by defining a set of rules or characteristics that applies to the style [13][3]. A summary of some of the main characteristics used in previous studies was composed in 2011 by Sheikha and Inkpen [3]. A copy of their summary is displayed in Table 2.1.

Main Characteristics of Informal Style Text	Main Characteristics of Formal Style Text
• It uses personal pronouns and the active voice.	• It uses impersonal pronouns and often the passive voice.
• It uses short simple words and sentences.	• It uses complex words and sentence.
• It uses Contractions (e.g., “won’t”).	• It does not use contractions.
• It uses many abbreviations (e.g., “TV”).	• It does not use many abbreviations.
• It uses many phrasal verbs.	• It uses appropriate and clear expressions, business, and technical vocabulary.
• The words that express rapport and familiarity are often used in speech, such as “brother”, “buddy”, and “man”.	• It uses politeness words and formulas such as “Please”, “Sir”.
• It uses a subjective style, expressing opinions and feelings.	• It uses an objective style, using facts and references to support an argument.
• It uses vague expressions and colloquial (slang words are accepted in spoken not in written text (e.g., “wanna” = “want to”)).	• It does not use vague expressions and slang words.

Table 2.1: The main characteristics of formal and informal text, cited from Sheikha and Inkpen [3].

Formality has also been automatically identified, using for example a classifier trained to detect formality in text. In 2016, Pavlick and Tetreault introduced a statistical sentence level formality classifier, which approximated the human perception of formality with high accuracy [13]. Their classifier has also later been used as an automatic evaluation tool for FST [4].

2.1.2 Related Work

Sheikha and Inkpen, 2011

In 2011 Sheikha and Inkpen generated formal and informal sentences using template-based NLG. They collected parallel lists of formal and informal words and phrases which were used to constructed templates. The temples were then used to replace words in sentences based on user preference. Their system could successfully generate both formal and informal sentences with a

high accuracy but Sheikha and Inkpen judged that their biggest contribution was the set of parameters used to construct formal and informal sentences [3].

Xu et al., 2012

One of the first studies that treated style transfer as a sequence-to-sequence task was conducted by Xu et al. in 2012. They used William Shakespeare's original plays and modern translations of the same plays, to generate a parallel corpus of 30,000 sentence pairs. Then they trained several phrase-based machine translation models using the data and evaluated the results using human judgement and standard automatic metrics such as BLEU. Xu et al. also introduced three new metrics based on cosine similarity, language model and logistic regression. These metrics measured how well the output matched the target style. [14]

Hu et al., 2017

One problem in the field of style transfer has been the lack of parallel data. To solve this, recent studies have tried to control specific attributes of the generated text without the use of parallel data.

Hu et al. [15] proposed a neural generative model which controlled the sentiment and the tense of generated text by learning disentangled latent representations. Their model could successfully generate short sentences and improved the accuracy of sentiment and tense attributes compared to previous generative models.

Rao and Tetreault, 2018

In 2018, Rao and Tetreault introduced the Grammarly's Yahoo Answers Formality Corpus (GYAFC) [4]. The corpus was created using the *Yahoo Answers L6 corpus* [16] and consisted of 110,000 informal / formal sentence pairs. The sentences in the corpus belonged to two domains, Entertainment & Music (E&M) and Family & Relationships (F&R). The corpus is introduced further in Section 2.2.

Rao and Tetreault also adapted existing Phrase-Based Machine Translation (PBMT) [14] and Neural Machine Translation (NMT) [17] approaches for the task of FST and evaluated the results using both automatic evaluation metrics, such as classifiers, and human judgement. The human-based evaluation, scored the results using the three criteria *formality*, *fluency* and *meaning preservation*. Rao and Tetreault also utilised the more standard automatic

metrics, BLEU, PINC and TERp. The automatic evaluation metrics had a slight preference towards the NMT models, while the PBMT model preformed better based on the human evaluation.

The BLEU and PINC scores of the PBMT and NMT models are displayed in Tables 2.2 and 2.3. [4]

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Informal	50.69	0.00	52.61	0.00
Formal Reference	100.0	69.79	100.0	67.83
PBMT	67.26	44.94	-	-
PBMT Combined	-	-	**74.32**	44.77
NMT Baseline	56.61	**56.92**	69.09	**51.00**
NMT Copy	58.01	56.39	69.41	50.93
NMT Combined	**67.67**	43.54	**74.60**	41.52

Table 2.2: BLEU and PINC scores of models on 500 test sentences from both the E&M and F&R domains in the GYAFC dataset for the *informal to formal* task. The scores were obtained by Rao and Tetreault [4].

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Formal	29.54	0.00	28.64	0.00
Informal Reference	100.0	81.14	100.0	80.69
PBMT	-	-	-	-
PBMT Combined	32.66	37.38	31.20	26.74
NMT Baseline	29.87	**58.38**	**34.69**	**48.68**
NMT Copy	31.29	55.42	33.76	47.52
NMT Combined	**34.07**	42.31	33.57	36.76

Table 2.3: BLEU and PINC scores of models on 500 test sentences from both the E&M and F&R domains in the GYAFC dataset for the *formal to informal* task. The scores were obtained by Rao and Tetreault [4].

Niu et al., 2018

Niu et al. [5], used multi-task learning to jointly perform monolingual FST and bilingual formality-sensitive machine translation. They used the GYAFC dataset [4] as FST data and showed that their joint model improved the quality of FST compared to the results obtained by Rao and Tetreault [4].

The BLEU score of their best performing model is presented in Table 2.4.

Xu et al., 2019

Xu et al. [6] also utilised the GYAFC dataset [4] and were the first to attempt to use the Transformer model [1] for the FST task. They proposed a Transformer model that incorporated formality classification feedback into the training process to assist the model learning. Using this technique they obtained state-of-the-art results on the FST task.

The BLEU scores of their models are presented in Table 2.4.

Wang et al., 2019

Wang et al. [7] incorporated pre-trained networks (in particular the GPT-2 model [18]) with rules for FST. Among other things, they trained a model (GPT-CAT) using a concatenation of the original source sentence and a pre-processed version of the sentence as input. The pre-processing consisted of normalizing the sentence using a set of rules handling for example capitalisation, repetition and slang. They trained and evaluated their results on the GYAFC dataset [4] and their results showed that their concatenated model (GPT-CAT) outperformed strictly fine-tuned GPT-2 models (GPT-Orig).

The BLEU and PINC scores for GPT-CAT and GPT-Orig are presented in Table 2.4.

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Bi-directional FT + multi-task learning (MultiTask-tag-style) [5]	72.13	-	75.37	-
Ours [6]	69.08	-	72.90	-
Ours w/ class-filter [6]	68.71	-	73.16	-
Ours w/ gec [6]	69.63	-	74.43	-
GPT-Orig [7]	69.30	47.35	75.65	42.20
GPT-CAT [7]	71.39	46.38	76.87	42.44

Table 2.4: BLEU and PINC scores of models on 500 test sentences from both the E&M and F&R domains in the GYAFC dataset for the *informal to formal* task. The models and scores were introduced by Niu et al. [5], Xu et al. [6] and Wang et al. [7].

2.2 GYAFC

The Grammarly's Yahoo Answers Formality Corpus (GYAFC) was introduced in 2018 by Rao and Tetreault [4]. The corpus consisted of 110,000 informal

/ formal sentence pairs and after creation, the corpus was made available for research purposes which resulted in several studies that use the corpus for FST [5][6][7].

The GYAFC dataset was constructed from the *Yahoo Answers L6 corpus* [16]. The *Yahoo Answers L6 corpus* consisted of questions and their corresponding answers, taken from the web site *Yahoo! Answers* [19]. The *Yahoo Answers L6 corpus* contained sentences from several different domains but GYAFC was created from only the Entertainment & Music (E&M) and Family & Relationships (F&R) domains, as these domains contained the most informal sentences [4].

When constructing GYAFC, formal versions of the sentences from the *Yahoo Answers L6 corpus* were manually created and Table 2.5 shows the number of sentences used by Rao and Tetreault for training, tuning and testing in both directions of FST. For the tuning and test sets, all sentences were given four reference sentences against which evaluation could be measured. [4]

The corpus was originally intended for the *informal to formal* FST task but could also be utilised for the *formal to informal* task.

Samples from both domains of the GYAFC dataset, can be found in Table 2.6.

		Informal to Formal		Formal to Informal	
	Train	Tune	Test	Tune	Test
E&M	52,595	2,877	1,416	2,356	1,082
F&R	51,967	2,788	1,332	2,247	1,019

Table 2.5: The number of sentences in the different parts of the GYAFC dataset. [4]

2.3 Evaluating Style Transfer

At the time of this project, there was no standard evaluation metric specifically designed for FST. The metrics BLEU [20] and PINC [21] were two automated evaluation methods that were often used and therefore considered standard. They were however originally created for the task of machine translation and therefore they had a few shortcomings when applied to the field of FST.

Manual evaluation using human judgement was another popular approach which could yield accurate results from a human perspective. The text was then often evaluated using criteria such as *meaning preservation*, *fluency* and *style* [4]. This method did however require a lot of manpower as well as

Entertainment & Music (E&M)	
Original informal	of corse i be wachin it evry day, my fav charachter is Inuasha
Reference formal	I watch it everyday, my favorite charachter is Inuasha.
Original informal	the lion went left because it was a shortcut
Reference formal	The lion turned to the left, for it was a shorter path.
Family & Relationships (F&R)	
Original informal	whatever they accuse you of they are doing it.
Reference formal	They are committing the very actions they accuse you of.
Original informal	Get it out in the open baby!!
Reference formal	Get it out in the open!

Table 2.6: Samples of sentences from the GYAFC training dataset from both the E&M and the F&R domains. [4]

extensive linguistic knowledge. Different panels of people may also assess text in different ways. This made the approach impractical to set as an official standard and therefore manual evaluation was often used as a complement to automated evaluation.

Below follows more thorough explanations of the standard automatic evaluation metrics, BLEU and PINC as well as a section describing how Rao and Tetreault [4] utilised the three criteria *formality*, *fluency* and *meaning preservation* for human-based evaluation.

2.3.1 BLEU

The Bilingual Evaluation Understudy (BLEU) metric was first introduced by Papineni et al. in 2002 [20] as an automatic evaluation metric designed to evaluate machine translations against reference human translations. The metric has been a popular choice for evaluating various NLG tasks, including style transfer [4][7][14].

The metric was designed to measure the semantic similarity between one or several human generated reference sentences and a machine generated candidate sentence. This was done based on the idea that if a candidate sentence was semantically similar to the references sentences, it would also shared more words and phrases with the reference sentences than a candidate sentence with less semantic similarity. Candidate translations were scored with a value between zero and 100, and the similarity was calculated by comparing n-grams* of the candidate and references sentences. The more n-

* Sequences of n tokens.

grams that matched, the higher the score of the candidate translation. The matches were position independent in order to avoid high scores for candidate sentences containing the same word multiple times. [20]

Papineni et al. found the best correlation with human judgements using n-grams with an n value of one to four. First four modified precision scores, p_n, were calculated, one for each value of n, according to Equation 2.1. Papineni et al. iterated through all candidate translations and then all n-grams in the candidate translation, calculating $Count_{Clip}(n\text{-}gram)$ divided by $Count(n\text{-}gram)$. $Count_{Clip}(n\text{-}gram)$ was the intersect between the number of occurrences of a specific n-gram in the candidate translation and the number of occurrences of the same n-gram in the references. $Count(n\text{-}gram)$ was the number of occurrences of a specific n-gram in a specific candidate translation. The quotients calculated with regard to each candidate sentence and each n-gram in the sentences, were then summed together to form a modified precision score.

These four modified precision scores were then multiplied by weights, w_n, calculated according to Equation 2.3 and these products were then summed together as described in Equation 2.4

As this implementation penalized candidate translations longer than their references, Papineni et al. [20] also introduced a brevity penalty for sentences shorter than or equal to their references in length. The brevity penalty was 1.0 when the candidate translation matched at least one of its references in length. Therefore the reference sentence used when calculating the penalty was the one that best matched the candidate translation in length. Papineni et al. called the length of the best matching sentence *best match length*. In order to not penalize the shorter sentences too much, the brevity penalty was calculated using the entire corpus according to Equation 2.2. In Equation 2.2, c was the total length of the candidate corpus and r was the sum of all *best match length*.

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} (Count_{Clip}(n\text{-}gram))}{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} (Count(n\text{-}gram))} \tag{2.1}$$

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{c}{r})} & \text{if } c \leq r \end{cases} \tag{2.2}$$

$$w_n = \frac{1}{N} \tag{2.3}$$

$$BLEU = BP * \exp(\sum_{n=1}^{N} w_n \log p_n) \tag{2.4}$$

BLEU has been proven to correlate well with human judgement in machine translation tasks [20] but one could argue that it lacks the capability to distinguish the real meaning of a sentence from pure likeness in the construction of words. This would result in the metric being inadequate in some tasks where the candidate translation can differ from the reference sentences but still have the same semantic meaning. Adding as many reference sentences as possible would increase the likelihood of a more fair evaluation. However, as long as there are correct sentences that are not represented as a reference, a generated sentence may receive a low score despite of it being adequate.

Another shortcoming of the metric is that it requires parallel data and preferably several reference sentences. A problem within many NLG tasks is the lack of data and specifically parallel data. Constructing parallel datasets for training and testing is a time consuming task that requires a lot of manpower, as well as significant linguistic knowledge. [14]

2.3.2 PINC

The Paraphrase In N-gram Changes (PINC) score was introduced in 2011 by Chen and Dolan [21] and was designed to measure the percentage of n-grams that are present in a candidate translation but not in the source sentence, from which the candidate translation was generated. It was meant as a complement to the BLEU score and can be seen as the inverse of BLEU. While BLEU rewards sentences similar to its references, PINC rewards candidate sentences that has changed from their original source.

In likeness with BLEU, PINC utilised n-grams and performed an intersect between the n-grams found in the candidate translations and the ones found in the source sentence. PINC was then calculated according to Equation 2.5 where s was the source sentence and c was the candidate translation.

$$PINC(s, c) = \frac{1}{N} \sum_{n=1}^{N} 1 - \frac{|n\text{-}gram_s \cap n\text{-}gram_c|}{|n\text{-}gram_c|} \tag{2.5}$$

PINC has also been shown to correlate well with human judgement and work well in combination with the BLEU score [21]. In contrast to BLEU, PINC does not require parallel data as the candidate translation is compared

to the source sentence and not a reference. For the task of style transfer, it does however lack the capability to distinguish real style change from just a change of word composition.

2.3.3 Human-Based Evaluation

As mentioned earlier, Rao and Tetreault [4] used the three criteria *formality*, *fluency* and *meaning preservation* to evaluate their FST results. This was done manually on 500 sentences from their test set using five judges on each sentence.

They rated the *formality* of a sentence on a scale from -3 to 3, the *fluency* was rated on a scale from 1 to 5 and *meaning preservation* was rated using the source sentence together with the candidate sentence on a scale from 1 to 6. All scales are presented in Table 2.7.

Score	Formality Scale	Fluency Scale	Meaning Preservation Scale
-3	Very Informal	-	-
-2	Informal	-	-
-1	Somewhat Informal	-	-
0	Neutral	-	-
1	Somewhat Formal	Other (incomplete sentences)	Completely dissimilar
2	Formal	Incomprehensible	Not equivalent but on same topic
3	Very Formal	Somewhat Comprehensible	Not equivalent but share some details
4	-	Comprehensible	Roughly equivalent
5	-	Perfect	Mostly equivalent
6	-	-	Completely equivalent

Table 2.7: The scales on which human-based evaluation was performed by Rao and Tetreault. [4]

Rao and Tetreault [4] also calculated the overall ranking of a model. Given the original source sentence, the target reference sentences and the candidate sentence, 500 test samples were ranked by five judges. The judges were asked to rank the candidate sentences according to their overall *formality*, taking into account *fluency* and *meaning preservation*. The overall ranking of a model was then calculated according to Equation 2.6 where $model$ represents the model that was to be scored, S was the subset of 500 source sentences, J was the five judgements, s_{model} was the candidate sentence generated by the model from the source sentence s and $rank(s_{model}, j)$ was the score of s_{model} in judgment j. Unfortunately it was not clear from the paper on what scale the judgements

were made. [4]

$$rank(model) = \frac{1}{|S|} \sum_{s \in S} \frac{1}{|J|} \sum_{j \in J} rank(s_{model}, j) \tag{2.6}$$

2.4 Word Embeddnings

Word embeddings are a very popular way of representing the words in a text so that they are readable for a computer. They consist of numerical representations of words that for example can capture the meaning of the words in terms of semantics and context.

2.4.1 Word2Vec

The Word2Vec embedding was introduced in 2013 [22] and aimed to give words with similar meaning, similar embeddings. When constructing the embeddings, a "window" of a chosen size was slid across a given one-hot encoded text, focusing the attention on a subset of the text. A subset consisted of a target word and its context. The methods then used a two layer neural network and consisted of two different algorithms, continuous bag-of-words and skip-gram. Using continuous bag-of-words, the target word was predicted from the context words and using skip-gram, the context words were predicted from the target word. In both cases the neural network was trained to predict the target word or the context, and at the end of training the weights of the network included the word embeddings of the words from the text.

A shortage of Word2Vec was its inability to distinguish the different meanings of a word when present in different contexts. In Word2Vec, a word was only represented using one vector and therefore it had only one possible meaning, which is not the case in reality.

2.4.2 WordPiece

In 2016 Wu et al. introduced the WordPiece embedding [23]. WordPiece did not try to capture the meaning of a word within its embeddings. Instead it used a fixed vocabulary where substrings / wordpieces were mapped to an embedding based on how common they were.

The vocabulary was first initialized with all characters present in the training data. Combinations of characters were then added to the vocabulary based on what maximized the likelihood of the training data. Two consecutive

letters were added as a wordpiece if the likelihood of the training data increased with that wordpiece in the vocabulary.

When tokenizing a text using the WordPiece embeddings, all words were split into substrings based on what wordpieces were present in the vocabulary. In order to know where the spaces in a text were, all substrings that did not mark the beginning of a word (hence, were not preceded by a space), were preceded by two number signs (##).

WordPiece embeddings are not suitable for all languages since not all languages are constructed based on words separated by spaces. For such languages a better choice would for example be SentencePiece embeddings [24], which mark spaces using underscore.

2.5 Deep Learning in NLG

2.5.1 Feed-Forward Networks

The basis for any deep learning model is the perceptron. The perceptron was introduced by Frank Rosenblatt and is a linear classifier, used in supervised learning. In short, a perceptron receives an input vector which is then multiplied with weights. The values are then summed together with a constant / bias and the sum is passed through an activation function. The result from the activation function is the output from the perceptron [25]. A visualisation of a perceptron is displayed in Figure 2.1.

Combining multiple perceptrons in a layered structure results in a feed-forward network. A feed-forward network is also used in supervised learning but performed non-linear classification and prediction. It consists of three types of layers: an input layer, an output layer, and hidden layers [25]. This structure can be seen in Figure 2.2.

A feed-forward network is often trained using back propagation and gradient descent. The network can for example be trained for pattern recognition, classification or prediction.

2.5.2 RNNs and LSTMs

In order to capture the true meaning of a word, the word must be viewed in relation to its context. One way to process this type of sequential data is to use a Recurrent Neural Network (RNN) [26]. During training, an RNN tries to predict the next token in the sequence based on the previous tokens in the

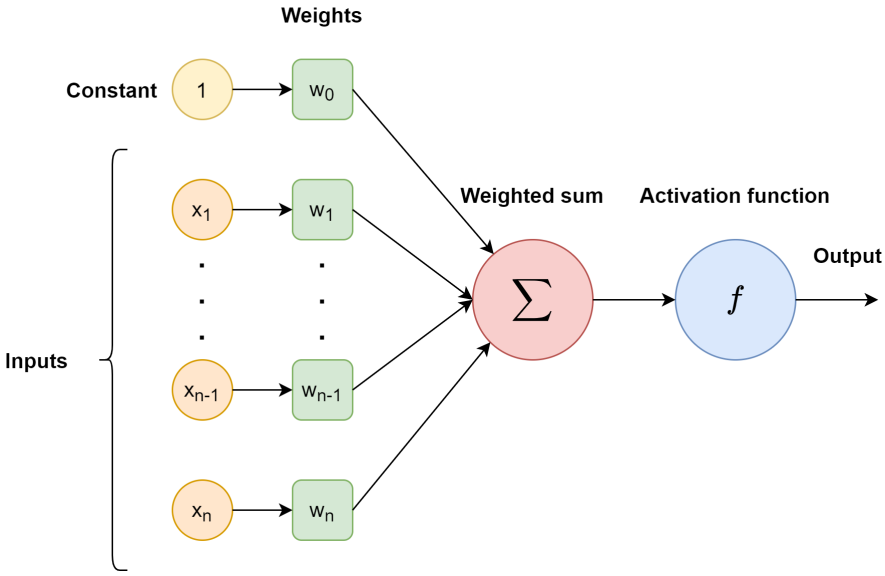

Figure 2.1: Perceptron.

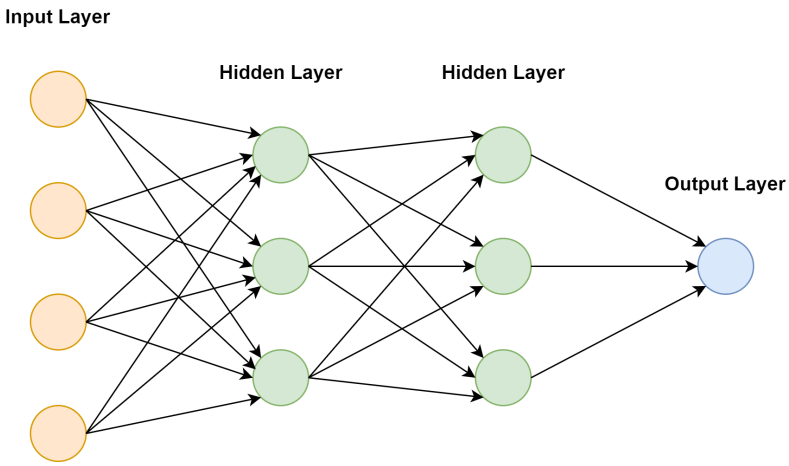

Figure 2.2: Feed-forward network.

sequence. Each cell in the RNN receives the output from the previous cell as part of its input and this way gains a sort of memory and context to each token.

One known problem with RNNs is vanishing gradients. This problem can occur when back propagation is used to calculate the gradients of the model. When a sequence is long enough and the RNN has to back propagate through several steps, the gradients can eventually become so small that the weights

in that step ceases to update. When this happens the network stops training in the earlier steps and the short term memory problem occur. [27]

Long Short Term Memory (LSTM) cells [27] are used to counteract the short term memory problem of RNNs. These cells consist of three gates, a forget gate, an input gate and an output gate, as displayed in Figure 2.3. The gates make it possible for the cell to decide what information from previous steps to keep and what to forget.

Long Short Term Memorys (LSTMs) are a more reliable version of RNNs but still have shortcomings. The LSTM is limited since it can not be truly bidirectional. It is possible to train the network in both a left-to-right and a right-to-left manner but each pass must be conducted separately and then be concatenated in order to obtain some sort of bidirectional nature.

The recurrent nature of these networks also prevent them from being trained in parallel. This is because each calculation is dependent on the previous calculation. Modern computers can therefore not fully utilise multiple threads to run calculations in parallel. Training such networks can therefore be a slow process.

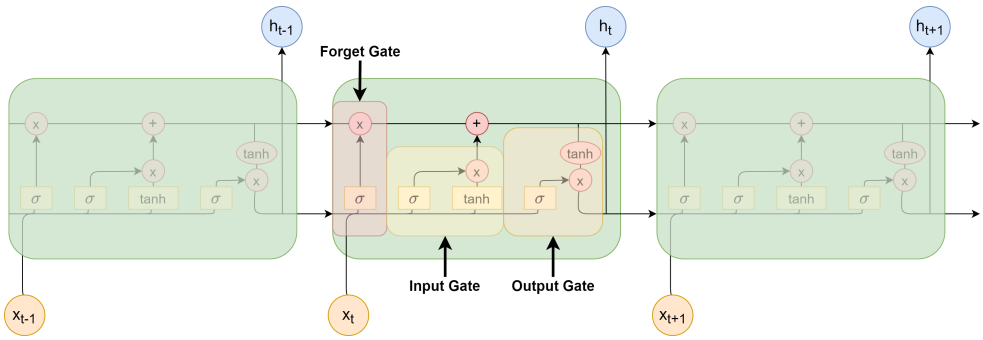

Figure 2.3: A Recurrent Neural Network (RNN) using Long Short Term Memory (LSTM) cells.

2.5.3 Encoder-Decoder Models

For sequence-to-sequence tasks such as FST, pure RNNs are not a suitable approach. This is because RNNs require the length of the input vector and the length of the output vector to be the same. In most sequence-to-sequence tasks however, the lengths of the input and output vectors can vary. Therefore, it is more common to solve sequence-to-sequence tasks using encoder-decoder models.

Encoder-decoder models consist of two main components, namely an encoder and a decoder. The purpose of the encoder is to take a sequence of some sort and encode it into a context vector that captures the essence of the sequence. In the case of written text, the context vector would represent the semantic meaning of the text. The decoder then takes this context vector and tries to decode the context vector into an output corresponding to the target output.

Using standard deep neural networks to construct an encoder-decoder model would however still mean that the length of the input and output vectors would have to be known in advance. This was noted in 2014 by Sutskever et al. [28] and Cho et al. [29] who instead proposed to use LSTMs and RNNs in encoder-decoder architectures. An example of such a model is presented in Figure 2.4.

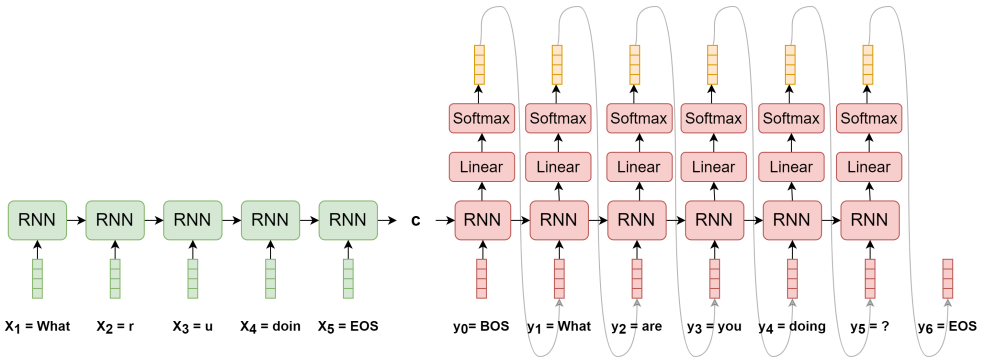

Figure 2.4: An RNN encoder-decoder model.

In their models, the encoder (the left part of Figure 2.4) received an input sequence $X_{1:n}$ and mapped this to a context vector c given by the last hidden state in the encoder. The decoder (the right part of Figure 2.4) received the context vector and a BOS token, which was needed to start the generation in the decoder. The decoder then defined the probability distribution of a target sequence $Y_{1:m}$ according to Equation 2.7. Using Bayes' rule, the probability distribution could be decomposed into conditional distributions of single target vectors according to Equation 2.8. In practice, the decoder defined a mapping between c and $Y_{0:i-1}$, and an embedded vector y'. y' was then passed through a linear transformation and a softmax operation, after which a probability distribution over all words had been produced. From this probability distribution, the next token was sampled. The decoder continued to generating new tokens until the EOS token was generated. This token marked

the end of a sequence. [28]

$$p_{dec}(Y_{1:m}|c) \tag{2.7}$$

$$p_{dec}(Y_{1:m}|c) = \prod_{i=1}^{m} p_{dec}(y_i|Y_{0:i-1}, c) \tag{2.8}$$

Sutskever et al. achieved close to state-of-the-art results on tasks such as the English-to-French translation task by compressing the input into a fixed-size context vector in the LSTM encoder and then transforming that vector into output in the LSTM decoder. The fixed-size context vectors did however limit the architecture. For input sequences larger than the size of the context vector the architecture under-performed, as stated by Bahdanau et al. [30]. This was because the model needed to capture the full content of the sequence into the context vector and there was simply too much information and not enough space in the vector.

To solve the problem of the fixed-size context vector, Bahdanau et al. introduced the attention mechanism in 2014 [30]. The attention mechanism helped the model to focus and place its attention on the more relevant pieces of the input sequence. So for a given input sequence, the model encoded the input into a series of vectors and then chose a subset of these to use when constructing the output. Attention weights described what input tokens to pay more attention to when generating the next output token. This freed the model from having to compress the full content of the input sequence into a single fixed-length context vector. The method generated output with high quality, regardless of the length of the input sequence.

2.5.4 Transformer Model

Largely based on the attention mechanism, Vaswani et al. proposed the well received Transformer model in 2017 [1]. The model did not use any RNNs but instead employed an encoder-decoder architecture composed of several layers of multi-head self-attention* sub-layers and feed forward sub-layers. A residual connection followed by layer normalization was used around each of the two sub-layers.

This model was also able to processes the input in parallel, which meant that modern computers could run parallel calculations in separate threads and

* Attention with respect to oneself.

hence decrease the computation time.

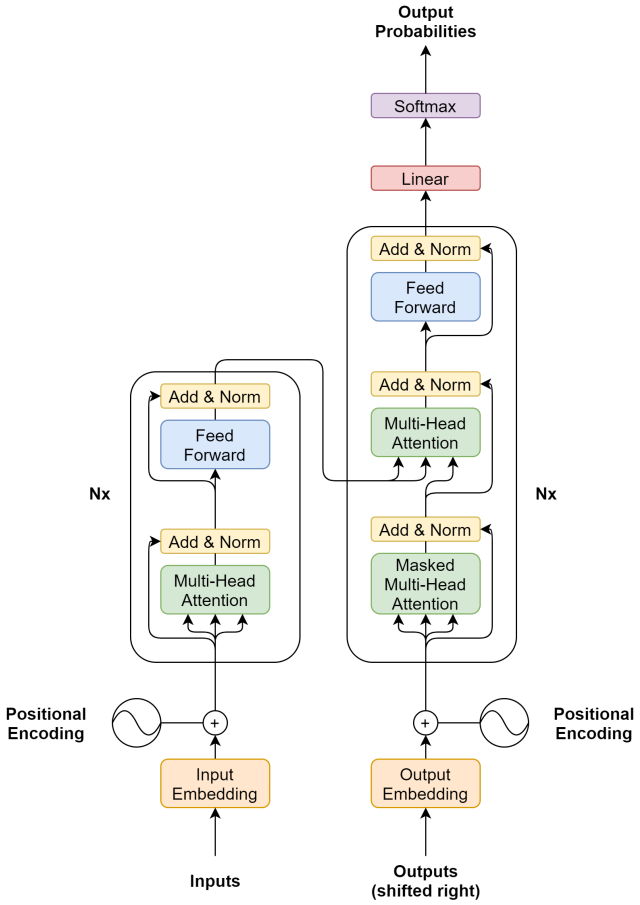

Figure 2.5: The Transformer model architecture. (Inspired by Figure 1 on page 3 of [1]).

The Transformer encoder (the left part of Figure 2.5) was composed of N=6 stacked layers that in turn consisted of a multi-head self-attention sub-layer and a feed forward sub-layer. The input sequence was first embedded and a positional encoding was added to each token embedding. The positional encoding provided the model with context information for each token, which was needed since all tokens were fed to the model simultaneously. Three linear projections were then used on the input embedding to create a *value*, *key* and *query* vector for each token embedding (V, K and Q in Figure 2.6 and 2.7 corresponds to the matrices containing these three vectors for all tokens). These three sets of vectors were then passed through a scaled dot-product

attention component, see Figure 2.6. Here a new context vector with attention was created for each token according to the steps below.

1. The scalar product of all combinations of *query* and *key* vectors were calculated. These described how similar the different vectors / tokens were.

2. The scalar products from the earlier step were then scaled by dividing them by the square root of the *query* and *key* dimensions.

3. A softmax function was then applied to obtain the weights on the values.

4. A context vector for each token was then calculated from the scalar products of the *value* vectors and the results from the earlier step.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.9}$$

The process described above was performed using matrices according to Equation 2.9. The process was repeated multiple times with different *key*, *query* and *value* vectors and the resulting context vectors were then concatenated. This process took place in the multi-head attention component, see Figure 2.7. The different *key*, *query* and *value* vectors could for example represent different semantic aspects of an input sentence.

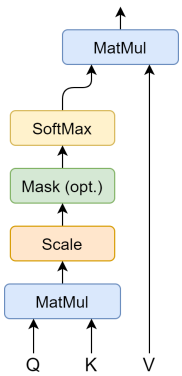

Figure 2.6: Scaled dot-product attention. (Inspired by Figure 2 (left) on page 4 of [1]).

The feed-forward sub-layer was a fully connected feed-forward network. It consisted of two linear transformations with a ReLU activation function in between.

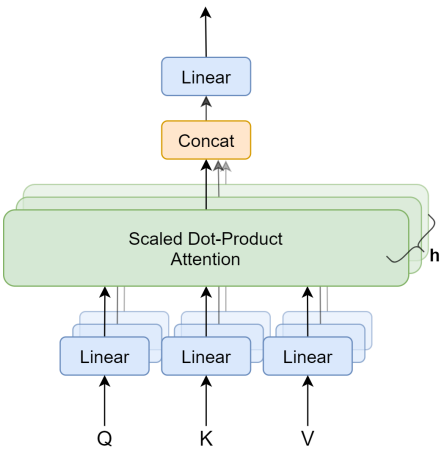

Figure 2.7: Multi-head attention. (Inspired by Figure 2 (right) on page 4 of [1]).

The decoder (the right part of Figure 2.5) was also composed of N=6 stacked layers. These layers consisted of the same two components as the encoder but also an additional third sub-layer, which performed a multi-head attention over the output from the encoder.

One important difference from the encoder was that the decoder could only see previous tokens in the input sequence. The future tokens were masked so that the decoder could predict the next token in the sequence without seeing it in advance.

Finally, the output from the decoder was passed through a linear transformation and a softmax operation to become next-token probability distributions, from where the prediction could be sampled.

2.6 Transfer Learning

Training models for NLP tasks require extensive compute and time resources. It is because of this reason, many of the more recent NLP models utilise transfer learning and more precisely, pre-training. Transfer learning is when a model developed and trained for a task is reused as a starting point for another task. In the case of modern NLP models, a pre-trained model is reused as a starting point for other NLP tasks. In order to apply a pre-trained NLP model to specific tasks there are two strategies, feature-based and fine-tuning.

The feature-based approach includes the pre-trained representations as

additional features to a task specific architecture. One example of a feature-based approach is ELMo [31]. ELMo generated features for NLP tasks through concatenating independently trained left-to-right and right-to-left LSTMs. When introduced, ELMo achieved new state-of-the-art for several NLP tasks, including sentiment analysis, named entity recognition and question answering. An illustration of the ELMo architecture can be seen in Figure 2.8.

The fine-tuning approach pre-trains a model using no task specific information and the parameters are then fine-tuned for a specific task. An example of a model using this approach is OpenAI GPT [32]. The OpenAI GPT architecture was based on the Transformer model from the paper by Vaswani et al. [1] and was constructed from stacked Transformer decoders. Because of the decoders technique of masking future tokens the OpenAI GPT was capable of text generation and achieve state-of-the-art results for many NLP tasks. It was trained using a unidirectional left-to-right architecture where every token was generated using only the previous tokens in the sequence and its source token, as input. This technique can however be very harmful to use in combination with specific NLP tasks where it is important to regard both directions, such as the question answering task and FST [2]. An illustration of the OpenAI GPT architecture can be seen in Figure 2.8.

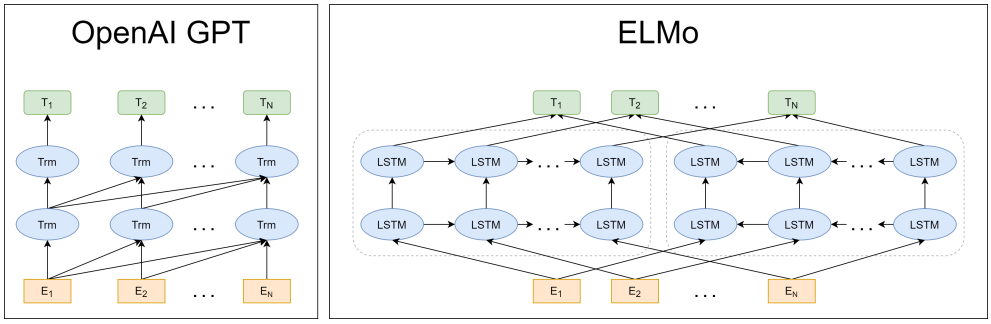

Figure 2.8: Architecture of OpenAI GPT and ELMo. (Inspired by Figure 3 on page 13 of [2]). Trm = Transformation.

The BERT model (which is thoroughly introduced in Section 2.7) was created with the aim to improve the fine-tuning approach. This was done by combining the bidirectional nature of ELMo with the approach of a masked language model, both implemented into stacked Transformer encoders. [2]

2.7 BERT

The Bidirectional Encoder Representations from Transformers (BERT) model was introduced in 2018 by Devlin et al. [2] and when introduced, the model obtained new state-of-the-art results on eleven NLP tasks. BERT was based on the Transformer model by Vaswani et al. [1] but differed from the OpenAI GPT model [32] in a few very important aspects.

Firstly, the BERT model was constructed of stacked encoders and was therefore not initially designed for generation tasks. It could however be used for generation tasks by leveraging the pre-trained weights of BERT to warm-start an encoder-decoder architecture.

Secondly, the model had a truly bidirectional nature and was therefore able to capture the context of a word from both directions simultaneously. Figure 2.9 illustrates the BERT architecture in comparison to the architectures of OpenAI GPT and ELMo in Figure 2.8.

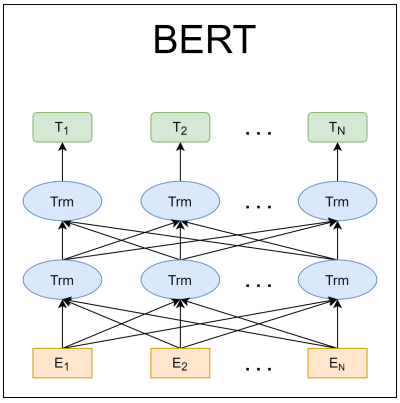

Figure 2.9: Architecture of BERT. (Inspired by Figure 3 on page 13 of [2]). Trm = Transformation.

In the paper by Devlin et al. [2], two BERT models were introduced, BERT_{BASE} and BERT_{LARGE}. BERT_{BASE} was created to have approximately the same size as OpenAI GPT, in order to compare the performances of the two models. BERT_{LARGE} on the other hand, was created to display the full potential of BERT and was therefore larger than BERT_{BASE} and took a longer time to train. BERT_{BASE} had 12 Transformer blocks, 12 self-attention heads and feed-forward networks with 786 hidden units respectively. BERT_{LARGE} had 24 Transformer blocks, 16 self-attention heads and feed-forward networks with 1024 hidden units respectively.

In order to make BERT capable of handling more tasks, its input was designed to be able to represent both a single sentence and a pair of sentences, as a single input sequence. The input was embedded using WordPiece embeddings [23] with a 30,000 token vocabulary and special tokens were used to mark different parts of the sequence. The token $[CLS]$ marked the beginning of a sequence and the token $[SEP]$ marked the ending of a sentence. In order to further distinguish the different sentences in an input sequence, a sentence embedding was added to every token. This embedding marked if the token belonged to sentence A or B. As suggested by Vaswani et al. [1], a positional embedding was also added to each token. A visualization of the input is provided in Figure 2.10.

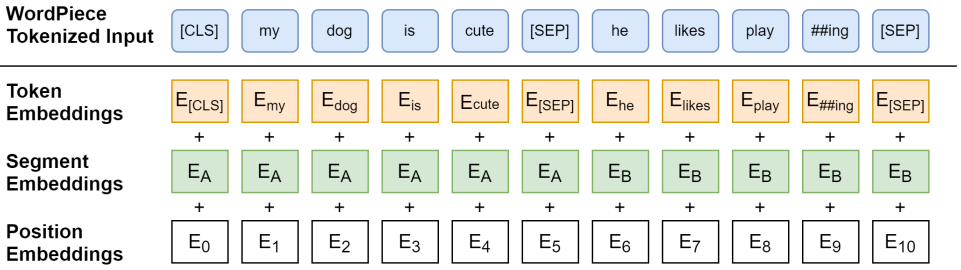

Figure 2.10: The BERT input representation. The embeddings consisted of the sum of the word embeddings, sentence embeddings and positional embeddings. (Inspired by Figure 2 on page 5 of [2]).

The pre-training of BERT was performed using two unsupervised tasks, Masked Language Model and Next Sentence Prediction, in practice however, these two tasks were performed simultaneously.

Task 1: Masked Language Model

Devlin et al. [2] claimed that a deeply bidirectional model would be more powerful than a left-to-right or right-to-left trained model. However, standard conditional language models can not be truly bidirectional since the tokens can then indirectly "see themselves" during training and the model would make trivial predictions [2]. To counteract this problem, BERT masked a percentage of the input tokens and was trained to predict these masked tokens. This way the tokens that the model was trying to predict could not "see themselves". When pre-training BERT, 15% of then tokens in an input sequence were masked at random.

Since the input sequences used when fine-tuning BERT did not contain any masked tokens, the model needed to prepare for this during pre-training as well. To do this, the masked tokens were not always replace by the token $[MASK]$. The randomly chosen masked token was replaced by the $[MASK]$ 80% of the time, replaced by another randomly chosen token 10% of the time and for the remaining 10% of the time, the token was left unchanged. The masked token was then be predicted using cross entropy loss.

Task 2: Next Sentence Prediction

An important quality in BERT was its ability to understand the relationship between sentences. When teaching the model this, two sentences were passed to the model as input. The second sentence B was the actual sentence following the first sentence A 50% of the time and the other 50% of the time, B was another random sentence. The parameter C in Figure 2.11 represents a binary number used during Next Sentence Prediction.

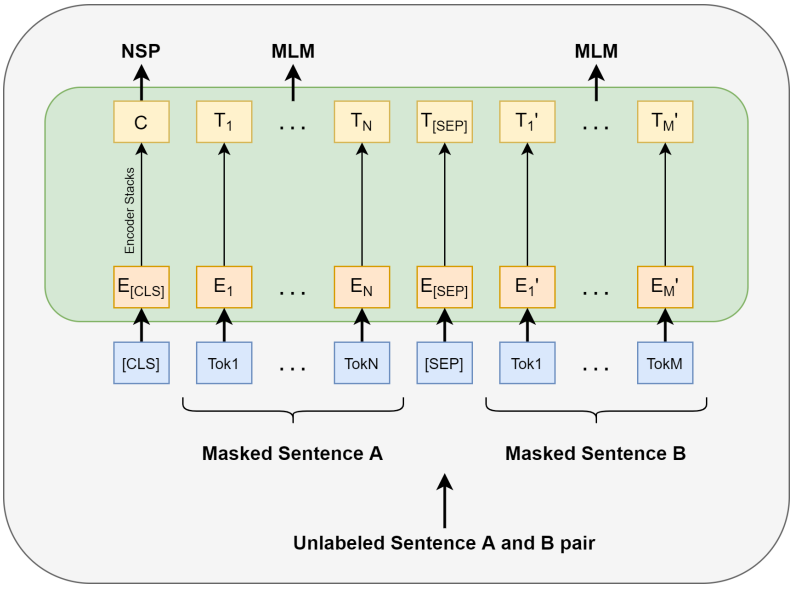

Figure 2.11: The overall pre-training procedure of BERT. (Inspired by Figure 1 on page 3 of [2]).

The pre-training of BERT was conducted using text from both the English Wikipedia (2,500M words) and BooksCorpus (800M words) [33][2]. After pre-training the BERT model had an understanding of the patterns of language and could be fine-tuned for specific tasks.

Compared to pre-training, the fine-tuning of BERT was relatively inexpensive. The task specific input and output was simple plugged into BERT and the parameters were fine-tuned end-to-end. BERT could be fine-tuned for several different NLP tasks.

2.7.1 Multilingual BERT (M-BERT)

The release of BERT also included a BERT model trained for multilingual understanding, Multilingual BERT (M-BERT) [2]. The model was trained on 104 languages simultaneously and was able to generalise across languages [34][35]. Pires et al. [34] hypothesised that the model was able to do this due to the fact that some word pieces that were shared between the various languages (such as number and URLs for example), had to be mapped into the same space. This would then force the other word pieces from different languages to also be mapped into shared spaces.

2.7.2 Swedish BERT (KB-BERT)

In 2020, the National Library of Sweden released a model based on BERT_{BASE} and pre-trained on material in Swedish [8]. Their model, called KB-BERT, was proven to outperform M-BERT as well as the Swedish BERT model produced by the Swedish Public Employment Service, Arbetsförmedlingen [8].

Like BERT_{BASE}, KB-BERT had 12 Transformer blocks, 12 self-attention heads and feed-forward networks with 786 hidden units respectively. KB-BERT was first pre-trained for one million steps with a max sequence length of 128 and a batch size of 512. Then it was trained for an additional 100,000 steps with a max sentence length of 512 and a batch size of 128. Finally the model was trained for approximately one million more steps with a max sequence length of 128 and a batch size of 512.

Input sequences were embedding using a 50,000 token WordPiece [23] vocabulary and the model was pre-trained using 260M sentence (3,497M words or 18,341MB of text) from newspaper articles, official reports of the Swedish government, social media, legal e-deposits and Swedish Wikipedia articles. There was a skew towards newspaper articles but this was also the most diverse corpus used.

The work demonstrated the importance of a large amount of data to train with as well as the lack of testbeds for the Swedish language.

2.7.3 Leveraging BERT for NLG

Since the BERT model was not initially intended for generation tasks it is not suitable for a sequence-to-sequence task such as FST in its original state. In order to utilise the pre-trained weights in BERT for FST, the BERT model can however be leveraged to warm-start an encoder-decoder model that in turn can be fine-tuned for any sequence-to-sequence task. This type of warm-started encoder-decoder model was extensively studied by Rothe et al. [36] who called the model BERT2BERT.

The BERT2BERT model consisted of an encoder and a decoder. The encoder architecture was simply a replica of a BERT architecture, since BERT is an encoder in its original state. In other words, before fine-tuning, the encoder in the BERT2BERT model behaved exactly like a pre-trained BERT model.

The decoder component of the BERT2BERT model did however differ from a standard BERT model in three ways.

Firstly, the bidirectional self-attention layers of BERT were changed into unidirectional self-attention layers. In the unidirectional layers each step could only "see" the previous tokens in the sequence. This was to prevent the model from "seeing" the next token that was to be generated. The fully connected layers that can be seen in Figure 2.9 were hence changed into layers resembling the OpenAI GPT layers in Figure 2.8.

Despite of this change, the bidirectional and unidirectional layers shared the same self-attention layer weights, e.g. the *query*, *key* and *value* projection weights of the unidirectional layers were initialized with those of BERT's bidirectional layers.

Secondly, each decoder block had to be conditioned on the context vector outputted from the encoder using cross-attention layers. Randomly initialised fully connected cross-attention layers were therefore added between the self-attention layers and the feed-forward layers in the blocks.

Thirdly, in order for the decoder to define the conditional probability distribution, an additional layer was added on top of the last decoder block. This layer mapped the output to a vector representing the similarity scores between the output vector and each word embedding. The weight parameters of this layer were initialised using the weight parameters of the word embedding. Finally a softmax operation was applied in order to obtain the conditional probability distribution.

Chapter 3

Methods

The research of this thesis was carried out in three phases.

Firstly, a pre-study was conducted, to gain relevant background knowledge in the subject.

Secondly, the GYAFC dataset was translated to Swedish and this new corpus was evaluated using an evaluation method defined in Section 3.2.1.

Thirdly, two encoder-decoder models were warm-started using BERT models and their hyper parameters were tuned. The models were then trained for both the *formal to informal* task as well as the *informal to formal* task and were then evaluated using BLEU and PINC.

The following sections describe the methodology used in the three phases.

3.1 Finding Relevant Work

Relevant work was found using the search engine Google Scholar [37]. Different combinations of the search phrases presented in Table 3.1 were used in the search engine and recent work was prioritised when browsing through the search results. Many relevant papers were also found through examining the reference lists of the papers found using keyword search on Google Scholar.

3.2 Dataset

For training, tuning and testing, the GYAFC dataset was used. For comparison purposes, the dataset was divided into training, tuning and test sets as suggested by Rao and Tetreault [4], see Table 2.5. The two domains E&M and F&R were however treated as one unit during training and tuning, resulting in the dataset displayed in Table 3.2

Keywords
BERT
Formality Style Transfer
Leveraging BERT for Text Generation
Swedish
Corpus
Google Translate
Translated Corpus
Style Transfer
Text Generation
NLP
NLG
Encoder-Decoder Models
Text Summarization
Evaluating Style Transfer in Text
Generation of Informal and Formal Sentence
LSTM
RNN

Table 3.1: List of keywords used during the pre-study.

		Informal to Formal			Formal to Informal		
Corpus	Train	Tune	Test (E&M)	Test (F&R)	Tune	Test (E&M)	Test (F&R)
GYAFC$_{en}$	104,562	5,665	1,416	1,332	4,603	1,082	1,019
GYAFC$_{swe}$	104,562	5,665	1,416	1,332	4,603	1,082	1,019

Table 3.2: Number of parallel data samples used during training, tuning and testing from the two corpus used in this thesis. During tuning and testing, different samples were used for the *informal to formal* task and the *formal to informal* task.

The Swedish corpus was constructed by translating the English corpus. This was done using the Google Translate API [38]. To prevent the sentences from affecting the translation of one another, the corpus was not translated as one unit but instead each sentence was translated as an individual.

From here on the original English GYAFC dataset will be referred to as GYAFC$_{en}$ and the Swedish version will be referred to as GYAFC$_{swe}$.

3.2.1 Dataset Evaluation

In order to compare the two FST models proposed in this thesis (described in Section 3.3), the quality of GYAFCswe and GYAFC$_{en}$ needed to be known. If

one of the corpus was inferior to the other, this might justify the model trained with that corpus to be inferior to the other model.

Since all sentences in GYAFC$_{en}$ were manually evaluated upon creation, it was reasonable to assume that the corpus had a high quality. The quality of GYAFCswe was however unknown and needed evaluation. It was therefore decided that GYAFCswe was to be evaluated in regard to how well the content had been preserved during translation from GYAFCen. For this task, inspiration was drawn from Rao and Tetreault [4], who evaluated their FST results according to the three criteria *formality*, *fluency* and *meaning preservation*, see Section 2.3.3. Since previous studies had deemed the criteria representative for the quality of FST, it was decided to use variants of the criteria for evaluating translations used for FST.

The *meaning preservation* score was basically used as suggested by Rao and Tetreault [4] but was shifted so that the scale went from zero to five instead of one to six (see Table 3.3).

Score	Meaning Preservation Scale
0	Completely dissimilar
1	Not equivalent but on same topic
2	Not equivalent but share some details
3	Roughly equivalent
4	Mostly equivalent
5	Completely equivalent

Table 3.3: The *meaning preservation* scale used for translation evaluation.

The *fluency* score was transformed into a *fluency preservation* score, by first scoring both the original English GYAFC$_{en}$ sentences and the translated Swedish GYAFC$_{swe}$ sentence using the *fluency* score in Table 2.7. The maximum difference possible between these two scores was four and therefore the score was calculated by subtracting the absolute difference of the two scores from four (see Equation 3.1). The *fluency preservation* score could hence take on a value between zero and four.

$$\begin{aligned} &fluency\ preservation\ score = \\ &\qquad 4 - |fluency_{English\ sentence} - fluency_{Swedish\ sentence}| \end{aligned} \tag{3.1}$$

The *formality* score was converted into a *formality preservation* score on the same basis as the *fluency* score. Here the maximum possible difference between the English and Swedish *formality* scores was six and therefore the

new score was calculated according to Equation 3.2 and could take on a value between zero and six.

$$formality\ preservation\ score = \\ 6 - |formality_{English\ sentence} - formality_{Swedish\ sentence}| \tag{3.2}$$

Whenever a sentence pair obtained a *formality preservation* score that was less than the maximum value six, it was also noted whether or not the Swedish sentence was *more* or *less* formal than the English original. This was done in order to know if the automatic translation had a tendency to generate sentences of a specific style.

By scoring 100 sentences from GYAFC$_{swe}$ using the three scores *meaning preservation*, *fluency preservation* and *formality preservation*, and then calculating the mean value for each score, the full corpus was given three overall scores. By dividing the scores with their respective maximum value, a percentage of the scores were also provided. A percentage of 100, indicated perfect preservation and a percentage of zero, indicated no preservation.

3.3 Model Design

Two models were trained for two separate FST tasks, namely the *formal to informal* task and the *informal to formal* task. The models were both constructed as BERT2BERT models, as described in Section 2.7.3. This architecture was chosen in order to leverage existing bidirectional models, that had been pre-trained specifically for Swedish respectively English. From here on the English BERT2BERT model implemented in this thesis will be referred to as BERT2BERT$_{en}$ and the Swedish version will be referred to as BERT2BERT$_{swe}$.

The BERT2BERT$_{en}$ model leveraged a cased version of BERT$_{BASE}$ for warm-starting its encoder-decoder model and was trained, tuned and tested on the English GYAFC$_{en}$ dataset.

The BERT2BERT$_{swe}$ model leveraged a cased version of KB-BERT for warm-starting its encoder-decoder model and was trained, tuned and tested on the Swedish GYAFC$_{swe}$ dataset.

The reason for choosing these two versions of BERT for warm-starting the encoder-decoder models, was that the two versions are similar in both pre-training procedure and performance. KB-BERT was also chosen over M-BERT since it has been proven to outperform M-BERT.

The National Library of Sweden also provided a version of the ALBERT model [39][40], which is a lite BERT model. However, the library did not provide any evaluation of this model in comparison to the original English ALBERT model and therefore the BERT model was chosen over ALBERT in this thesis. The BERT model also constitutes the basis of ALBERT, which means that any results obtained in this thesis should be applicable to the ALBERT model as well.

3.3.1 Hyper Parameter Tuning

The hyper parameters of the models were tuned manually using the tuning datasets. The hyper parameters that were investigated were batch size, learning rate and the number of training epochs. Different combinations of the parameters were tested and the variation that yielded the best PINC and BLEU scores for the tuning dataset were chosen.

When investigating the performance based on batch size, it was made clear that a larger batch size outperformed smaller ones both in terms of score and training time. The memory constraints of the available GPU used for training did however limit the batch size to 48.

When choosing the learning rate, the goal was to gain an increase in performance to the models after each epoch without oscillating scores. The learning rate was very correlated to the batch size, such that a smaller batch size benefited from a smaller learning rate and a larger batch size benefited a larger learning rate.

The number of epochs was chosen after the learning rate and batch size had been decided. It was chosen with the goal of maximizing tuning scores and not overfitting the model.

The final hyper parameters are presented in Table 3.4.

Hyper Parameter	Value
Batch Size	48
Learning Rate	0.00005
Number of Epochs	4

Table 3.4: The hyper parameter values used during fine-tuning of the final models.

3.3.2 Model Evaluation

BERT2BERT$_{en}$ and BERT2BERT$_{swe}$ were evaluated using the two metrics BLEU and PINC which are described in Sections 2.3.1 and 2.3.2. The BLEU score was used to evaluate how well the sentences generated by the models matched the reference sentences provided in their respective corpus and the PINC score was used to evaluate how many new tokens had been introduced into the generated sentences in comparison to the source sentences.

These two scores were chosen because they have been used in other research which made comparisons possible. Other advantages were that the scores were automatic and did not require any prior training or evaluation, which would have been the case if for example a classifier for identifying formality would have been introduced.

In Chapter 5, the two scores are both presented in comparison to each other, as well as presented in relation to the lengths of the source sentences. This was done in order to see the quality of the models in comparison to different types of sentences, different lengths of sentences in this case. The plots were constructed by dividing the generated sentence into groups based on the lengths of their source sentences. Each group was then scored with both the BLEU and PINC metric and a third-degree polynomial was fitted to the data.

3.4 Reliability and Validity

Since GYAFC$_{swe}$ had an inferior quality compared to GYAFC$_{en}$, the validity of the comparison between BERT2BERT$_{swe}$ and BERT2BERT$_{en}$ was compromised. In order to counteract this, the preservation of the translated corpus, GYAFC$_{swe}$, was evaluated. The results of this evaluation was taken into account when comparing the two models, so that the quality of the models could be analysed with respect to their own conditions.

For the same purpose, the BLEU and PINC scores of the original source sentences and reference sentences in the two corpus were calculated. Using these scores and the output scores of the two models, the models were also evaluated with regard to how much they improved their respective input data.

In order to increase the reliability of the results, the statistical significance of the results was calculated. The models scores were compared to the BLEU and PINC scores of the original source sentences and reference sentences, to see if the models scores were significantly different from the BLEU and PINC scores of the corpus, or if any difference in the scores was simply coincidental.

This was done using a two-tailed t-test.

A t-test assumes that the means of the different samples are normally distributed. The central limit theorem states that the distribution of the sample means of a population will be approximately normally distributed, provided the sample size is sufficiently large. Based on this theorem and the fact the the sample size used in this thesis was deemed sufficiently large, a two-tailed t-test could be reliably used for the significance test.

Chapter 4

Development

This chapter presents the practical aspects of the method. The first section describes how the Swedish corpus was evaluated and the second section presents the libraries, frameworks, hardware and software that were used to implement and run the two BERT2BERT models.

4.1 Human-Based Evaluation of GYAFC$_{swe}$

The GYAFC$_{swe}$ dataset was manually evaluated using the method defined in section 3.2.1. Evaluation was performed on a subset of 100 GYAFC$_{swe}$ sentences, equally divided between the E&M and F&R domains and the informal and formal sentences in both subsets, according to Table 4.1. The sentences were chosen so that both sentences in a parallel informal / formal sentences pair were evaluated. In other words, if an informal E&M sentence was chosen for evaluation, the formal version of the same sentence was also evaluated.

Style \ Domain	E&M	F&R
Formal	25	25
Informal	25	25

Table 4.1: The number of GYAFC$_{swe}$ sentences evaluated from each subset of data.

GYAFC$_{en}$ sentences	GYAFC$_{swe}$ sentences
Thank you for the notification, I will not allow my children to watch that.	Tack för meddelandet, jag tillåter inte att mina barn tittar på det.
However, yes, your statement is correct.	Men ja, ditt uttalande är korrekt.
I don't think there's a song of his I didn't like.	Jag tror inte att det finns en sång av hans jag inte gillade.

Table 4.2: Examples of three sentences from GYAFC$_{en}$ and their respective GYAFC$_{swe}$ translations.

Examples

In Table 4.2 three examples of sentences from GYAFC$_{en}$ are displayed together with their respective GYAFC$_{swe}$ translations. In the first sentence pair, the GYAFC$_{swe}$ sentence received the maximum *formality preservation* and *fluency preservation* scores but the meaning of the sentence had changed since "I will not allow" implied that it will not be allowed in the future, but "jag tillåter inte" implies that it is not allowed presently. A more accurate translation would for example be "Tack for meddelandet, jag kommer inte tillåta mina barn att titta på det". The translation did therefore receive a *meaning preservation* score of 3 out of 5 (Roughly equivalent).

In the second sentence pair, the GYAFC$_{swe}$ sentence did not receive a full *formality preservation* score. The GYAFC$_{en}$ sentence received a score of 2 (Formal) but the GYAFC$_{swe}$ sentence only received a *formality* score of 1 (Somewhat Formal). This assessment was done based on the Swedish phrase "Men ja," (in English "But yes,") and the notion that this phrase is not quite as formal as "However, yes,". The GYAFC$_{swe}$ sentence was therefore given a *formality preservation* score of 5 out of 6, ($6 - |2 - 1| = 5$).

In the third sentence pair, the GYAFC$_{swe}$ sentence did not receive a full *fluency preservation* score. The meaning of the GYAFC$_{swe}$ sentence was clear but the grammar was not correct, which affected the fluency. The GYAFC$_{en}$ sentence received a *fluency* score of 4 (Comprehensible) but the GYAFC$_{swe}$ sentence only received a *formality* score of 3 (Somewhat Comprehensible). The GYAFC$_{swe}$ sentence therefore received a *fluency preservation* score of 3 out of 4, ($4 - |4 - 3| = 3$).

4.2 Model Implementation

Both BERT2BERT$_{en}$ and BERT2BERT$_{swe}$ were implemented using Python [41] code, the Huggingface Transformers library [42] and initialised using a Huggingface EncoderDecoderModel [43]. The BERT2BERT$_{en}$ model used

the pre-trained *bert-base-cased* model [44] to warm-start the EncoderDecoderModel, and BERT2BERT$_{swe}$ used the pre-trained *KB / bert-base-swedish-cased* model [40] for the same purpose. The sequence length was limited to 128 tokens for both models and the text was tokenized using the Huggingface BertTokenizer library [45], which was initialized using the vocabularies from the respective models.

For fine-tuning, a transformers Seq2SeqTrainer [46] was used, which utilises the Adam Algorithm for training [47]. Based on empirical investigations described in Section 3.3.1, the following hyper parameters were selected. A batch size of 48, an initial learning rate of 0.00005 and four epochs of training.

4.2.1 Hardware and Software

The BERT code was originally written in C++, but the authors also released pre-trained versions in Python [41] using the Tensorflow library [48]. The pre-trained KB-BERT model was however only released in Python using the PyTorch library [49] hosted on Huggingface [40][8]. Since NLP researchers from Huggingface has made a PyTorch version of the English BERT [44], that could reproduce the results of the original BERT, it was decided that Huggingface and PyTorch was to be used for the implementation in this project. This meant that both BERT2BERT$_{en}$ and BERT2BERT$_{swe}$ could be fine-tuned using the same code and results could be compared with a greater reliability.

Another reason for using PyTorch was due to its support of the NVIDIA apex package [50]. This package enables a mixed floating point precision that can scale down the default float32 values to float16 values. This can reduce GPU memory constraints, which in turn allows larger batch sizes to be used.

To reduce the time of fine-tuning, Google Cloud Platform [38] was used. A Compute Engine Instance was created which ran the operating system Debian 10 [51], Python 3.7 [41], PyTorch 1.7 [49] and NVIDIA CUDA 11.0 [50]. It was initialized using the Deep Learning VM, version M65 [52] and used an N1 high memory machine. The machine ran 2 vCPUs with 6.50 GB of system memory each. The VM also used a NVIDIA Tesla T4 GPU [53].

Chapter 5

Results and Analysis

The following chapter contains the results of the thesis. The chapter is divided into two sections. The first section presents the results from the evaluation of the Swedish corpus and the second section presents the results from the evaluation of BERT2BERT$_{swe}$ and BERT2BERT$_{en}$.

5.1 Quality of GYAFC$_{swe}$

In Table 5.1, the results from the evaluation of GYAFC$_{swe}$ are shown.

- The *meaning preservation* score of the 100 evaluated GYAFC$_{swe}$ sentences was 4.30 out of 5 (86.0%) and ranged between 4.08 (81.6%) and 4.52 (90.4%) amongst the different subsets of evaluated sentences.
- The *fluency preservation* score of the 100 evaluated GYAFC$_{swe}$ sentences was 3.75 out of 4 (93.8%) and ranged between 3.64 (91.0%) and 3.88 (97.0%) amongst the different subsets of evaluated sentences.
- The *formality preservation* score of the 100 evaluated GYAFC$_{swe}$ sentences was 5.72 out of 6 (95.3%) and ranged between 5.64 (94.0%) and 5.80 (96.7%) amongst the different subsets of evaluated sentences.

It is clear from the last two columns in Table 5.1, that the formal sentences which did not receive a perfect *formality preservation* score, were in general less formal than their original English sentences. It is also clear that the informal sentences which did not receive a perfect *formality preservation* score, were in general more formal than their original English sentences.

In Figures 5.1, 5.2 and 5.3, the distributions of the *meaning preservation* score, *formality preservation* score and *fluency preservation* score are shown

in relation to each other. These findings showed that most sentences preserved much of their *meaning*, *formality* and *fluency* after being translated but also that the *meaning preservation* score had a wider distribution than the other two scores.

When a translated sentence did not receive a perfect *meaning preservation* score, the new meaning of the sentence was not necessarily given to both the formal and informal versions in the parallel sentence pair. Often one sentence in the pair was given the new meaning and the other sentence preserved its original meaning and therefore had a higher *meaning preservation* score.

Domain	Style	#sentences	Meaning PS [0, 5]	Fluency PS [0, 4]	Formality PS [0, 6]	More Formal	Less Formal
E&M	Formal	25	4.24 (84.8%)	3.64 (91.0%)	5.64 (94.0%)	1	5
E&M	Informal	25	4.52 (90.4%)	3.88 (97.0%)	5.68 (94.7%)	6	1
F&R	Formal	25	4.36 (87.2%)	3.72 (93.0%)	5.80 (96.7%)	1	4
F&R	Informal	25	4.08 (81.6%)	3.76 (94.0%)	5.76 (96.0%)	5	1
E&M	Both	50	4.38 (87.6%)	3.76 (94.0%)	5.66 (94.3%)	7	6
F&R	Both	50	4.22 (84.4%)	3.74 (93.5%)	5.78 (96.3%)	6	5
Both	Formal	50	4.30 (86.0%)	3.68 (92.0%)	5.72 (95.3%)	2	9
Both	Informal	50	4.30 (86.0%)	3.82 (95.5%)	5.72 (95.3%)	11	2
Both	Both	100	4.30 (86.0%)	3.75 (93.8%)	5.72 (95.3%)	13	11

Table 5.1: *Meaning preservation* scores (PS), *fluency preservation* scores (PS) and *formality preservation* scores (PS) of 100 GYAFC$_{swe}$ sentences from the training subset. The Domain column describes from which domain the evaluated sentences in each line were taken from and the Style column describes which style the sentences in each line had. The #sentences column displays the number of sentences that were evaluated within that domain and style. In the Meaning PS, Fluency PS and Formality PS columns, the first value is the score of the sentences and the value within parentheses is the score divided by the highest possible score for that criteria. The value within parentheses hence represents to what percentage the sentences were preserved with regard to the specific criteria. The last two columns present the number of GYAFC$_{swe}$ sentences that did not have a perfect *formality preservation* score and whether these sentences where *more* respectively *less* formal than their original English sentence.

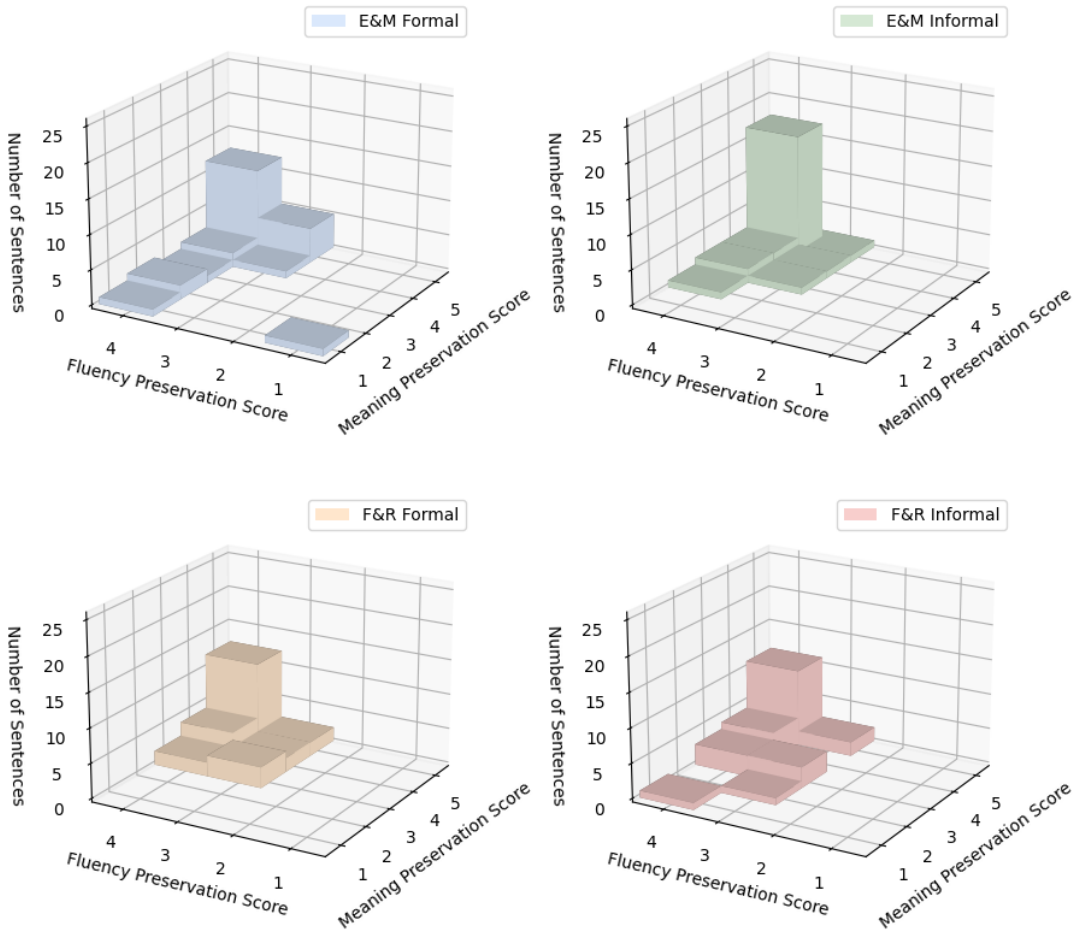

Figure 5.1: The distribution of the evaluated GYAFC$_{swe}$ sentences *meaning preservation* scores compared to their *fluency preservation* scores.

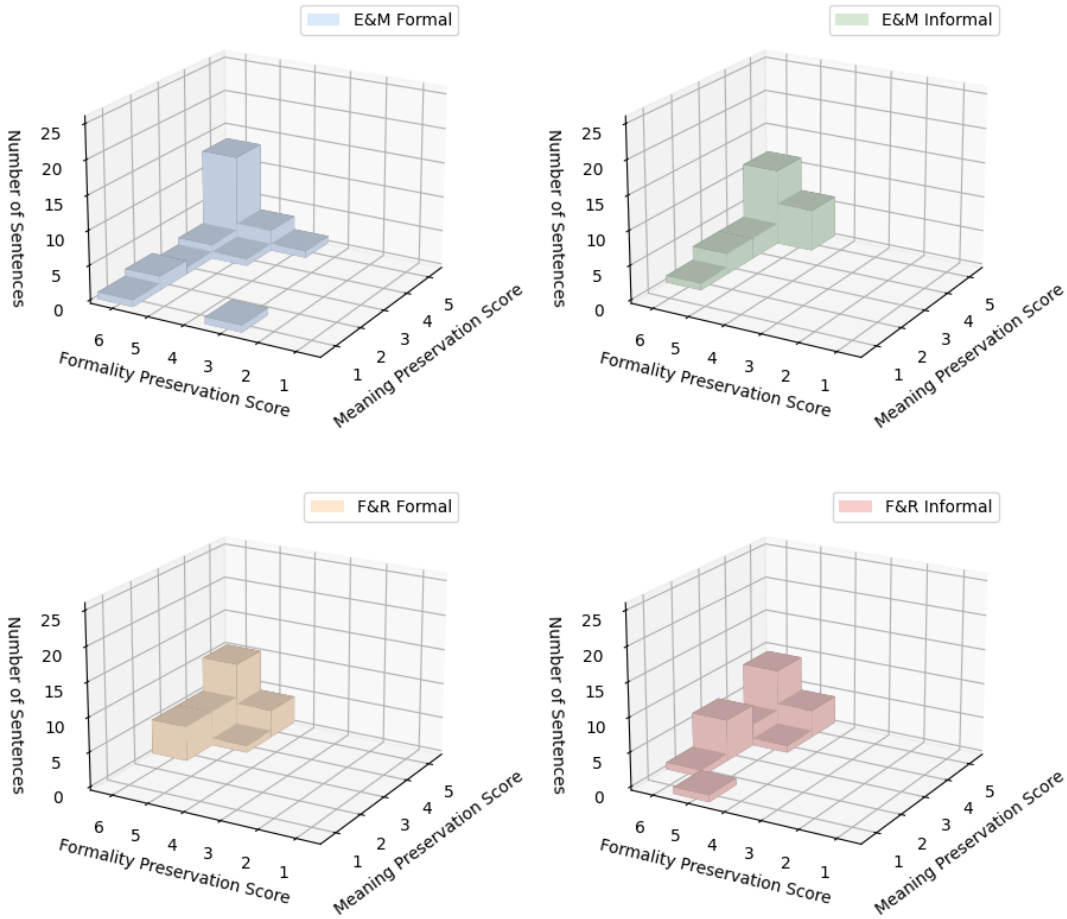

Figure 5.2: The distribution of the evaluated GYAFC$_{swe}$ sentences *meaning preservation* scores compared to their *formality preservation* scores.

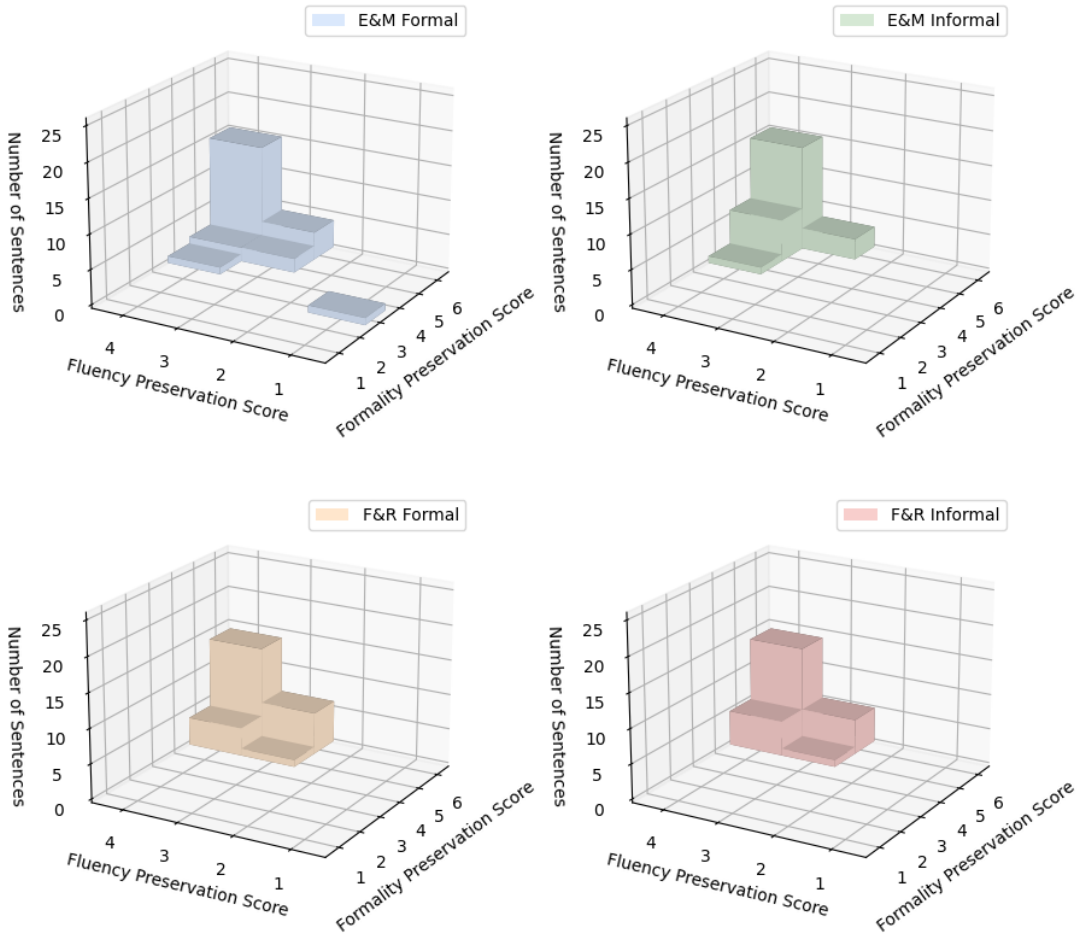

Figure 5.3: The distribution of the evaluated GYAFC$_{swe}$ sentences *formality preservation* scores compared to their *fluency preservation* scores.

5.2 Formality Style Transfer (FST)

Tables 5.2, 5.3, 5.4 and 5.5 display the final BLEU and PINC scores for BERT2BERT$_{en}$ and BERT2BERT$_{swe}$ on the *informal to formal* task and the *formal to informal* task. For comparison purposes the scores of the original source sentences and the reference sentences are also shown in the same tables.

It was observed from these tables that:

- The F&R domain received higher BLEU scores than the E&M domain.
- The F&R domain received lower PINC scores than the E&M domain.

This was expected since it was the same behavior as reported by other studies, see Tables 2.3, 2.2 and 2.4.

It was also noted that:

- All reference sentences received a BLEU score of 100.0.
- All original source sentences received a PINC score of 0.00.

The BLEU score measured how similar a set of sentences were to their respective reference sentences and a higher BLEU score indicated more similarities. Hence, it was not unexpected that the reference sentences received the maximum BLEU score, since they were perfect matches of themselves.

The PINC score measured how much a set of sentences differed from their respective source sentences and a lower PINC score indicated less differences. Hence, it was neither unexpected that the original source sentences received the lowest possible PINC score, since they did not differ at all from themselves.

From Tables 5.2, 5.3, 5.4 and 5.5 it was also observed that:

- GYAFC$_{swe}$ received higher BLEU scores than GYAFC$_{en}$ on the original source sentences.
- GYAFC$_{swe}$ received lower PINC scores than GYAFC$_{en}$ on the reference sentences.

Based on the previously mentioned definitions of the BLEU and PINC scores, these results indicated that after translation, the formal and informal sentences approached each other in word and phrase structure, and therefore also in *formality*.

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Informal GYAFC$_{en}$	50.35	0.00	51.77	0.00
Formal References GYAFC$_{en}$	100.0	67.85	100.0	66.08
BERT2BERT$_{en}$	64.50*	56.57*	69.35*	53.01*

Table 5.2: BLEU and PINC scores of the GYAFC$_{en}$ test set for the *informal to formal* task. Scores marked with * were significantly different from the original BLEU score / reference PINC score with a p-value < 0.01.

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Informal GYAFC$_{swe}$	57.72	0.00	60.93	0.00
Formal References GYAFC$_{swe}$	100.0	63.10	100.0	60.25
BERT2BERT$_{swe}$	59.52	48.77*	64.69*	41.93*

Table 5.3: BLEU and PINC scores of the GYAFC$_{swe}$ test set for the *informal to formal* task. Scores marked with * were significantly different from the original BLEU score / reference PINC score with a p-value < 0.01.

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Formal GYAFC$_{en}$	30.22	0.00	30.30	0.00
Informal References GYAFC$_{en}$	100.0	79.22	100.0	78.39
BERT2BERT$_{en}$	38.28*	55.84*	39.33*	50.13*

Table 5.4: BLEU and PINC scores of the GYAFC$_{en}$ test set for the *formal to informal* task. Scores marked with * were significantly different from the original BLEU score / reference PINC score with a p-value < 0.01.

	E&M		F&R	
Model	BLEU	PINC	BLEU	PINC
Original Formal GYAFC$_{swe}$	37.17	0.00	39.98	0.00
Informal References GYAFC$_{swe}$	100.0	75.96	100.0	73.25
BERT2BERT$_{swe}$	40.18	45.96*	45.40*	39.76*

Table 5.5: BLEU and PINC scores of the GYAFC$_{swe}$ test set for the *formal to informal* task. Scores marked with * were significantly different from the original BLEU score / reference PINC score with a p-value < 0.01.

Tables 5.2 and 5.3 present the results for the *informal to formal* task. From these two tables it was observed that:

- BERT2BERT$_{en}$ received higher BLEU scores than BERT2BERT$_{swe}$ on the *informal to formal* task.
- BERT2BERT$_{en}$ received higher PINC scores than BERT2BERT$_{swe}$ on the *informal to formal* task.
- BERT2BERT$_{en}$ had a bigger difference between the BLEU scores of the original source sentences and the models, than BERT2BERT$_{swe}$, on the *informal to formal* task.

These results did not only suggest that the BERT2BERT$_{en}$ model produced more satisfactory sentences than the BERT2BERT$_{swe}$ model, but also that the BERT2BERT$_{en}$ model had learnt more during training than the BERT2BERT$_{swe}$ model.

The results for the *formal to informal* task are presented in Tables 5.4 and 5.5. From these two tables it was observed that:

- BERT2BERT$_{en}$ received lower BLEU scores than BERT2BERT$_{swe}$ on the *formal to informal* task.
- BERT2BERT$_{en}$ received higher PINC scores than BERT2BERT$_{swe}$ on the *formal to informal* task.
- BERT2BERT$_{en}$ had a bigger difference between the BLEU scores of the original source sentences and the models, than BERT2BERT$_{swe}$, on the *formal to informal* task.

Although the BERT2BERT$_{swe}$ model obtained higher BLEU scores than the BERT2BERT$_{en}$ model on this task, these results also indicated that the BERT2BERT$_{en}$ model had learnt more during training than the BERT2BERT$_{swe}$ model.

Figures 5.4, 5.5, 5.6 and 5.7 displays the BLEU and PINC scores of sentences generated by BERT2BERT$_{en}$ and BERT2BERT$_{swe}$ for both domains, in relation to the lengths of the original source sentences. From these figures it was observed that:

- All informal source sentences were shorter than 128 characters.
- Most formal source sentences were shorter than 128 characters.
- All formal source sentences were shorter than 200 characters.
- Most sentences were between 30 to 80 characters long.
- The highest BLEU scores were obtained at a sentence length of approximately 30 to 80 characters.
- The lowest PINC scores were obtained at a sentence length of approximately 30 to 80 characters.
- The lowest BLEU scores were obtained at a sentence length of approximately 0 to 30 characters and 80 to 200 characters.
- The highest PINC scores were obtained at a sentence length of approximately 0 to 30 characters and 80 to 200 characters.

These results confirmed that the two scores behaved as the inverse of each other. When the BLEU score increased, the PINC score decreased, and vice versa. These results also suggested that the BLEU score increased for sentence lengths where much data was provided and that the PINC score decreased under then same circumstances.

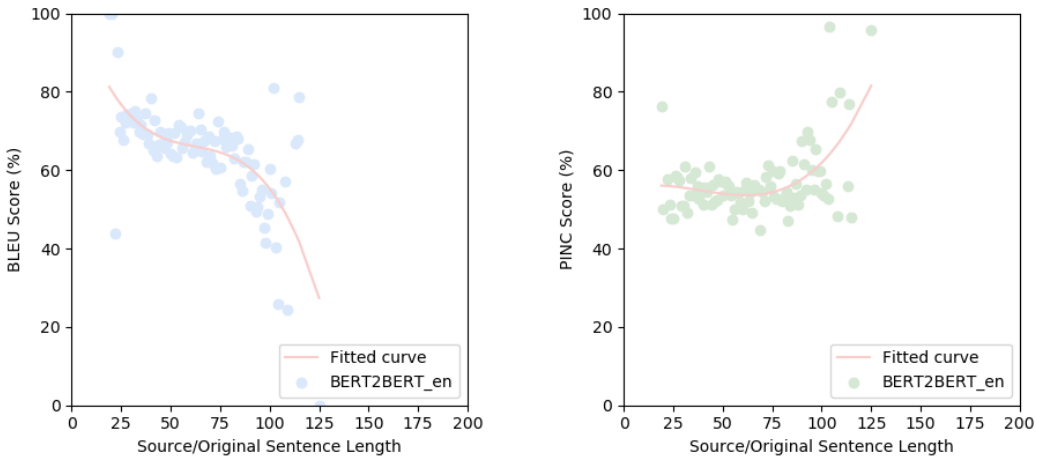

Figure 5.4: BLEU and PINC scores in relation to the length of the source sentence for BERT2BERT$_{en}$ on the *informal to formal* task. The scores were calculated using the full test dataset and the domains E&M and F&R were combined.

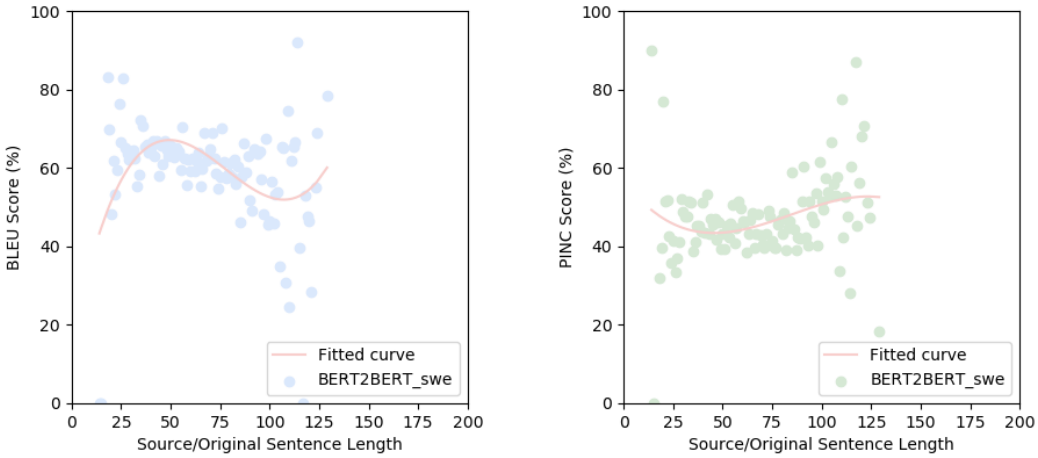

Figure 5.5: BLEU and PINC scores in relation to the length of the source sentence for BERT2BERT$_{swe}$ on the *informal to formal* task. The scores were calculated using the full test dataset and the domains E&M and F&R were combined.

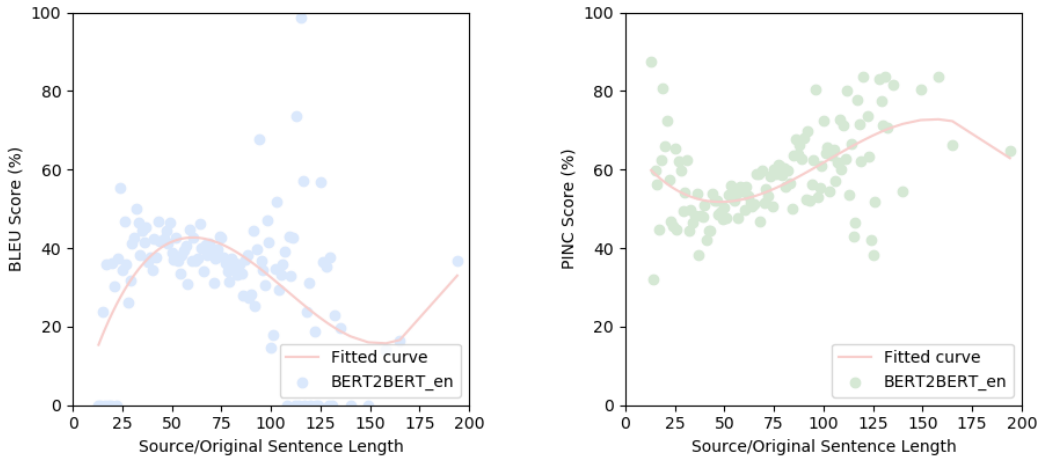

Figure 5.6: BLEU and PINC scores in relation to the length of the source sentence for BERT2BERT$_{en}$ on the *formal to informal* task. The scores were calculated using the full test dataset and the domains E&M and F&R were combined.

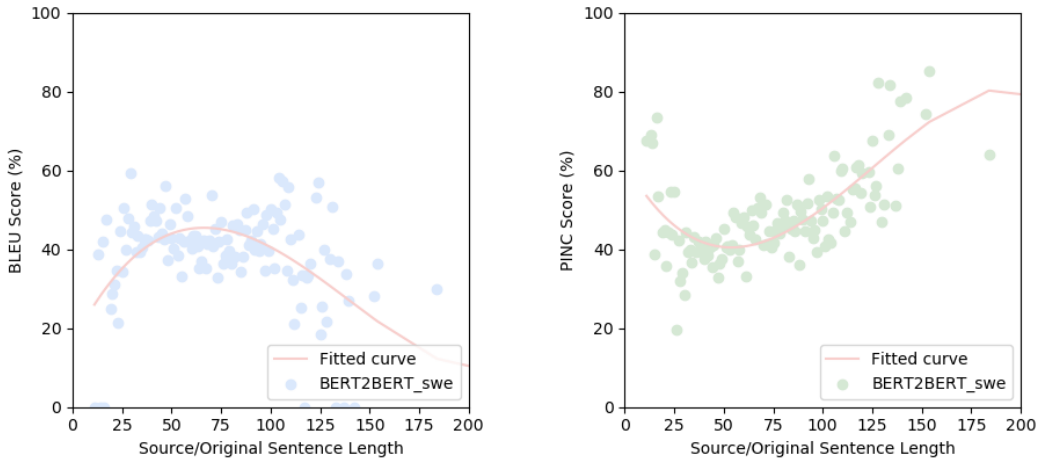

Figure 5.7: BLEU and PINC scores in relation to the length of the source sentence for BERT2BERT$_{swe}$ on the *formal to informal* task. The scores were calculated using the full test dataset and the domains E&M and F&R were combined.

In Tables 5.6, 5.7, 5.8 and 5.9, samples of sentence from the two corpus are presented together with their corresponding model output.

The observed behaviors of the BERT2BERT$_{en}$ model were the following:

- The BERT2BERT$_{en}$ model appeared to be able to both add (in the *formal to informal* task) and remove (in the *informal to formal* task) contractions.
- The BERT2BERT$_{en}$ model appeared to be able to change letters from lowercase to capital and vice versa.
- The BERT2BERT$_{en}$ model appeared to be able to concatenate sentences for a more informal style and divide longer sentences into several shorter sentences for a more formal style.
- Sometimes, the BERT2BERT$_{en}$ model failed to generate a sentence with the same meaning as the original source sentence.

The observed behaviors of the BERT2BERT$_{swe}$ model were:

- The BERT2BERT$_{swe}$ model appeared to be able to change letters from lowercase to capital and vice versa.
- The BERT2BERT$_{swe}$ model appeared to be able to concatenate sentences for a more informal style and divide longer sentences into several shorter sentences for a more formal style.
- Sometimes, the BERT2BERT$_{swe}$ model failed to generate a sentence with the same meaning as the original source sentence.
- Occasionally, the BERT2BERT$_{swe}$ model generated the same sentence as the source sentence.

Entertainment & Music (E&M)	
Original informal	i don't no what r u talking about.
Reference 0 formal	I'm sorry, I do not know what you are referring to.
Reference 1 formal	I am not sure what you are talking about.
Reference 2 formal	I do not know what you are talking about.
Reference 3 formal	I do not know what you are talking about.
BERT2BERT$_{en}$ output	I do not know what you are talking about.
Family & Relationships (F&R)	
Original informal	Remember, TILL DEATH DO YOU PART!!
Reference 0 formal	Remember, untill death do you part.
Reference 1 formal	Remember that it is "Till death do you part".
Reference 2 formal	Remember, "Until death do you part".
Reference 3 formal	Remember, the vow is until death do you part!
BERT2BERT$_{en}$ output	Remember, " Death, will you pay? "

Table 5.6: Samples of sentences from the GYAFC$_{en}$ dataset from both the E&M and the F&R domains, with the corresponding model output. The samples were taken from the test dataset on the *informal to formal* task. [4]

Entertainment & Music (E&M)	
Original informal	Jag vet inte vad du pratar om.
Reference 0 formal	Jag är ledsen, jag vet inte vad du hänvisar till.
Reference 1 formal	Jag är inte säker på vad du pratar om.
Reference 2 formal	Jag vet inte vad du pratar om.
Reference 3 formal	Jag vet inte vad du pratar om.
BERT2BERT$_{swe}$ output	Jag vet inte vad du pratar om.
Family & Relationships (F&R)	
Original informal	Kom ihåg, TILL DÖDEN DELAR DU !!
Reference 0 formal	Kom ihåg, tills döden skiljer sig.
Reference 1 formal	Kom ihåg att det är "Till döden skiljer sig från dig".
Reference 2 formal	Kom ihåg "Till döden skiljer du dig".
Reference 3 formal	Kom ihåg att löftet är tills döden skiljer sig!
BERT2BERT$_{swe}$ output	Kom ihåg att du delar.

Table 5.7: Samples of sentences from the GYAFC$_{swe}$ dataset from both the E&M and the F&R domains, with the corresponding model output. The samples were taken from the test dataset on the *informal to formal* task. [4]

Entertainment & Music (E&M)	
Original formal	I think they are into drama. I do not understand why.
Reference 0 informal	Drama Queens... Dunno, can't understand it myself.
Reference 1 informal	They into dat drama, not really feelin' it tho.
Reference 2 informal	They like the drama but idk why.
Reference 3 informal	No idea why but they are drama queens.
BERT2BERT$_{en}$ output	i think they are into drama and i don't see why.
Family & Relationships (F&R)	
Original formal	They will think you are leering if you look at them creepily.
Reference 0 informal	if you look creepy and you look..they'll all think you're leering.
Reference 1 informal	if you give them creepy looks they'll think ur leering
Reference 2 informal	they will thing u a creep if u leer at them
Reference 3 informal	dun keep looking at them like that, they'll think your'e leering or somethin'
BERT2BERT$_{en}$ output	If you look at them, they'll think you're leering.

Table 5.8: Samples of sentences from the GYAFC$_{en}$ dataset from both the E&M and the F&R domains, with the corresponding model output. The samples were taken from the test dataset on the *formal to informal* task. [4]

Entertainment & Music (E&M)	
Original formal	Jag tror att de gillar drama. Jag förstår inte varför.
Reference 0 informal	Drama Queens ... Dunno, kan inte förstå det själv.
Reference 1 informal	De bryr sig om datadrama, känner inte riktigt det.
Reference 2 informal	De gillar drama men idk varför.
Reference 3 informal	Ingen aning varför men de är dramadrottningar.
BERT2BERT$_{swe}$ output	Jag tror att de gillar drama, jag vet inte varför.
Family & Relationships (F&R)	
Original formal	De kommer att tro att du lutar om du tittar på dem läskigt.
Reference 0 informal	om du ser läskig ut och du ser ut ... de kommer alla att tro att du drar.
Reference 1 informal	om du ger dem läskiga utseende kommer de att tänka att du läser
Reference 2 informal	de kommer att ge dig en kryp om du lär dig
Reference 3 informal	fortsätt att titta på dem så, de kommer att tänka att du ler eller något
BERT2BERT$_{swe}$ output	om du tittar på dem kommer de att tro att du lutar dig åt dem.

Table 5.9: Samples of sentences from the GYAFC$_{swe}$ dataset from both the E&M and the F&R domains, with the corresponding model output. The samples were taken from the test dataset on the *formal to informal* task. [4]

Chapter 6

Discussion

In this chapter the results of the thesis are discussed in relation to each other, the method, related work and the purpose of the thesis. The chapter is divided into two sections. In the first section the quality of the Swedish corpus is discussed and the second section focuses on the two BERT2BERT models.

6.1 Quality and Evaluation of GYAFC$_{swe}$

The results of this study indicated that after translation, the sentences of the corpus maintained a fairly high quality. The *fluency* and *formality* of the GYAFC$_{swe}$ sentences were mostly preserved in comparison to the sentences in GYAFC$_{en}$, and even though the *meaning* of the sentences tended to change somewhat more during translation, it was still largely preserved.

Even though the formality was mostly preserved, the results also suggested that the parallel informal / formal sentence pairs in the translated GYAFC$_{swe}$ dataset, were more similar in terms of style than the parallel informal / formal sentence pairs in the original GYAFC$_{en}$ dataset. The was because the informal sentences were translated to more formal versions of themselves and the formal sentences were translated to more informal versions, resulting in the parallel informal / formal sentence pairs approaching each other in style.

Some characteristics that divided the informal sentences from their formal version in the GYAFC$_{en}$ dataset, were not possible to translate into equivalent Swedish versions and were therefore also problematic to score. For example the informal English phrase "it's", had a formal version "it is". When translating this phrase to Swedish, both the informal and formal versions read "det är". The English language often contain contractions in informal phrases and the sentence can be made formal by removing the contraction. The same

was true for GYAFC$_{en}$, as many parallel sentence pairs only differed on the use of contractions. This feature was however completely lost during translation to Swedish. By extension, this meant that some GYAFC$_{swe}$ informal / formal sentence pairs were exactly the same sentence or very alike.

These types of language specific characteristics were not regarded when constructing the manual evaluation method for translations. The Swedish phrase "det är" was neither formal nor informal and did therefore not affect the *formality preservation* scoring. This meant that the *formality preservation* score was correct but not necessarily representative when evaluating the conditions on which BERT2BERT$_{swe}$ was trained.

One interesting finding was that not only the *formality preservation* score suggested that the translated sentences approach each other in style. The BLEU and PINC scores indicated the same thing. The original source sentences in GYAFC$_{swe}$ obtained a higher BLEU score than the corresponding sentences in GYAFC$_{en}$ and the reference sentences in GYAFC$_{swe}$ obtained a lower PINC score than the corresponding sentences in GYAFC$_{en}$. These results pointed to a likeness between the source and reference sentences in GYAFC$_{swe}$ that did not exist in GYAFC$_{en}$. This finding indicated that the manual evaluation metric defined in this thesis and the BLEU and PINC scores complied well with each other and reached the same conclusion regarding the translated sentences. The BLEU and PINC scores may even have found that the informal / formal sentence pairs approach each other more, than what the *formality preservation* score indicated.

One important thing to consider, is whether or not the manual scoring of GYAFC$_{swe}$ was affected by the knowledge of what the corpus was to be used for and the current understanding of how the BERT2BERT model functioned. It was not possible to completely exclude these concerns but the risk was deemed low considering that the evaluation was performed before the models were implemented. The fact that the BLEU and PINC scores complied fairly well with the manual evaluation, further supports this assessment. Finally, by dividing the evaluation of the translated corpus into the three criteria *formality preservation*, *fluency preservation* and *meaning preservation*, the risk may have been lowered compared to simply evaluating the sentences based on one criteria such as *quality preservation*.

It would however be interesting and yield more objective results, if an independent group of people with extensive linguistic knowledge were to evaluate the translations with regard to the same three criteria.

It would also be interesting to let the same group of people manually edit the Swedish corpus. The BERT2BERT$_{swe}$ model could then be trained using

a corpus of the same quality as GYAFC$_{en}$.

6.2 Formality Style Transfer (FST)

This study found that the technique of warm-starting an encoder-decoder model using a BERT model was a promising approach. On the *informal to formal* task, BERT2BERT$_{en}$ achieved a BLEU score of 64.50 and 69.35, which both are significantly different from the scores 50.35 and 51.77 obtained by the original source sentences. The same scores were also close to the scores obtained by Rao and Tetreault [4] presented in Table 2.2. The scores did however not reach the standard of more recent research, presented in Table 2.4. These modern models did however, also incorporate other techniques (apart from fine-tuning) in order to boost performance.

It would be interesting to try and combine the models presented in this thesis with for example the concatenation technique presented by Wang et al. [7] and see if this would boost the performance of BERT2BERT$_{en}$ and BERT2BERT$_{swe}$.

Using a larger batch size and larger learning rate during fine-tuning, was also believed to boost the performance of BERT2BERT$_{en}$ and BERT2BERT$_{swe}$. During the tuning of hyper parameters, it was found that a larger batch size not only allowed for faster fine-tuning but also yielded higher BLEU and PINC scores.

Regarding the quality of BERT2BERT$_{swe}$ compared to that of BERT2BERT$_{en}$, the results showed the following.

On the *informal to formal* task, BERT2BERT$_{swe}$ obtained lower BLEU and PINC scores than BERT2BERT$_{en}$ on both domains. On the *formal to informal* task, BERT2BERT$_{swe}$ obtained higher BLEU scores and lower PINC scores than BERT2BERT$_{en}$ on both domains. This outcome was not unexpected, as the quality of GYAFC$_{swe}$ was found to be lower than that of GYAFC$_{en}$. It was then natural that the quality of BERT2BERT$_{en}$ surpassed that of BERT2BERT$_{swe}$, since a model fine-tuned using informal / formal sentence pairs that were close to each other in style, would naturally result in a model that was less able to generate sentences with a specific style, compared to a model trained on a corpus with less similar sentences. Even though the sentences in the test dataset suffered from the same drawback as the training dataset, the model would obtain lower scores than a model with more distinct differences between the informal and formal sentences.

The reason for BERT2BERT$_{swe}$'s high BLEU score on the *formal to informal* task, was most likely the high initial source sentence score. From

the results in Tables 5.4 and 5.5 it was clear that the original source sentences of GYAFC$_{swe}$ had BLEU scores that were almost the same as the final BLEU scores of BERT2BERT$_{en}$. This meant that, as long as BERT2BERT$_{swe}$ improved the sentences at all, the model would receive a higher BLEU score than BERT2BERT$_{en}$.

Despite BERT2BERT$_{swe}$'s difficulty to fine-tune using an imperfect corpus, it did in most cases obtain significantly different scores than those of the original source sentences and the reference sentences. This was an encouraging finding, as it implied that BERT2BERT$_{swe}$ had the capability to perform FST and probably the potential to match the quality of BERT2BERT$_{en}$ under the right circumstances.

One interesting but unanticipated finding of this study was whether or not a corpus could be automatically translated and then used to train models in that language. While the technique showed some promise and deserve further investigation it was clear that formality is not necessarily something that transfers well between languages. Contractions for example, occur a lot in English informal sentences but can not be translated into equivalent Swedish translations. Machine translation could however be used as a tool to speed up the creation of corpus in languages other than English. A lot of the sentences in GYAFC$_{swe}$ were perfect translations and even though some of the sentences were found wanting, GYAFC$_{swe}$ was successfully used to fine-tune a model which could transform sentences into a different level of formality.

Chapter 7

Conclusions and Future Work

In this chapter the conclusions of the thesis project are presented. The chapter also addresses limitations of the project and suggestions for future work.

7.1 Conclusions

This goal of this thesis project was to investigate how the quality of a model trained for FST in Swedish compared to an equivalent model trained for FST in English. The aim was to leverage the BERT model, since pre-trained versions of the model already existed for both languages.

Using BERT to warm-start an encoder-decoder model proved to be a promising method, as the English version obtained results matching those of previous studies. The Swedish model did not reach the same standard as the English model, but was nevertheless found to obtain significantly different scores compared to the scores of the original source sentences used as input.

The reason for the inferior results of the Swedish model was most likely the inferior quality of the corpus used for fine-tuning of that model. The corpus used for fine-tuning the Swedish model, was created through machine translating the manually created and corrected corpus used for fine-tuning the English model. A manual evaluation of a subset of the Swedish corpus confirmed that the corpus retained a fairly high quality but that the difference between formal and informal versions of sentences had been reduced during translation.

These results indicated that the Swedish model had the capacity to match the English model, if given the same initial conditions.

The study indicated that the technique of using a translated corpus for fine-tuning, was not optimal but not without benefits either. It was clear that

formality is not necessarily something that transfers well between languages and for this reason, the technique should not be used as the sole tool when creating corpus for the task of FST. The technique was however deemed appropriate to use as an aid to speed up the creation of corpus, which could later be manually corrected.

These results proved that models designed with the English language in mind, could reliably be used for the Swedish language. For Hejare AB, this meant that the company could create a FST tool using models originally designed for the English language. The two models implemented in this thesis could potentially be used but with caution. The users needed to be aware of the fact that the sentences generated by the models sometimes needed manual correction.

Finally, the study also highlighted the need for creating large task specific corpus in Swedish.

7.2 Limitations

A limitation of this study was that the translated corpus was manually evaluated by the author. Even though Swedish was the authors mother tongue and the author was also fluent in English, the author did not posses the extensive linguistic knowledge that would have been preferable when evaluating the translations. Despite of this, the scores were deemed reliable, since they were in agreement with the BLEU and PINC scores.

7.3 Future work

A natural progress of this work is to correct the Swedish corpus and introduce more distinct differences between the informal and formal sentences. Using this corpus, the Swedish model could be fine-tuned on the same premises as the English model and the conclusions drawn in this thesis could potentially be strengthened.

It would also be interesting to manually check both the sentences generated by the two current models and the sentences in their two corpus. This would be done in order to gain a further understanding into how the text degradation introduced during translation, affected the generated sentences. By understanding what aspects of the Swedish corpus that decreased the quality of the Swedish model, the Swedish corpus could be corrected with a better understanding into what needed to be changed. The Swedish model

could also potentially be re-implemented with this understanding of the corpus in mind. Moreover, this information could help determine if the models are fit to be used in the commercial market.

In order to confidently use the models, it would also be beneficial to conduct a study were users are allowed to test the models and then evaluate their experience. The trust to NLP tools can be high amongst users and therefore the models should be evaluated with regard to the user.

Future work could also improve the models proposed in this thesis by increasing the batch size during fine-tuning or utilising the BERT_{LARGE} model for warm-starting the encoder-decoder model. In order to utilise the BERT_{LARGE} model, the model would however have to be pre-trained using text in Swedish. It is also conceivable that the models could be improved by incorporating rules for FST as proposed by Xu et al. [6] and Wang et al. [7].

Further studies could also assess the possibility of using other models for the task of FST in Swedish. Given enough resources, the GPT-CAT model introduced by Wang et al. [7] could for example be pre-trained and fine-tuned for Swedish use. Alternatively, a multilingual model could be tested for the task. The recently introduced mT5 model [54] has for example shown great promise.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] F. A. Sheikha and D. Inkpen, "Generation of Formal and Informal Sentences," in *Proceedings of the 13th European Workshop on Natural Language Generation*, 2011, pp. 187–193.

[4] S. Rao and J. R. Tetreault, "Dear Sir or Madam, May I Introduce the GYAFC Dataset: Corpus, Benchmarks and Metrics for Formality Style Transfer," *CoRR*, vol. abs/1803.06535, 2018. [Online]. Available: http://arxiv.org/abs/1803.06535

[5] X. Niu, S. Rao, and M. Carpuat, "Multi-Task Neural Models for Translating Between Styles Within and Across Languages," *arXiv preprint arXiv:1806.04357*, 2018.

[6] R. Xu, T. Ge, and F. Wei, "Formality Style Transfer with Hybrid Textual Annotations," *arXiv preprint arXiv:1903.06353*, 2019.

[7] Y. Wang, Y. Wu, L. Mou, Z. Li, and W. Chao, "Harnessing Pre-Trained Neural Networks with Rules for Formality Style Transfer," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3564–3569.

[8] M. Malmsten, L. Börjeson, and C. Haffenden, "Playing with Words at the National Library of Sweden–Making a Swedish BERT," *arXiv preprint arXiv:2007.01658*, 2020.

[9] "Hejare AB," URL: https://www.hejare.se/, Last accessed 28 January 2021.

[10] "Momang," URL: https://momang.io/, Last accessed 28 January 2021.

[11] F. Heylighen and J.-M. Dewaele, "Formality of Language: definition, measurement and behavioral determinants," *Interner Bericht, Center "Leo Apostel", Vrije Universiteit Brüssel*, vol. 4, 1999.

[12] E. Hovy, "Generating Natural Language under Pragmatic Constraints," *Journal of Pragmatics*, vol. 11, no. 6, pp. 689–719, 1987.

[13] E. Pavlick and J. Tetreault, "An Empirical Analysis of Formality in Online Communication," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 61–74, 2016.

[14] W. Xu, A. Ritter, W. B. Dolan, R. Grishman, and C. Cherry, "Paraphrasing for Style," in *Proceedings of COLING 2012*, 2012, pp. 2899–2914.

[15] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, "Toward Controlled Generation of Text," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1587–1596.

[16] "Yahoo Datasets," URL: https://webscope.sandbox.yahoo.com/catalog.php?datatype=l, Last accessed 21 February 2021.

[17] H. Jhamtani, V. Gangal, E. Hovy, and E. Nyberg, "Shakespearizing Modern Language Using Copy-Enriched Sequence-to-Sequence Models," *arXiv preprint arXiv:1707.01161*, 2017.

[18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[19] "Yahoo! Answers," URL: https://answers.yahoo.com/answer, Last accessed 21 February 2021.

[20] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[21] D. Chen and W. B. Dolan, "Collecting Highly Parallel Data for Paraphrase Evaluation," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 190–200.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.

[23] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," *arXiv preprint arXiv:1609.08144*, 2016.

[24] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing," *arXiv preprint arXiv:1808.06226*, 2018.

[25] F. Rosenblatt, "Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms," Cornell Aeronautical Lab Inc Buffalo NY, Tech. Rep., 1961.

[26] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[27] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *arXiv preprint arXiv:1409.3215*, 2014.

[29] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv preprint arXiv:1406.1078*, 2014.

[30] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv preprint arXiv:1409.0473*, 2014.

[31] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[32] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding with Unsupervised Learning," *Technical report, OpenAI*, 2018.

[33] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 19–27.

[34] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is Multilingual BERT?" *arXiv preprint arXiv:1906.01502*, 2019.

[35] S. Wu and M. Dredze, "Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT," *arXiv preprint arXiv:1904.09077*, 2019.

[36] S. Rothe, S. Narayan, and A. Severyn, "Leveraging Pre-trained Checkpoints for Sequence Generation Tasks," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 264–280, 2020.

[37] "Google Scholar," URL: https://scholar.google.com/, Last accessed 28 April 2021.

[38] "Google Cloud," URL: https://cloud.google.com, Last accessed 28 January 2021.

[39] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *arXiv preprint arXiv:1909.11942*, 2019.

[40] "Kb/bert-base-swedish-cased, Hugginface Model," URL: https://huggingface.co/KB/bert-base-swedish-cased, Last accessed 16 April 2021.

[41] "Python," URL: https://www.python.org/, Last accessed 28 January 2021.

[42] "Transformers, Hugginface Library," URL: https://huggingface.co/transformers/, Last accessed 16 April 2021.

[43] "EncoderDecoderModel, Hugginface Library," URL: https://huggingface.co/transformers/model_doc/encoderdecoder.html#encoderdecodermodel, Last accessed 16 April 2021.

[44] "bert-base-cased, Hugginface Model," URL: https://huggingface.co/bert-base-cased, Last accessed 16 April 2021.

[45] "BertTokenizer, Hugginface Library," URL: https://huggingface.co/transformers/model_doc/bert.html#berttokenizer, Last accessed 16 April 2021.

[46] "Seq2SeqTrainer, Hugginface Library," URL: https://huggingface.co/transformers/main_classes/trainer.html#seq2seqtrainer, Last accessed 16 April 2021.

[47] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[48] "Tensorflow," URL: https://www.tensorflow.org/, Last accessed 28 January 2021.

[49] "PyTorch," URL: https://pytorch.org/, Last accessed 28 January 2021.

[50] "Apex (A PyTorch Extension), NVIDIA," URL: https://nvidia.github.io/apex/, Last accessed 16 April 2021.

[51] "Debain," URL: https://www.debian.org/, Last accessed 16 April 2021.

[52] "Deep Learning VM, Google Cloud," URL: https://cloud.google.com/deep-learning-vm, Last accessed 16 April 2021.

[53] "NVIDIA Tesla T4 GPU," URL: https://www.nvidia.com/en-us/data-center/tesla-t4/, Last accessed 16 April 2021.

[54] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[55] "GYAFC Github," URL: https://github.com/raosudha89/GYAFC-corpus, Last accessed 28 April 2021.

Appendix A

Gain Access to the GYAFC$_{swe}$ Dataset

The GYAFC$_{swe}$ dataset was created from the GYAFC dataset which was constructed using the *Yahoo Answers corpus: L6 - Yahoo! Answers Comprehensive Questions and Answers version 1.0* [16]. In order to gain access to the GYAFC$_{swe}$ dataset, one must first gain access to the *Yahoo Answers corpus* and then the GYAFC dataset. Both the *Yahoo Answers corpus* and the GYAFC dataset can be accessed free of charge for research purposes by following the instruction provided on the GYAFC github page [55]. Once access has been given to both the L6 corpus and the GYAFC corpus, both approvals can be emailed to Maria Lindblad (mali4@kth.se) along with a description of how the data will be used, and access will be provided to the GYAFC$_{swe}$ corpus.

For DIVA

{
"Author1": {
"Last name": "Lindblad",
"First name": "Maria",
"Local User Id": "u10k7p9q",
"E-mail": "mali4@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science ",
}
},
"Degree": {"Educational program": "Master's Programme, Computer Science, 120 credits"},
"Title": {
"Main title": "A Comparative Study of the Quality between Formality Style Transfer of Sentences in Swedish and English, leveraging the BERT model",
"Language": "eng" },
"Alternative title": {
"Main title": "En jämförande studie av kvaliteten mellan överföring av formalitetsstil på svenska och engelska meningar, med hjälp av BERT-modellen",
"Language": "swe"
},
"Supervisor1": {
"Last name": "Sidorova",
"First name": "Yulia",
"Local User Id": "u19ku6t3",
"E-mail": "yuliasid@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science ",
"L2": "Computer Science" }
},
"Supervisor2": {
"Last name": "Svensson",
"First name": "Henrik",
"E-mail": "henrik@hejare.se",
"Other organisation": "Hejare AB"}
},
"Examiner1": {
"Last name": "Kann",
"First name": "Viggo",
"Local User Id": "u1kd2mni",
"E-mail": "viggo@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science ",
"L2": "Computer Science" }
},
"Cooperation": { "Partner_name": "Hejare AB"},
"Other information": {
"Year": "2021", "Number of pages": "xv,73"}
}