Numerical Commonsense Reasoning Across Languages

Dayyán O'Brien

MInf Project (Part 1) Report
Master of Informatics
School of Informatics
University of Edinburgh

2023

Abstract

Recent work has shown that pre-trained language models (PLTMs) do not exhibit numerical commonsense in English (e.g a car has *four* wheels). In this project, we introduce a new multilingual numerical reasoning task, MNUMERSENSE, which contains 22k probes in Chinese (9.4k probes), Russian (9.1k probes), and Arabic (3.8k probes).

We find that, while finetuning improves results, poor performance occurs across mBERT, xlm-RoBERTa, mT5, mBART, and mGPT. A thorough exploration of this performance finds models occasionally struggle to attend to parts of a sequence necessary for reasoning and tend to predict numbers that exist only within a small subset of the possible predictions. We find that cross-lingual learning can occur and that given enough training samples, models learn plural forms in numeric reasoning. Finally, we explore linguistic-specific phenomena in each of our languages. Specifically, we look at Russian case declension, Arabic declension, and Chinese word similarity in numerical reasoning.

Research Ethics Approval

This project obtained approval from the Informatics Research Ethics committee.
Ethics application number: 6800
Date when approval was obtained: 2022-06-17
The participants' information sheet and a consent form are included in the appendix.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Dayyán O'Brien*)

Acknowledgements

I want to thank my supervisor, Prof. Mirella Lapata for all their help, support, and guidance throughout the project. I am grateful for all of the time and effort she has dedicated in order to ensure that I succeed in this project. I also want to thank Tom Sherborne for his guidance in collecting data and structuring the project.

To my dear friends and family. Thank you for your support.

Above all, I want to thank God. The All-Powerful, the All-Wise, and the All-Glorious.

Table of Contents

Chapter 1

Introduction

This section provides a brief overview of our project, including the motivation, objectives of the project, and our contributions. Finally, we provide the outline of the report.

1.1 Motivation

The use of natural language systems is growing throughout the world today and commonsense reasoning is an area of development we must work on in order to ensure reliable, interpretable, and trustworthy systems. In this project, we look at numerical reasoning across languages and aim to evaluate and contrast how different languages perform on these and if they take advantage of linguistic-specific phenomena. Numerical reasoning is using commonsense inferences about numbers. For example, a dog has *two* eyes and a car has *four* wheels.

There currently does not exist a dataset that addresses numerical reasoning across multiple languages, and this project fills this gap. To create this dataset, we crowdsourced translation based on an English numerical reasoning dataset called NUMERSENSE by Lin et al. (2020). This consists of Arabic, Russian, and Chinese and provides a valuable resource for researchers working on commonsense and multilinguality in natural language processing (NLP). More multilingual datasets are important as they democratize NLP away from English and improve access to NLP systems and research for those who don't speak the language.

Using this dataset, we perform experiments on mBERT, xlm-RoBERTa, mT5, mBART, and mGPT. Our experiments reveal that models struggle to interpret numerical reasoning, even when finetuned. They often get stuck predicting only a small subset of numbers, almost randomly, and struggle to attend to important parts of a sentence. We also find severe problems with object-bias, where models consistently predict the same number regardless of the subject noun. These reasons motivate the need to improve state-of-the-art PTLMs in numerical reasoning. We look at low-resource (few samples) training in Arabic, which is important as there are many existing languages without many samples and we need models that can learn well under these constraints. We see that while

performance is worse than other languages, it still achieves close performance. We also look at weather models can take advantage of plurality, we find that PTLMs can take advantage of plurality in Russian after finetuning but fail to learn this reasoning in Arabic.

We also perform experiments on cross-linguality, seeing if representations of numeric reasoning are language-specific or if they can be learned across different languages. These results were typically worse than solely monolingual results. However, they provide promising results that our models can leverage knowledge learned from one language to improve performance in another. We additionally contrast our experiment with machine translation (MT) systems, finding poorer results than our new dataset. Though MT still roughly corresponds to crowdsourced performance and represents the initial issues we analyzed.

Finally, we explore language-specific phenomena. These experiments are crucial for developing models that can handle the nuances and differences of languages. We look at declension in both Russian and Arabic, finding that pre-trained models severely struggle with predicting both the correct number and its declension type. We also see that Russian struggles with instrumentals but performs strongly when no declension is required. Finally, we look at word similarity in Chinese. We find that models don't attend much on number units, and typically predict the same regardless.

1.2 Objectives

- Creating a multilingual dataset for numerical reasoning.
- Exploring and analyzing the performance of different state-of-the-art PTLMs on multilingual reasoning.
- Exploring if cross-lingual learning can take place from one language to another.
- Seeing if models take advantage of linguistic phenomena across and specific to languages in relation to numerical reasoning.

1.3 Contributions

The contributions to this project are as follows:

- Created a numerical reasoning dataset with over 22k examples.
- Evaluated and investigated performance on mBERT, xlm-roBERTa, mBART, mT5 and mGPT.
- Analysed the impact of plurality on model performance.
- Performed language-specific experiments on case declension in Russian, declension in Arabic, and Chinese word reliance.

1.4 Report outline

The report is structured as follows:

- Chapter 2 provides the background required for this project. It covers our dataset, the language models which we use, and related work to our project.
- Chapter 3 describes how we collect our data, crowdsourcing metrics, and the experiments we performed to ensure that the data is of high quality.
- Chapter 4 covers how we set up our models for experiments, describing their preprocessing, finetuning, and inference.
- Chapter 5 provides a brief description and motivation along with the result and analysis for all the experiments we perform.
- Finally, we conclude the report in Chapter 6. This includes our conclusion, a results overview, and potential future work.

Chapter 2

Background and Literature Review

This chapter describes the background information and a literature review of related work to this project. We explain language models before looking at numerical reasoning in relation to our task. Finally, we look at related work on commonsense reasoning and probing language models.

2.1 Language models

In this section, we discuss the language models for commonsense. This section introduces the Transformer and looks at how its architecture is applied to BERT, RoBERTa, BART, T5, and GPT-3. We then see how these models function on multilingual models.

2.1.1 Transformers

The basic Transformer (Vaswani et al., 2017) is a deep learning model that uses self-attention. This is where we encode our input in some way that allows us to better represent our data for learning, then we can decode it to generate an output. The architecture of the Transformer can be seen in Figure 2.1, where the left half is the encoder and the right is the decoder. Self-attention is an attention mechanism where we use the positional encoding of a sequence in order to find some representation of it. To add this positional encoding we use a sinusoidal function based on the position and model dimensionality. For example, if we looked at the sentence, "A dog has two eyes and four legs", we would recognize that *two* is dependent on *dog* and *eyes*. Attention is a technique that aims to select parts of an embedding to pay attention to.

Gated recurrent neural networks (GRUs) (Cho et al., 2014) process some nth token based on its input, $n-1$. Theoretically, this should lead to some input being able to suitably propagate throughout a network. As we increase our network size we fall at risk to the Vanishing Gradient Problem (VGP) (Pascanu et al., 2013). This means any representation of the input disappears and no learning can take place. Attention, however, allows us to access any previous state in the sequence. In the simplest form, this would be the weighted average of the inputs.

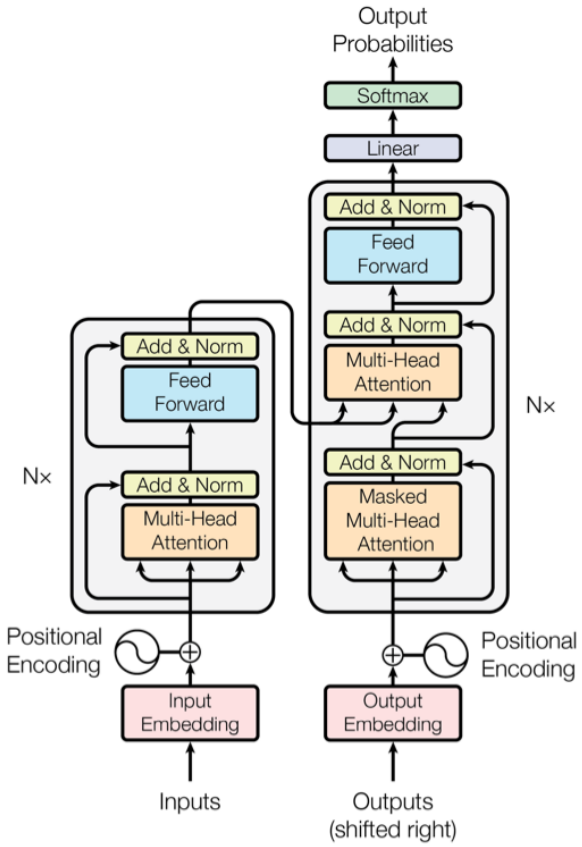


Figure 2.1: The architecture of the Transformer - figure taken from (Vaswani et al., 2017).

$$\mathbf{x}_{pooled} = \sum_{t=1}^{T} a(\mathbf{e}^{(t)})\mathbf{e}^{(t)}, \mathbf{e}^{(t)} = embedding(\mathbf{x}^{(t)}; V) \tag{2.1}$$

The particular attention we look at is "Scaled Dot-Product Attention". The input is made up of queries and keys of size d_k and the values at dimension d_v. In practice, we look at a set of queries (Q) simultaneously, with its set of keys (K) and values (V). Figure 2.2 shows us an example of how attention on English to French translation is performed.

$$Attention(Q,\ K,\ V) = softmax(\frac{QK^T}{\sqrt{d_k}}V) \tag{2.2}$$

This only performs a single attention function. Multi-head attention is the practice of linearly projecting the keys, queries, and values h times, each with different projections learned in parallel yielding d_v dimensional outputs. These are then concatenated and then projected. This allows us to attend to multiple parts of a sequence independently.

We also investigate some of our results through attention distribution visualization. We get the attention scores for all the heads in relation to some *number word*. This allows us to see what parts of a sentence the model finds important in relation to the number.

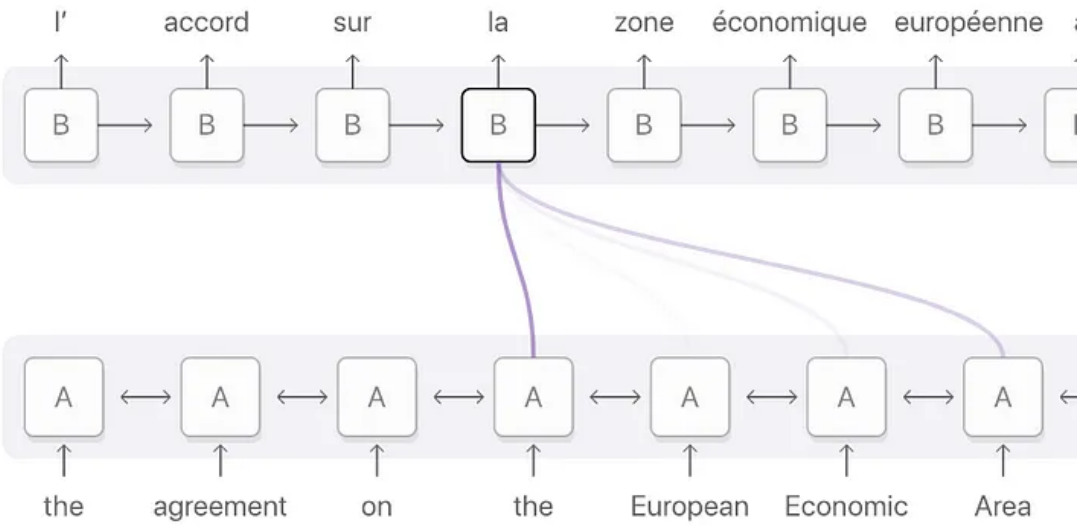


Figure 2.2: Attention on the translation from English (key) to the French word 'la' (query). Darker lines indicate stronger attention. Figure taken from (Olah & Carter, 2016)

2.1.2 BERT & RoBERTa

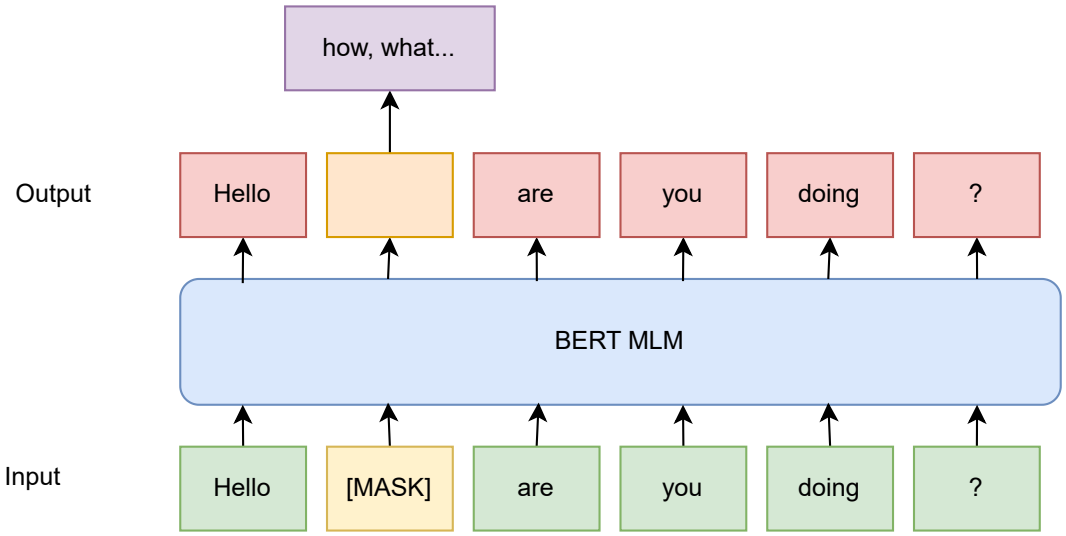


Figure 2.3: BERT/RoBERTa's masked word filling objective.

Based on the Transformer architecture, BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) is a language model which has been achieving state-of-the-art results on a variety of NLP problems. BERT reads the entire sentence at once. This has been shown to give a better understanding of the text as you can see dependencies on all the surroundings (i.e. any words in that sequence) of a word. It achieves this by masking a random word in a sentence and attempting to predict the masked word, as seen in Figure 2.3. It has a second learning objective of next-sentence prediction, however, we do not need to use this in this project. Models such as BERT are usually pre-trained on an unlabelled, plain text corpus, allowing it to have some

understanding of the language it's trained on. It can then be fine-tuned for a specific task. Its architecture can be seen as the left half (encoder) of the Transformer in Figure 2.1.

RoBERTa (Liu et al., 2019) (Robustly Optimized BERT Pretraining Approach) was then introduced as BERT was found to be significantly under-trained. This follows a similar architecture to BERT, but instead trains for longer and does not have next sentence prediction as a learning objective. It also uses dynamic masking, which ensures that the same sentence is masked at different positions over each epoch.

2.1.3 GPT-3

GPT-3 (Generative-Pre-trained Transformer 3) (Brown et al., 2020) is an autoregressive model, meaning it only processes input from left to right. This means it cannot take advantage of BERTs bidirectional learning, however, it can be trained on a much larger dataset. Autoregressive LMs have been shown to perform well as few-shot learners, which is where a learner can be trained on a task with just a few examples. The architecture of GPT is the right half (decoder) of Figure 2.1. The decoder means that GPT is able to generate text, unlike BERT and RoBERTa.

2.1.4 BART

BART (Bidirectional and Auto-Regressive Transformer) (Lewis et al., 2019) is a sequence-to-sequence model that is pre-trained by corrupting text with some arbitrary noise function and learning how to recreate this text. Pretrained objectives also include text infilling, which masks a span of text and aims to predict it from a single *<mask>* token. Sentences are shuffled in a random order based on full stops. The model uses a bidirectional encoder, similar to BERT, and uses an autoregressive decoder, similar to GPT (meaning it can generate text). Such an architecture can be seen in Figure 2.4.

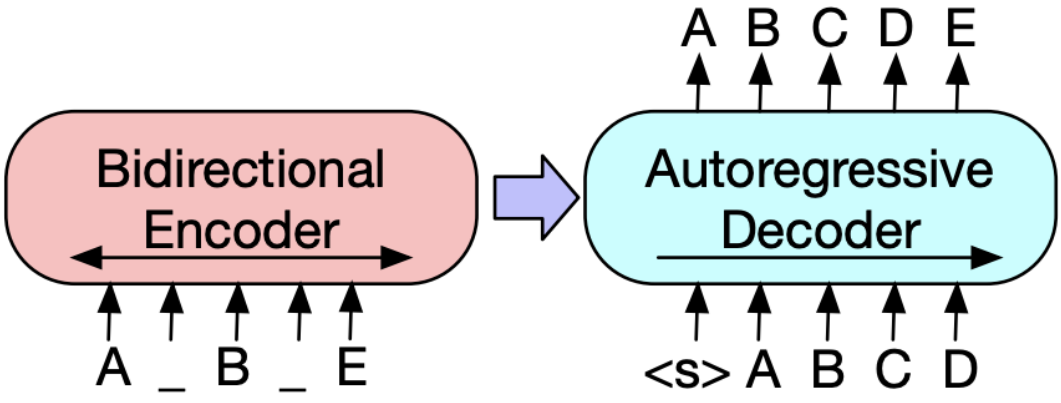


Figure 2.4: The architecture of BART. Figure taken from (Lewis et al., 2019)

2.1.5 T5

T5 (Text-to-Text Transformer Transformer) (Raffel et al., 2019) is based on the vanilla encoder-decoder transformer, with a pre-trained unsupervised objective of randomly dropping out words and replacing them with a sentinel token. This is a token that spans any length and is then trained to predict the gap. This can be seen in Figure 2.5. T5 is trained on some supervised learning objectives, for example, you can prepend the prefix 'summarize:' to make the model summarise the succeeding paragraph.

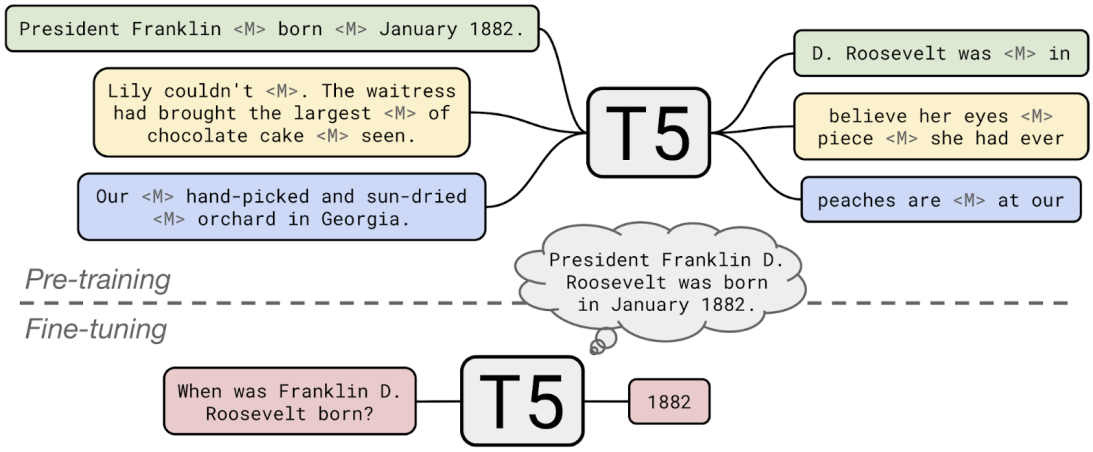


Figure 2.5: Pretraining and finetuning of T5, sentinal tokens are indicated by <M>. Figure taken from (Raffel et al., 2019)

2.1.6 Multilingual models

The models which we have discussed all have multilingual variants, allowing them to perform on multiple languages. All of these models have been pre-trained on English, Arabic, Russian, and Chinese.

mBERT (Devlin et al., 2018) is trained on 102 languages, which were chosen based on the top 100 largest Wikipedias. XLM-RoBERTa (Conneau et al., 2019) pre-trained on CommonCrawl Corpus instead of Wikipedia as CommonCrawl is significantly larger. The model was trained on cross-lingual masked language modeling objectives and has been found to perform significantly better than mBERT on a variety of cross-lingual benchmarks. mGPT (Shliazhko et al., 2022) is based on the GPT-3 architecture using GPT-2 sources and is trained on 60 languages using Wikipedia and the Colossal Clean Crawled Corpus (C4). mBART (Liu et al., 2020) trained on large mono-lingual corpora using the same objective as BART. This can then be fine-tuned to a specific task. mT5 (Xue et al., 2020) trained on C4. Unlike T5, it is not pre-trained on any downstream tasks.

2.2 NumerSense

Commonsense knowledge consists of facts about the world that are considered widely known. Numerical commonsense knowledge is a type of knowledge that we can use to understand a numerical relation between entities. NUMERSENSE (Lin et al., 2020) is a numerical reasoning dataset where the goal is to guess a number between zero and ten in a masked sentence, examples of this task is found in Figure 2.6. This is the task that we will be translating and performing experiments on. These tasks come from a variety of categories such as objects, maths, and geography. A full list of these categories, with examples can be found in Table 2.1.

Category	Example
Objects	A car has four wheels.
Biology	People have two lungs.
Geography	The UK is made up of four countries.
Maths	Two plus two is four.
Physics	1G is ten meters per second, per second.
Unit	There are three meals in a day.
Geometry	A triangle has three sides.
Misc.	There are no princes in the United States.

Table 2.1: Example sentences for each category in NUMERSENSE

Recent research has shown the pre-trained language models PTLMs may possess the commonsense necessary for this. Lin et al. (2020) reported that BERT and RoBERTa, even when fine-tuned, perform poorly on this dataset. The full results for these are found in Table 2.2.

	Accuracy
pretrained	
bert-base	32.0
bert-large	37.6
xlm-roberta-base	36.0
xlm-roberta-large	45.9
gpt-2	29.9
finetuned	
bert-large	50.0
xlm-roberta-large	54.0
human bound	$89.7^{(\alpha)}/96.3^{(\beta)}$

Table 2.2: Results from (Lin et al., 2020). α = no external information, β = Wikipedia is allowed.

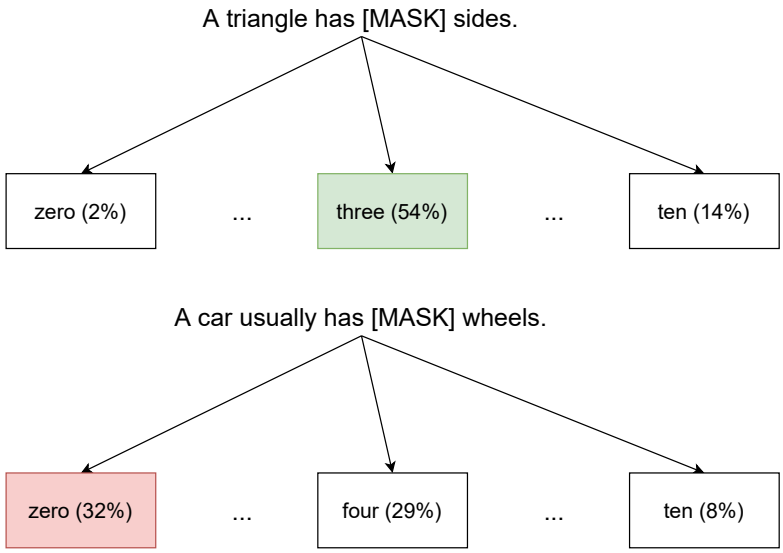


Figure 2.6: Coloured boxes indicate the chosen number, where green means the prediction is true to life. The percentage indicates example probabilities attributed by to that guess, with the maximum being selected.

2.3 Related work

In this section, we look at the related work of our project. We discuss the potential of PTLMs to understand or encapsulate common sense. We then look at previous work on numerical reasoning, which covers some existing tasks and attempts to see if PTLMs encode numbers properly. We then look at existing multilingual commonsense tasks and cross-lingual learning. Finally, we look at previous attempts at probing PTLMs.

2.3.1 Commonsense in PTLMs

Prior work has been done that shows PTLMs may possess commonsense knowledge. Petroni et al. (2019) argued the PTLMs could store relational knowledge from the training data and act as knowledge bases. This is potentially advantageous as you have to do no extra work to insert knowledge into these models. They found BERT performed well in retrieving factual information and relations.

Additionally, Bouraoui et al. (2019) argued that BERT captured relational knowledge beyond its word embeddings. They look at these relations in a variety of domains, including commonsense. Such behavior was not replicated in NUMERSENSE, and it was found that LMs do not possess numerical commonsense in English. We aim to look at this same problem in a cross-lingual context.

2.3.2 Numerical commonsense

There have been a number of studies that have explored numerical reasoning. Forbes & Choi (2017) and Goel et al. (2019) have both looked at comparison problems (e.g. a stone is heavier than a feather) in PTLMs. (Goel et al., 2019) found that BERT

performed well in comparison tasks. Wallace et al. (2019) examines how NLP models embed numbers (the same way as text). They explore if a number can be decoded from the word embedding (e.g. "71" -> 71.0. They find that BERT struggles to understand numbers in a large range [1,1000], but generally performs well in a small range [1, 99]. They conclude that this is likely due to sub-word pieces not being suitable for encoding numbers.

NUMBERGAME is another numerical reasoning task proposed by Mishra et al. (2020) that evaluates numerical reasoning across eight formats. These formats are called missing numerical knowledge, maths in other domains, quantitative comparison, completion type, reading comprehension with explicit math, reading comprehension with implicit maths, quantitative natural language inference, and arithmetic word problems. In particular, Mishra et al. (2020) found that model performance was extremely poor when numerical knowledge beyond the sentence was required.

2.3.3 Multilinguality

Mikolov et al. (2013) explore the embeddings of words across languages. They find that similar words in different languages are embedded similarly, and a linear mapping is all that's required to learn this equivalency. This implies promise in cross-lingual learning and the potential benefit of combining knowledge from multiple languages.

There have been a number of multilingual benchmarks, such as TYDI (Clark et al., 2020), which is a QA benchmark. There is also XTREME (Hu et al., 2020) and XGLUE (Liang et al., 2020), which are both multi-task multilingual benchmarks. However, these do not measure commonsense reasoning. XCOPA (Ponti et al., 2020) evaluates multilingual causal commonsense reasoning. X-CSR (Lin et al., 2021), introduces two new datasets, X-CSQA and X-CODAH. These evaluate QA tasks and the ability to complete the most plausible sentences respectively. There are no existing datasets for multilingual numerical commonsense and this project will serve as that extension.

2.3.4 Probing

Probing tasks have been performed on PTLMs that analyze linguistic phenomena. Clark et al. (2019) explore the relationship between attention weights and linguistic syntax. They look at BERT and find that attention heads match linguistic syntax and co-reference.

Additionally, Tenney et al. (2019) explore BERT and finds that its layers discover the classic NLP pipeline (POS tagging, parsing, NER, etc.) In particular, they see that the lower layers represent syntactic connections and the higher layers represent complex semantic relations. However, Niu et al. (2022) argue that, while BERT does replicate linguistically founded relations, their representation is more nuanced than being divided into layers. Similar to Lin et al. (2020), we can see how attention behaves in Figure 2.7. The peaks represent attention to that word in the said layer, we see that *chicken* and *road* are attended to in the middle layers. Implying some connection between those words and *crossed*.

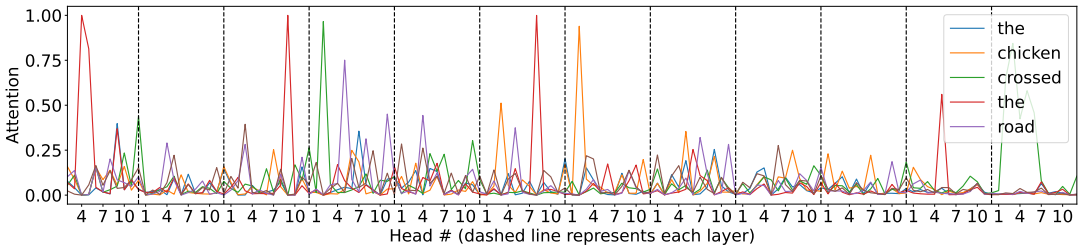


Figure 2.7: Attention distribution of "*crossed*" from "*the chicken crossed the road*" on mBERT.

In addition to the work described in Section 2.3.1, some work has been done probing PTLMs for commonsense. Talmor et al. (2019) find that different PTLMs have different reasoning abilities (i.e RoBERTa is able to perform reasoning where BERT is not). For example, RoBERTa can compare numbers, even in a zero-shot setting while BERT cannot. However, these models are generally not able to deal with common sense abstractly. For example, RoBERTa can perform comparison tasks of ages, but if these ages are much older than the average human, it quickly breaks.

Zhou et al. (2019) evaluates a variety of PTLMs on seven benchmarks. These benchmarks are Conjunction Acceptability (Zhou et al., 2019), Winograd Schema Challenge (Levesque et al., 2012), Sense Making (Wang et al., 2019), Sense Making with Reasoning (Wang et al., 2019), Situations With Adversarial Generations (SWAG) (Zellers et al., 2018), HellaSWAG (Zellers et al., 2019) and Argument Reasoning Comprehension Task (Habernal et al., 2018). They find that, while language models are able to perform simple reasoning tasks (e.g He didn't get sleep therefore he is tired), they fail for problems that require multiple inference steps (e.g I asked him to clean my car and the room, and he only cleaned the car. So I won't pay him).

We will implement probing tasks in our analysis, including an analysis of attention and object-bias experiments.

Chapter 3

Data Collection

This chapter describes our data collection efforts for this project. We first discuss the languages we chose, and why. Then, we discuss how we collected our data, its quality, and its limitations.

3.1 Language choice

There were a number of potential and interesting languages for our task, and the primary consideration for these was plurality. They behave in interesting ways when working with numbers. Arabic nouns can be either singular, dual, or plural, which may give the models more clues in classifying numbers. You can see examples of this below, with the different attachments to طالب.

```
english => arabic
one student => طالب واحد
two students => طالبتان
three students => ثلاثة طلاب
```

Russian plurals are split into three groups; singular, 2-4, and 0 & 5+. Which is visible in the attachments to студент.

```
english => russian
one student => один студент
two students => два студента
five students => пять студентов
```

This could prove an interesting contrast to Arabic. Both Arabic and Russian have a rich morphology with their numbers, unlike English. Chinese, on the other hand, does not have plural forms. Numbers generally have one form, however, there are a few exceptions, such as having two forms for 'two', one for counting (二) and the other for finance (贰). All these languages have different alphabets, which should make for interesting analysis in cross-lingual experiments.

3.2 Data collection

In this section, we discuss data collection through machine translation and crowdsourcing. Crowdsourcing is the practice of getting a group of people to produce data. We first look at machine translation. Then, we look at crowdsourcing metrics and Amazon MTurk. We conclude that its quality in pilot studies is poor. Then, we perform data collection in-house and find a significant increase in data quality. Finally, we discuss the limitations of our translations.

3.2.1 Machine Translation

As a baseline, we translated our dataset using Google's Cloud Translation API. This should give a soft lower bound on our performance as well and help in evaluating the quality of translations as discussed in the succeeding subsections. As these use machine translation (MT), we weren't able to specify rules for formatting, such as ensuring numbers are in their written form and bracketed. This can be seen in Table 3.1. Due to the untrustworthiness of MT and its poor results in maintaining bracketing, we instead try collecting our data through crowdsourcing.

	Success rate
Russian	0.88
Chinese	0.76
Arabic	0.73

Table 3.1: Success rate of well-formatted sentences versus all sentences (n = 9648) for each language using Google Translate. The success rate is the % of sentences that have been translated while maintaining the correct formatting.

3.2.2 Crowdsourcing metrics

BLEU

BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002) is a metric ranging from zero to one, in which you can evaluate how similar two strings are. It's typically used for evaluating MT quality however it is also useful in crowdsourcing. BLEU allows us to see how close our crowd-sourced translation is to MT. If someone gets consistently high BLEU scores over this pair, then they are likely using machine translation software.

It is calculated as follows (for some n-gram):

$$\text{BLEU} = \min(1, \frac{\text{output-length}}{\text{reference-length}})(\prod_{i=1}^{n} \text{precision}_i)^{\frac{1}{n}} \tag{3.1}$$

TER

TER (Translation Edit Rate) (Snover et al., 2006) is a measurement of the number of changes required to one string in order to recreate the other. Similar to BLEU, this can be used to evaluate how different a machine-translation/crowdsourced pair is. Snover et al. (2006) found that TER has higher correlations with human judgments. Edits are the number of changes (insertion, substitution, deletion) needed to get back to the original sentence.

$$\text{TER} = \frac{\text{\# of edits}}{\text{average \# of reference words}} \tag{3.2}$$

3.2.3 MTurk

Amazon Mechanical Turk (MTurk) is a crowdsourcing platform in which 'workers' from across the world can partake in jobs called Human Intelligence Tasks (HITs). Crowdsourcing is an effective means of creating a high-quality multilingual dataset versus machine translation (Behnke et al., 2018). MTurk is popular in data collection due to the ease of finding workers and the low cost to collect data. Additionally, its quality is generally high when compared to professionals (Zaidan & Callison-Burch, 2011).

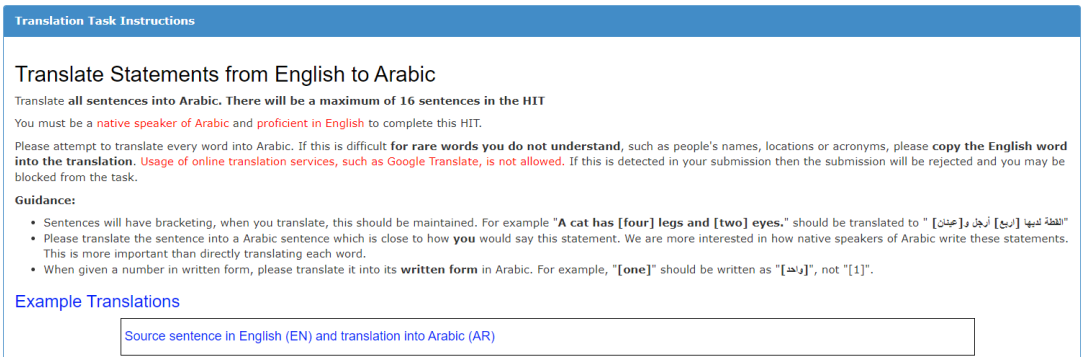
There are a number of ethical concerns about using this platform. It's argued that the platform can be demeaning to workers, as results appear like 'magic' (Schuster, 2014). It's the only option for many of those in poverty (Semuels, 2018), however, the median salary is ∼$2/hr (Hara et al., 2017). We also see a difference in salary on gender and country of residence (Hara et al., 2019). Tasks are paid per hit, making it difficult to pay on an hourly basis. We attempt to alleviate this by estimating the time per HIT and then extrapolating the pay to £15/hr. We believe this to be reasonable as it's significantly higher than the UK living wage (£10.42). We also anonymize the data and add a feedback box for users.

We selected the countries available to our MTurk based on the highest worker quality for that language based on (Pavlick et al., 2014). We create an interface, as seen in Figure 3.1. The first page contains instructions, with boxes where they could confirm they were a native speaker of the language and wouldn't use online machine translation programs. The next page contained 16 English sentences and a box to enter their translation. As machine translation was banned, our script disabled copy/pasting.

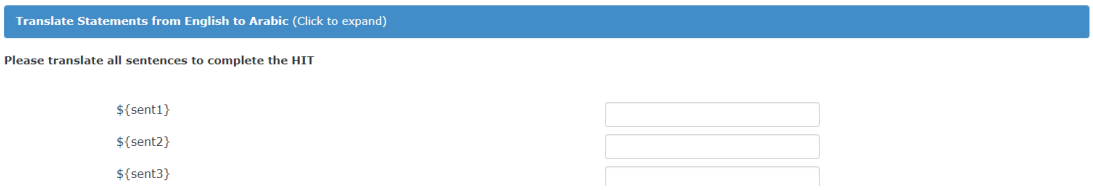
The results for the quality of each language can be seen in Table 3.2. These are on a pilot study of 3 HITs (48 sentences), for 3 workers (if all HITs are valid). Immediately, we see that only a small percentage of HITs were accepted in all languages. We found that the same users claimed to be native to every language and bad data was a common

occurrence. While they did have to state they were a native speaker, we had no way to validate this.

Oftentimes Google Translate was used (classified as such when BLEU > 0.8), or they didn't attempt the task at all, either entering nothing or entering the original English.



(a) Instructions page



(b) Translation page

Figure 3.1: Interface for English-Arabic MTurk translation

	bleu < 0.8	lang > 0.5	len > 0.75	brac $= 1$	num of hits	overall
russian	0.56	0.60	0.80	0.60	25	0.08
chinese	0.38	0.77	0.77	0.58	26	0.08
arabic	0.45	0.55	0.55	0.55	22	0.00

Table 3.2: Success rate of each HIT for different quality measurements. All need to be true for a HIT to be accepted. Lang is if the sentence is classified in its respective language by LANGID, len is if the crowdsourced sentence is within 7 words of the original sentence, brac is a check for correct bracket formatting. Overall checks if all of the previous quality measurements are true.

3.2.4 In-house

We then collected the data in-house, by this, we mean from students/staff at the university. This allowed us to personally vet each translator, allowing us to ensure that they are a native speaker. It also meant any questions related to language-specific phenomena could be asked. In order to increase time efficiency, we attached the machine translation of each sentence beside the English, as seen in Figure 3.1. If the quality of the translation was correct, the person could simply copy and paste it. Otherwise, they were instructed to edit/rewrite as necessary.

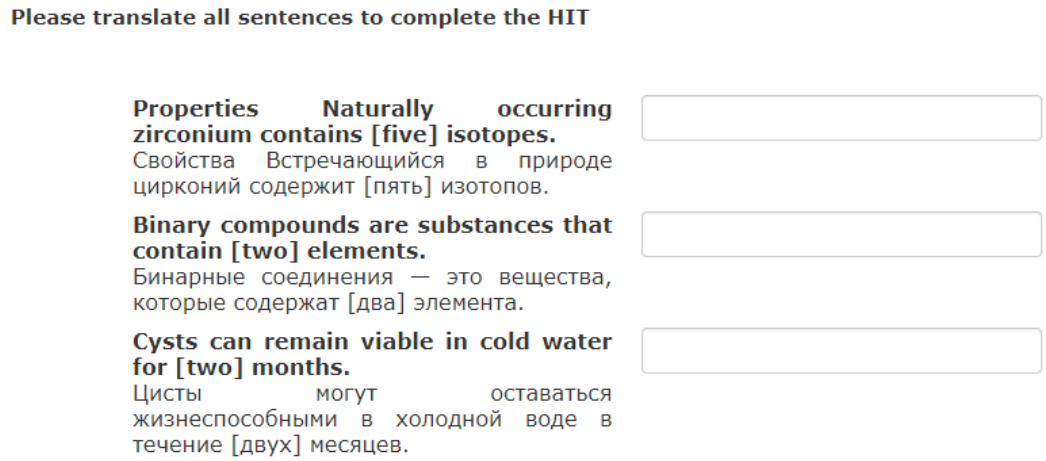


Figure 3.1: Example sentences for in-house translation from English to Russian.

We also decided to only translate approximately half of the dataset to Arabic, to allow for low-resource experiments. We can see the final translation results in Table 3.3. These results are discussed in-depth in Section 3.2.5.

	in house sr	mt sr
russian (n = 10444)	0.98	0.88
chinese (n = 10444)	0.98	0.76
arabic (n = 5504)	0.72	0.73

Table 3.3: Success rate (sr) of well-formatted sentences for each language using in-house translation (n = total number of in-house sentences translated).

In order to evaluate a potential translator, we first gave them a set of 112 pilot sentences. These could then be compared across translators for each language. We see the pilot results in Table 3.4.

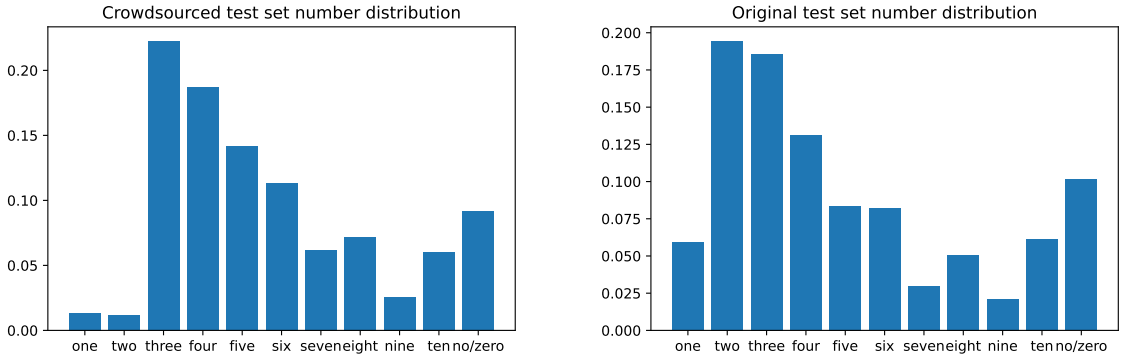
	BLEU	TER	Exact
Russian	0.80	0.14	0.57
Chinese	0.77	0.11	0.21
Arabic	0.76	0.15	0.43

Table 3.4: In-house translation results from the same set of pilot sentences (n = 112). Exact = sentence exactly matching MT

3.2.5 Limitations

We required the number, and only the number to be in written form and bracketed. For example, *"A cat has [four] legs and [two] eyes."* This is necessary for evaluating and training the task. However, this created issues when translating to Arabic. As discussed

earlier, Arabic has three plurals, including a dual. Often, when referring to a dual noun in Arabic it is unnatural to also say the number associated with it. Our example would be translated as "القطة لديها [اربع] أرجل و[عينان]." The problem with this sentence is that *"[two] eyes"* translates into *"[eyes]"*, where the plural of eyes implies two. You can see the effect this has on the number distribution by comparing the sub-figures in Figure 3.2.



(a) Crowd-sourced distribution.

(b) Original English distribution.

Figure 3.2: Normalised distributions of the frequency of each number in the test set (same sentences across all languages).

There were also some ungrammatical/bad sentences from the original data, such as *"T tetrahedron A volumetric connection element that connects [four] positions in a field."* Some sentences were impossible to translate into a given language as there was no word/phrasing for the given English. For example, *"Collaborative commerce is the extension of core business processes beyond a company's [four] walls."* is impossible to translate into Chinese. Both of these cases made sentences that were impossible to translate.

It's important to recognize that the dataset suffers from a cultural bias due to originally being in English. What's true in one language may not be true in another. Consider the sentence *"Continents are the [seven] main divisions of land on Earth."*. This is true, in English. However, continents are a cultural phenomenon (Lewis & Wigen, 1997) and in other parts of the world the world consists of four to six continents. Our project left all translations based on English commonsense.

We were not able to translate the original test set as its gold (true values for sentences) was withheld. Additionally, the validation set did not replicate the style or structure of the training/test sentences. The training set consists of sentences like *"Electrons are the smallest and lightest of the [three] particles and they have a negative charge."* while the validation set had *"how do you win at tic-tac-toe get [three] of your symbols in a row"*. Sentences in this style never appear in training. The test and validation sets contain 1,500 and 500 sentences. These sentences are the same (but translated) for all languages. Sentences are only added to test/validation if they're valid across all languages. If only some of the sentences can be properly translated, they're added to the training set.

Chapter 4

Approach

In this chapter, we discuss our preprocessing steps and the necessary adjustments to the original task required for finetuning and inference.

4.1 Preprocessing

All our data is initially formatted with its sentence and brackets around any numbers from zero to ten. All numbers are in written form and all possible numbers are listed in Appendix A.2. Any sentences that were incorrectly formatted or contained different numbers to the English original were thrown away.

Once that step has been completed, we remove bracketing on the training set. For our test/validation set, given a sentence, we select a *number word* at random and replace it with a <*mask*>, storing its true value in a gold set. If there are other bracketed numbers in that sentence, their brackets are also removed.

Our dataset is tokenized in two ways, depending on the model used. WordPiece (Wu et al., 2016) is used by mBERT and assumes that the input text uses spaces to separate words, if it's a language like Chinese, it tokenizes it into its component characters. SentencePiece (Kudo & Richardson, 2018) is used by xlm-RoBERTa, mBART, MT5, and m-GPT, and tries to use language-specific pre-tokenizers.

There were a few issues with the SentencePiece tokenization. In Chinese, it often tried to combine a number with its unit (years, pounds, etc.). As we are trying to mask the number only, we cannot finetune these. So, we performed a split on the specified numerical characters during pre-tokenization, meaning the units are no longer attached.

4.2 Finetuning and inference

In this section we discuss how we finetune and infer for our task, we look at each model in the project and compare how they achieve this.

4.2.1 mBERT

Finetuning and inference for mBERT was very similar to that described in Lin et al. (2020). We mask the *number words* in each sentence and finetune BERT on only these tokens. Inference is performed by replacing the mask with possible number tokens and seeing if those with the top score match the gold. Our contribution is extending the words to Appendix A.2, depending on the language used.

Scores are calculated as follows:

$$s(\text{word}) = s(\text{token}_{\text{word}} \mid \text{sent}) \tag{4.1}$$

Where $s()$ is the scoring function (model output logits), *sent* is the full sentence, and $\text{token}_{\text{word}}$ is the token for the *number word*. After all scores are predicted, we perform a softmax.

4.2.2 xlm-RoBERTa

With SentencePiece, some numbers were tokenized into multiple parts, such as *"nine"* into "*_ni", "ne"*. NUMERSENSE did not cover this case, so we adjust our finetuning to mask all subwords that make up a *number word*. For inference, RoBERTa predicts only one <*mask*> token. Single-token number words are calculated the same as mBERT. For multi-token words, we input succeeding masks for each token. For example, if we were trying to infer *"nine"* it would look like this:

```
_a _cat _has <mask> <mask> _lives
_a _cat _has _ni    ne     _lives
```

Scores are calculated as follows:

$$s(\text{word}) = \frac{\sum_1^n s(\text{token}_i \mid \text{sent}, \text{token}_{i-1}, ..., \text{token}_1)}{n} \tag{4.2}$$

Where token_i refers to the ith token that makes up the word and n is the total number of tokens.

4.2.3 mBART

mBART works mostly the same as xlm-RoBERTa. The only difference of note is that BART can perform mask-fill on variable lengths, so we only require a single mask token.

4.2.4 mT5

T5 cannot make use of the standard mask-fill pipeline. So instead, we make use of a workaround. We replace any *number words* in a sentence with a sentinel token. These are a unique mask token for T5, as discussed in Section 2.1.5.

For example,

```
input: A cat has two eyes and four legs
```

Is finetuned as:

```
input: A cat has <extra_id_0> eyes and <extra_id_1> legs
labels: <extra_id_0> two <extra_id_1> four <extra_id_2>
```

Getting the loss using these inputs and labels effectively performs a mask-fill.

During inference, we replace our mask with a sentinel token. Then, we calculate the score in the same way as Equation 4.2.

4.2.5 mGPT

With mGPT we get the probability of each sentence, this is done by replacing the mask with some *number word* and getting its loss. Whichever word gets the lowest loss is then our prediction. We do not report finetuning results as the model never learned.

Chapter 5

Experiments

In this chapter, we describe our experiments and their results. Experiment 1 looks at whether fine-tuning can improve performance and, which model(s) perform best. Experiment 2 compares the performance of machine-translated sentences versus those of crowd-sourced. Experiment 3 is a breadth study of performance across languages, it explores which models perform best and why. The experiments also explore whether cross-lingual training improves performance. Finally, Experiment 4 is a depth study. It explores some phenomena for the languages selected in the project.

In this chapter, we look at the results of the experiments discussed in the previous chapter and analyze our findings. Model details can be found in Appendix A.

5.1 Experiment 1: Models

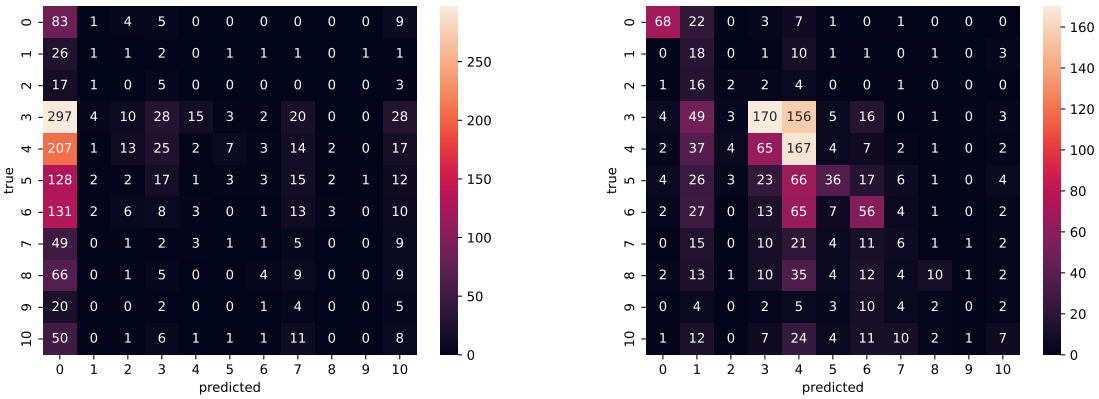
These experiments relate to the models that we're using in this project, we compare these models and look at why they perform differently.

5.1.1 Experiment 1a: Zero-shot versus training

This experiment compares zero-shot models to trained ones. We perform pre-trained and finetuned experiments on our models. We measure this experiment on English, Russian, Arabic, and Chinese. This experiment should be a useful metric as a general overview of how data can impact performance.

We found that all results (Table 5.1) improved in performance after the training, which was expected. We found that Arabic had the smallest improvement, which is likely explained by the low training size.

Some model-language combinations had a large increase in performance, in particular, we found pre trained Chinese mT5 and pre-trained Russian mBART generally performed poorly. These significantly improved over finetuning. Interestingly, Russian had the worst pretraining performance on mBART despite having the second-largest pretrained corpus size, as per Table 5.5.



(a) Pretrained. (b) Finetuned.

Figure 5.1: Heatmap of the confusion matrix on Russian mBART. The y-axis refers to the true number, ordered from 0 to 10 (top to bottom). The x-axis refers to the predicted number, ordered from 0 to 10 (left to right). The number and colour of each block imply how many have that true-prediction combination. We use this format for all heatmaps in the project.

To see the impact of finetuning, we first compare our pretrained (Figure 5.1a) and finetuned (Figure 5.1b) model. When pretrained, mBART tens to just predict a single number (0), implying that the model doesn't really care about the sentence in prediction. However, when finetuned we see a diagonal line begin to form. Furthermore, we look at the attention distribution on pre-trained (Figure 5.2) and finetuned (Figure 5.3) mBART on a Russian sentence. We see that the pretrained model struggles to represent any attention on 'паль' (finger) across each layer. On the other hand, finetuning shifts the language model to pay attention to 'паль'.

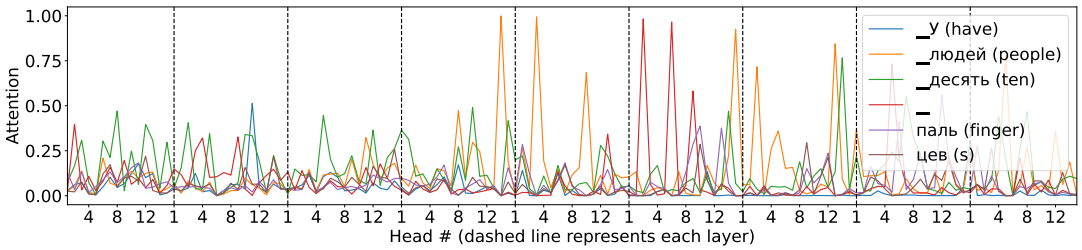


Figure 5.2: Encoder attention distribution on "десять" (ten) from "У людей десять пальцев" (people have ten fingers) for pretrained mBART.

We also see that pretrained mBART predicts sporadically over many of the potential numbers (Figure 5.4a). When it is finetuned (Figure 5.4b) the model only predicts a small subset of the numbers (0, 3, 4, and 5). This gives good results, even though it doesn't *understand* the problem as the test distribution has many cases for these numbers.

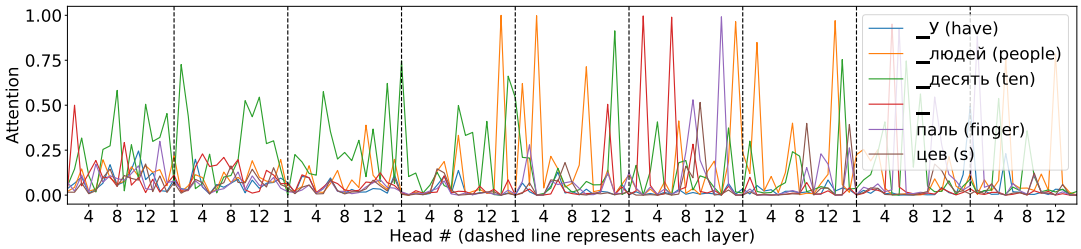
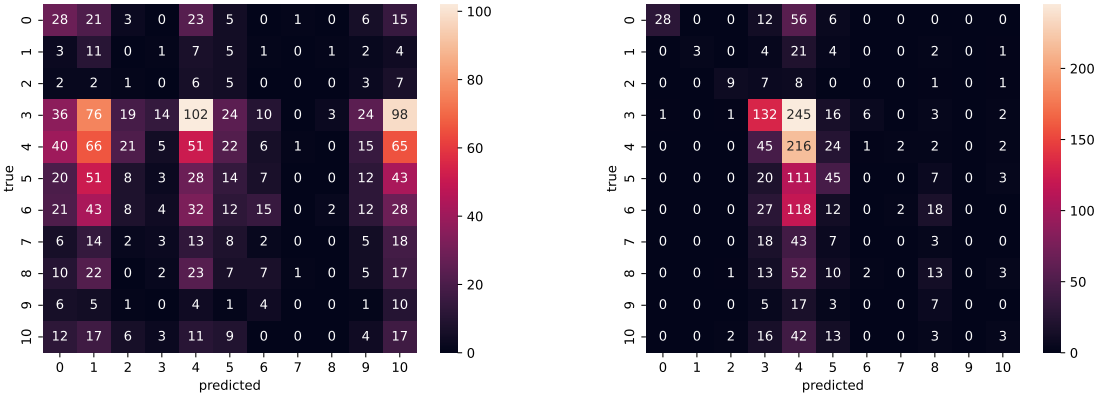


Figure 5.3: Encoder attention distribution on "десять" (ten) from "У людей десять пальцев" (people have ten fingers) for finetuned mBART.



(a) Pretrained.

(b) Finetuned.

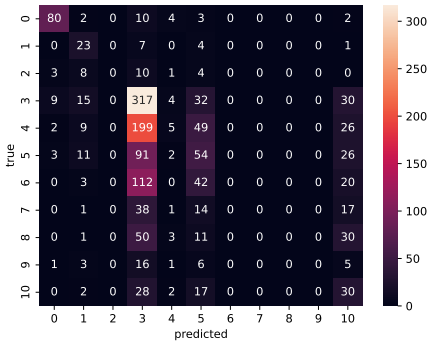
Figure 5.4: Heatmap of the confusion matrix on English mBART.

5.1.2 Experiment 1b: Best model(s)

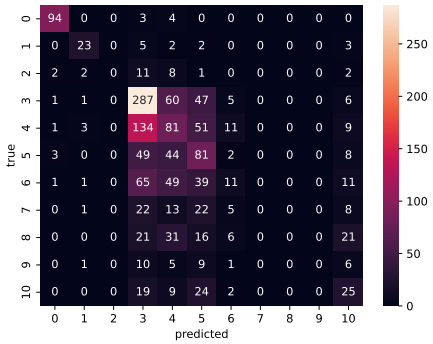
In this experiment, we look at the results discussed in Experiment 1a and look for the model(s) that perform best for each language. We then look at the models in-depth to see what insights of each model entail our results.

The best model across our experiments was xlm-RoBERTa large (Table 5.1). As expected, the largest of each language model performed the best. This is because larger models have more trainable parameters. Unsurprisingly, xlm-roberta-base outperformed bert-base-multilingual-uncased due to its longer training time.

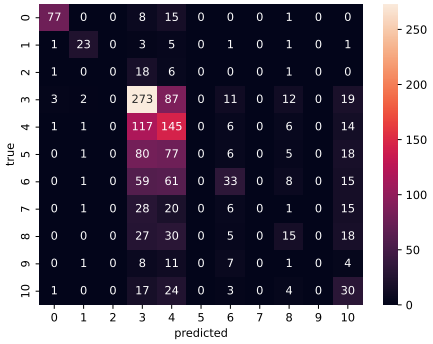
We can also compare *how* the models perform differently. Finetuned Arabic performance (Figure 5.5) tends to predict from only a subset of numbers (a subset of 0, 3, 4, 5, and 10) across our models, instead of trying to predict across all numbers. From this subset, predictions seem to be made almost randomly, indicating a lack of understanding of the model. Interestingly, this subset differs depending on the model. By comparing the attention distribution of a finetuned encoder-decoder (xlm-roberta-base) and autoregressive (mt5-base) model (Figure 5.6), we see that T5's distribution is much more sporadic. It tries to attend to every word. Whereas RoBERTa only aims for a few words (*one* and *in*), but misses on *Cancer*, which is important for the overall inference.



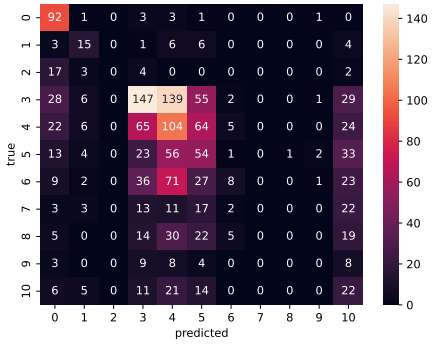
(a) bert-base-multilingual-uncase



(b) xlm-roberta-large

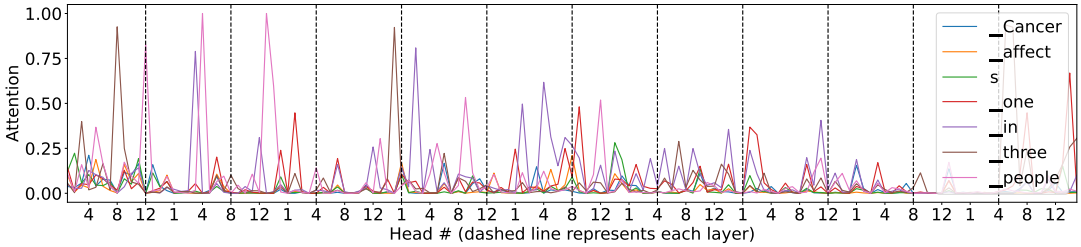


(c) t5-large

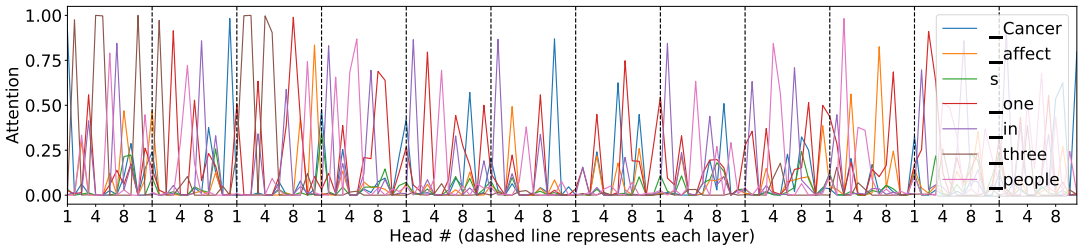


(d) mbart-large-cc25

Figure 5.5: Heatmap of the confusion matrix for models finetuned on Arabic.



(a) xlm-RoBERTa-base



(b) mT5-base encoder

Figure 5.6: Attention distribution on '*three*' from '*Cancer affects one in three people.*'

	English	Chinese	Russian	Arabic
pretrained				
bert-base-multilingual-uncased	28.3	18.1	24.1	26.7
xlm-roberta-base	30.9	23.9	27.6	28.4
xlm-roberta-large	33.8	27.9	30.2	34.1
mt5-small	23.9	5.20	12.7	30.6
mt5-base	32.5	5.50	16.9	33.8
mt5-large	35.9	6.50	22.7	36.7
mbart-large-cc25	10.1	17.3	8.70	12.0
m-gpt	33.9	26.7	29.7	20.0
finetuned				
bert-base-multilingual-uncased	46.0	39.7	46.9	33.9
xlm-roberta-base	45.7	41.0	35.9	37.0
xlm-roberta-large	**48.5**	**43.7**	**48.4**	**40.1**
mt5-small	36.5	31.4	35.1	31.7
mt5-base	41.0	33.8	40.7	37.0
mt5-large	45.9	34.5	43.6	39.7
mbart-large-cc25	29.0	27.7	36.0	29.5

Table 5.1: Results for Experiments 1a and 1b (Section 5.1.1 and 5.1.2). Accuracies of pre-trained/fine-tuned models on different languages. Hyper-parameters can be found in Appendix A.1.

5.1.3 Experiment 1c: Object bias

In this experiment, we investigate if our pre-trained models are biased toward certain numbers. We get two sentences and their translations in English, Russian, Arabic, and Chinese. We then fill the subject noun with 1000 random words of that language and investigate its results. Do words such as 'legs' and 'sides' bias our models to predict a particular number? Is this reasoning constant across languages?

Our results (Table 5.2) show that our models generally don't change behavior with different words filling the subject noun. This implies a heavy bias toward specific numbers. We also see that larger models tend to be less biased. The numbers across our models for "*All [X] have <mask> sides.*" and its translations were 0, 1, 2, 3, 4, 8, 9, and 10. For "*All [X] have <mask> sides.*" these numbers were 0, 1, 2, 3, 4, 5, 6, and 10.

Interestingly, sentences that were direct translations of each other didn't necessarily bias towards the same number across the same model. For example, with xlm-roberta-base, "*All [X] have <mask> sides.*" is biased towards the number 2, while its Chinese translation, "所有*[X]*有*<mask>*个边。", biases towards 1. This may imply that the reasoning of a sentence with the same semantic meaning may be different depending on the language. The behavior of random cross-lingual crossover was especially prevalent in languages that had the poorest performance in our pretraining experiments (Table 5.1).

Sentence	xlm-roberta-base	xlm-roberta-large
All *[X]* have to have <mask> legs.	2 (97.8)	2 (78.5), 4 (21.2)
All *[X]* have <mask> sides.	2 (98.9)	2 (99.5)
所有*[X]*都必须有<mask>条腿	2 (99.1)	3 (62.3), 2 (28.9)
所有*[X]*有<mask>个边。	1 (99.9)	2 (57.7), 3 (41.3)
Все *[X]* должны иметь <mask> ног.	2 (98.7)	2 (48.5), 5(46.1)
Все *[X]* имеют <mask> стороны.	2 (99.5)	2 (99.0)
[X] يجب أن يكون لكل <mask> أرجل.	3 (99.3)	3 (59.8), 4 (35.1)
[X] كل <mask> جوانب.	3 (94.5)	3 (75.4), 4 (24.2)

Sentence	mbart-large-cc25	mt5-small
All *[X]* have to have <mask> legs.	4 (57.5), 0 (39.4)	2 (86.9), 1 (11.7)
All *[X]* have <mask> sides.	4 (96.0)	2 (99.5)
所有*[X]*都必须有<mask>条腿	1 (98.0)	1 (99.3)
所有*[X]*有<mask>个边。	1 (78.7), 0 (20.3)	1 (88.1), 2 (10.9)
Все *[X]* должны иметь <mask> ног.	0 (98.8)	1 (99.5)
Все *[X]* имеют <mask> стороны.	0 (61.2), 10 (33.3)	2 (99.5)
[X] يجب أن يكون لكل <mask> أرجل.	9 (74.7), 4 (17.3)	3 (87.6), 0 (10.4)
[X] كل <mask> جوانب.	8 (87.3)	3 (85.0), 6 (13.0)

Sentence	mt5-base	mt5-large
All *[X]* have to have <mask> legs.	2 (49.0), 1 (35.6), 0 (15.4)	2 (81.6)
All *[X]* have <mask> sides.	0 (50.4), 2 (48.9)	2 (81.2), 0 (17.5)
所有*[X]*都必须有<mask>条腿	0 (74.7), 1 (25.1)	1 (98.0)
所有*[X]*有<mask>个边。	0 (87.5), 1 (12.5)	1 (78.7), 0 (20.3)
Все *[X]* должны иметь <mask> ног.	1 (64.5), 0 (35.3)	1 (96.8)
Все *[X]* имеют <mask> стороны.	1 (97.4)	1 (72.3), 2(26.7)
[X] يجب أن يكون لكل <mask> أرجل.	3 (97.7)	3 (82.5), 4 (16.2)
[X] كل <mask> جوانب.	3 (64.4), 0 (35.3)	3 (82.7), 10 (12.5)

Sentence	m-gpt	mbert-base
All *[X]* have to have <mask> legs.	2 (97.8)	2 (99.1)
All *[X]* have <mask> sides.	2 (99.9)	3 (51.3), 2 (48.6)
所有*[X]*都必须有<mask>条腿	2 (99.3)	2 (99.1)
所有*[X]*有<mask>个边。	4 (99.4)	1 (99.9)
Все *[X]* должны иметь <mask> ног.	3 (49.7), 9 (24.2), 8 (20.5)	5 (93.5)
Все *[X]* имеют <mask> стороны.	0 (96.7)	2 (78.9), 1 (16.3)
[X] يجب أن يكون لكل <mask> أرجل.	8 (90.7)	3 (60.2), 0 (17.8)
[X] كل <mask> جوانب.	10 (58.0), 8 (35.0)	3 (58.0), 0 (39.5)

Table 5.2: Results for Experiment 1c. Sentences are ordered as pairs of *All [X] have to have <mask> sides* and *All [x] have <mask> side* in the respective languages. Results are a list made of elements in the format: *predicted number (occurrence %)*. Only numbers with $\geq$ 100 instances were included. mbert-base is bert-base-multilingual-uncased

5.2 Experiment 2: Machine translation vs crowd-sourcing

This experiment compares our crowd-sourced data to that made through machine translation. This can give a rough indication of the quality of our datasets. GPT is not discussed here as it's not "fine-tuned" on a training set like other models.

In this experiment, we compare the performance of our crowdsourced results versus machine translation (Table 5.3). Generally, our crowd-sourced results are better, which indicates that the results from it are better for training. The reasons for this are twofold; higher quality training data, and more valid training sentences.

Crowd-sourced Arabic had roughly the same training size as its MT, yet generally had improved results. We do find two exceptions to improvement. As the exceptions are rare and only have a small difference to the crowdsourced accuracies (0.8 and 0.4), we don't take them as significantly indicative of the overall quality of our dataset.

	Chinese MT	Russian MT	Arabic MT
bert-base-multilingual-uncased	38.3 (-)	45.3 (-)	34.4 (-)
xlm-roberta-base	35.0 (-)	36.7 (+)	36.5 (-)
xlm-roberta-large	43.3 (-)	47.9 (-)	36.7 (-)
mt5-small	29.9 (-)	32.1 (-)	29.7 (-)
mt5-base	24.5 (-)	39.6 (-)	37.4 (+)
mt5-large	29.9 (-)	43.4 (-)	36.5 (-)
mbart-large-cc25	19.0 (-)	28.5 (-)	29.7 (-)

Table 5.3: Results for Experiment 2 (Section 5.2). Models are finetuned on MT training set. The sign (+/-) indicates whether performance was improved when compared to Table 5.1.

5.3 Experiment 3: Across languages

These experiments are focused on languages used across our dataset and they are a breadth study. We evaluate the best-performing languages, and explain why. We also look at whether cross-lingual data improves performance.

5.3.1 Experiment 3a: Best language performance

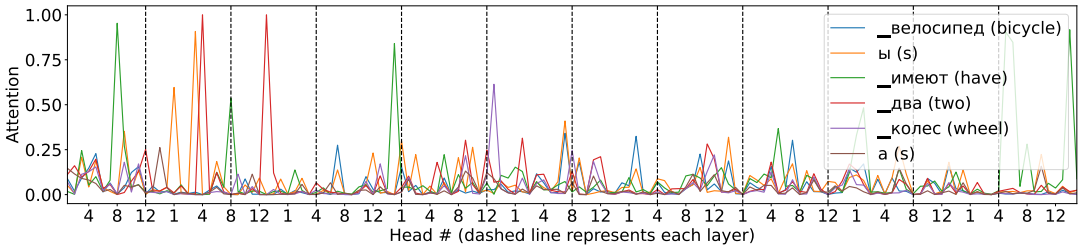
This is a relatively simple experiment where we find which languages perform best across our models. We use the insights from this experiment as a motivation to look at how each language predicts and the factors behind its performance. We perform case studies on plural forms, attention distribution, and general differences between the languages and dataset to investigate this.

We find that Russian and English generally perform the best across our models (Table 5.4). This is partially due to those languages being the largest when pretrained (Table 5.5). We also see that Chinese generally performs closely to English/Russian,

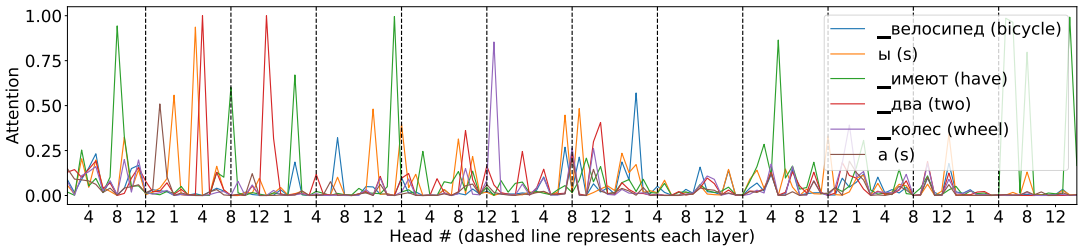
while Arabic does not, which implies that the training size is a significant factor in performance.

	mbert-b	xlmr-b	xlmr-l	mt5-s	mt5-b	mt5-l	mbart-l	mgpt
English								
pretrained	28.3	30.9	33.8	23.9	32.5	35.9	10.1	**33.9**
finetuned	46.0	**45.7**	**48.5**	**36.5**	**41.0**	**45.9**	29.0	-
Chinese								
pretrained	18.1	23.9	27.9	5.20	5.50	6.50	17.3	26.7
finetuned	39.8	41.0	43.7	31.4	33.8	34.5	27.7	-
Russian								
pretrained	24.1	27.6	30.2	12.7	16.9	22.7	8.70	29.7
finetuned	**46.9**	35.9	48.4	35.1	40.7	43.6	**36.0**	-
Arabic								
pretrained	26.7	28.4	34.1	30.6	33.8	36.7	12.0	20.0
finetuned	33.9	37.0	40.1	31.7	36.0	39.7	29.5	-

Table 5.4: Results for Experiment 3a (Section 5.3.1). Accuracies of different languages across a variety of models. Model names are: bert-base-multilingual-uncased, xlm-roberta-base, xlm-roberta-large, mt5-small, mt5-base, mt5-large, mbart-large-cc25 and m-gpt respectively.



(a) Pretrained xlm-RoBERTa-base



(b) Finetuned xlm-RoBERTa-base

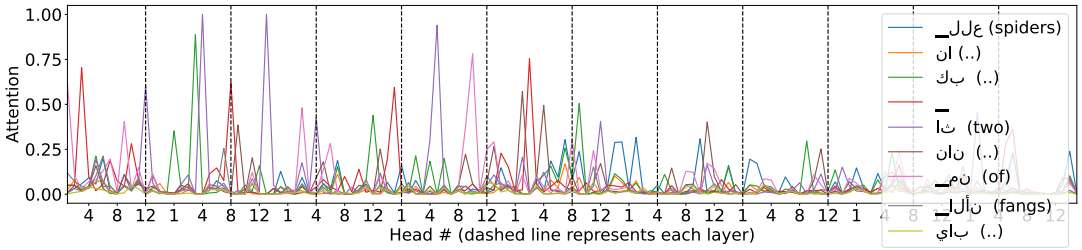
Figure 5.7: Attention distribution on два (two) from велосипеды имеют два колеса (bicycles have two wheels).

Arabic's pretrained performance was generally the best. We believe this is due to the test distribution (Figure 3.2) being skewed to only allow sentences that have a valid translation in every language. This meant that 'easy' guesses for other languages (one and two) didn't appear as often.

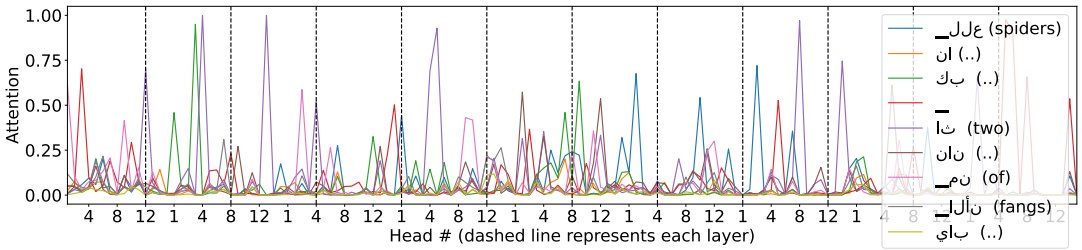
	English	Russian	Chinese	Arabic
pretrained size (GB)	10401	3615	186	237
train size (# total sent)	7918	7153	7420	1876

Table 5.5: Size per language in C4 corpus and our training sets. GB stands for Gigabytes.

In Section 3.1, we mentioned one of the motivations for choosing our language is plurality, we investigate this here by looking at how the attention distribution is performed on these suffixes.



(a) Pretrained xlm-RoBERTa-base



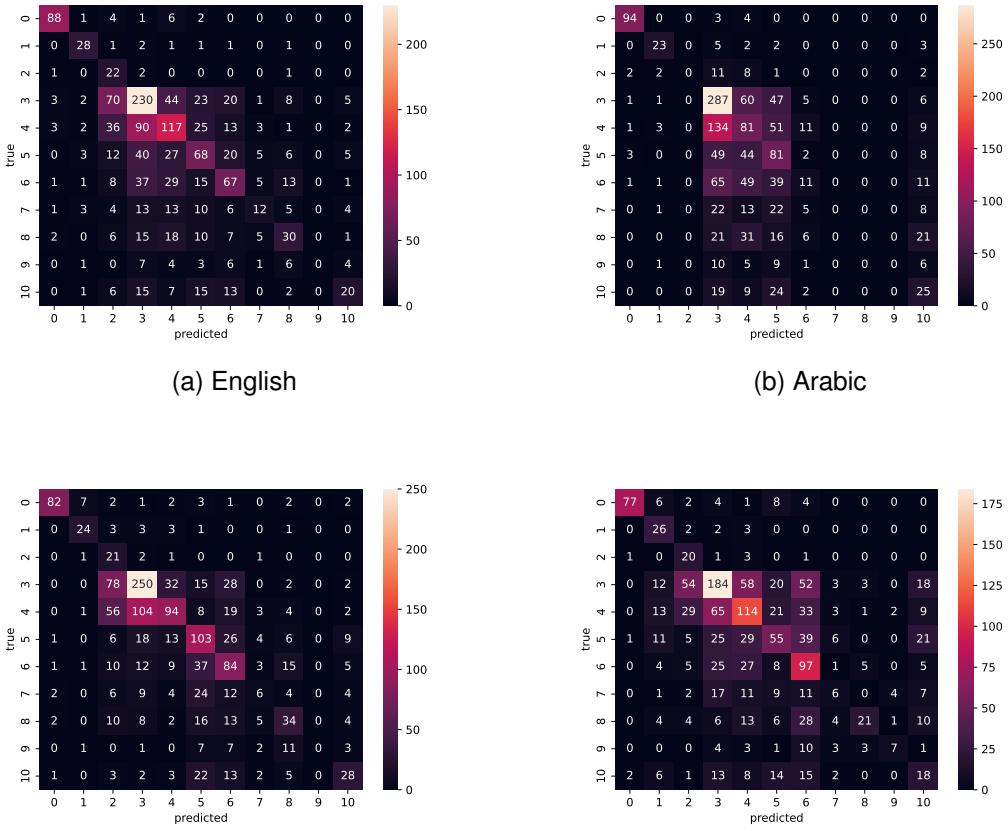
(b) Finetuned xlm-RoBERTa-base

Figure 5.8: Attention distribution on "اثنان" (two) from ".اللعناكب اثنان من الأنياب" (spiders have two fangs). Tokens don't always translate so (..) indicates its a part of the preceding word.

In Figure 5.7, we see that our finetuned LM pays more attention to a (a plural suffix that implies 2-4) in the upper layers, while the pre-trained model does not. This indicates the LMs *learns* to use plural forms for Russian inference. With Arabic (Figure 5.8), we find little improvement on using dual forms after being finetuned. There is no attention on *fangs*. This is partially due to the low number of 'two' cases (62 examples) in the Arabic training set. You can see this dual as the green in this figure.

We can see how the language impacts what models predict in Figure 5.9. Specifically, we look at xlm-roberta-large, but similar patterns emerge in our other models. The

diagonal line is the correct prediction. In English, Russian and Chinese we see that the models tend to predict numbers between 2-6 when it is wrong. While there is a visible diagonal in these languages (i.e correct predictions), the predictions in this area cause confusion. For numbers outside this range, xlm-roberta-large usually guesses correctly. The numbers these models guess mostly seem to follow their *training* distribution. Arabic only guesses in a tight range (0, 3-5), and never guesses 2, 6, 7, 8, or 9 across its predictions. Interestingly, 9 is never guessed in English or Russian either.



(a) English (b) Arabic

(c) Russian (d) Chinese

Figure 5.9: Heatmap of the confusion matrix for xlm-roberta-large finetuned on different languages.

5.3.2 Experiment 3b: Cross-lingual performance

In this experiment, we investigate whether a cross-lingual dataset improves performance, based on the discussion in Section 2.3.3. We look at English data added to each of our languages, and a training set of all languages mixed together. We perform two variants in this experiment; balanced and padded. Balanced experiments have the same % of each language in the training set, while padded adds extra sentences if they exist.

We found that cross-lingual experiments generally performed worse than the language by itself. This is because, while the total number of sentences is similar to the monolingual training sets, there are fewer sentences in the language we are testing for. We get slightly

lower accuracy for half the sentences in the original training file, implying that some cross-lingual learning did occur.

On our multilingual set, we found performance to be worse than, but still close to the average performance of training each language individually. In our balanced example, we only provided $\sim$ 300 samples per language, yet still got similar performance, indicating that cross-lingual learning occurred.

Our padded performance had a performance on par with, or worse than the balanced set. This was despite our balanced set having a smaller training size than its padded equivalent. This shows that a larger training set may not result in increased performance.

	mbert-b	xlmr-b	xlmr-l	mt5-s	mt5-b	mt5-l	mbart-l
padded							
cn + en	37.8 (-)	41.9 (+)	42.7 (-)	28.1 (-)	28.5 (-)	29.1 (-)	27.9 (+)
ru + en	43.9 (-)	45.3 (-)	45.4 (-)	31.0 (-)	39.5 (-)	41.9 (-)	29.2 (-)
ar + en	31.0 (-)	35.7 (-)	35.7 (-)	30.8 (-)	33.5 (-)	36.4 (-)	31.5 (+)
all	38.9	36.1	39.7	31.4	32.4	33.5	20.6
balanced							
cn + en	37.5 (-)	37.3 (-)	40.0 (-)	30.3 (-)	30.5 (-)	31.5 (-)	27.9 (+)
ru + en	44.1 (-)	45.8 (-)	46.2 (-)	31.5 (-)	39.5 (-)	40.0 (-)	30.1 (-)
ar + en	29.1 (-)	40.1 (-)	36.7 (-)	29.4 (-)	32.9 (-)	35.9 (-)	29.5 (=)
all	38.3	37.1	36.6	34.6	35.5	36.5	25.2
avg	41.6	39.9	45.2	34.3	37.9	40.9	30.6

Table 5.6: Results for Experiment 3b (Section 5.3.2). The sign (+/-) indicates whether the performance was improved when compared to Table 5.4. *avg* is the average of the three languages given from Table 5.4, and *all* uses a multilingual training set.

5.4 Experiment 4: Language-specific phenomena

This experiment serves as a depth study for each of the languages we have crowdsourced. In Russian we experiment with the model's ability to capture case declension, in Arabic we explore declension and in Chinese, we measure word reliance.

5.4.1 Experiment 4a: Russian case declension

Declension in Russian serves primarily to delineate the grammatical and semantic information contained in words in a sentence. Declension in the suffix gives the gender, number, and case of words in Russian, with case declension varying depending on a word's gender and number. Do models understand not only the number but also the type of number?

We experiment with the ability of Russian to predict the case in declension in a numerical context. The categories for these are made up of nominal, accusative, dative,

instrumental, prepositional, and genitive. Some declensed numbers are homonyms, when this is true we categorize it as a member of all possible cases that fit its meaning.

	Nominal	Accus	Dative	Instru	Prep	Genitive	None	Total
pretrained								
mbert	8.76	10.7	11.6	10.2	14.4	15.0	97.4	16.8
xlm-b	25.0	26.0	22.9	2.04	19.1	19.7	**100**	26.2
xlm-l	26.3	27.7	23.9	2.04	23.7	24.4	98.7	28.7
mt5-s	5.40	5.62	8.64	0.00	6.65	7.64	89.7	10.9
mt5-b	10.4	10.4	11.3	0.00	10.2	11.0	94.9	14.9
mt5-l	13.6	14.7	16.3	0.00	19.4	19.7	96.2	20.4
mbart	4.09	3.93	0.997	0.00	0.486	0.478	78.2	6.20
gpt	22.6	23.6	5.98	**34.7**	22.4	22.8	94.9	27.2
finetuned								
mbert	49.5	50.1	30.9	24.5	39.1	39.3	93.6	46.1
xlm-b	48.9	50.0	**53.2**	2.04	42.0	42.2	94.9	46.5
xlm-l	**50.5**	**51.5**	37.5	2.04	**42.1**	**42.4**	84.6	**47.7**
mt5-s	30.5	31.9	22.6	12.2	28.0	28.7	93.6	33.1
mt5-b	35.3	36.8	26.2	28.6	34.5	35.0	89.7	38.3
mt5-l	37.7	39.2	30.9	18.4	40.2	40.3	92.3	41.4
mbart	31.8	31.2	16.9	0.00	34.8	34.9	75.6	33.5
n	685	712	301	49	617	628	78	1500

Table 5.7: Results for Experiment 4a. n is the total number of each declension type in the test set. Accus, Instru, Prep, and None are accusative, instrumental, prepositional, and no declension respectively. Model names are: bert-base-multilingual-uncased, xlm-roberta-base, xlm-roberta-large, mt5-small, mt5-base, mt5-large, mbart-large-cc25 and m-gpt respectively.

We can see the result for this experiment across our models in Table 5.7. Generally, performance in xlm-roberta-large was the best and our fine-tuned models are generally *okay* at predicting nominal, accusative, dative, prepositional, and genitive numbers. The exception to this is instrumentals, which are words that are being used as an instrument to a sentence (e.g I ate with the **spoon**). Instrumentals have extremely poor pre-trained performance and only a slight improvement when fine-tuned (if any). The poor performance could be partially caused by the rarity of its use in our training set relative to the other case types. Interestingly, m-gpt performs best on instrumentals.

Performance on no declension was by far the best. These consisted of words that implied zero or none of and indicate that our model is good at understanding when there is none of a particular object. While we do see the best performance for no declension on pre-trained xlm-roberta-base, this is likely caused by a better generalization of the problem and not a worse understanding of the problem. You could always guess none, and get 100% accuracy, but that doesn't mean you've learned the problem.

5.4.2 Experiment 4b: Arabic declension

This experiment is similar to Experiment 4a, except that it does not analyze a specific type of declension. Instead, we explore declension as a whole to see if the model predicts the number as well as its exact declension.

	mbert-b	xlmr-b	xlmr-l	mt5-s	mt5-b	mt5-l	mbart-l	mgpt
pretrained								
original	26.7	28.4	34.1	30.6	33.8	36.7	12.0	20.0
declension	12.5	12.0	14.8	11.8	12.3	14.3	3.73	11.7
finetuned								
original	33.9	37.0	40.1	31.7	36.0	39.7	29.5	-
declension	28.0	30.8	34.4	13.2	17.5	18.1	22.3	-

Table 5.8: Results for Experiment 4b. Accuracies of Arabic when both declensed and not declensed. Model names are: bert-base-multilingual-uncased, xlm-roberta-base, xlm-roberta-large, mt5-small, mt5-base, mt5-large, mbart-large-cc25 and m-gpt respectively.

Our results are outlined in Table 5.8, we look specifically at how much worse our declension performance is compared to predicting solely the number. We see that when pretrained, our models only understand the correct declensed type between half and a third of the time relative to our original experiments. When finetuned, most of our models get much better and understand declension, with this gap narrowing significantly. The exception is the T5 models. While there is an improvement in performance when finetuned, they still fail in predicting the correct declension type. This indicates that T5 may not understand the problem when it guesses correctly and may be a reason for its poorer performance in Arabic when compared to xlm-roberta.

5.4.3 Experiment 4c: Chinese word reliance

In this experiment, we have measured Chinese word reliance. In Chinese, characters attach to one another (instead of saying '*birds*', you would say '*flock of birds*'). We will perform an object bias test across different units for numbers ('years, months, sets, pieces). We pick two sentences, 所有*[X]* 都必须有<mask> *[Y]* 腿(All *[X]* have to have mask legs) and 所有*[X]* 都有<mask> *[Y]* 边。(All *[X]* have mask sides). We replace the [X] with a random Chinese word and test over a set of units by replacing [Y]. These units are 个(piece), 套(set), 次(number), 岁(year), 层(layer), 分(minute), 月(month), and 条(slip).

Our results found for 所有*[X]* 都必须有<mask> *[Y]* can be found in Table 5.9 and the results for 所有*[X]* 都有<mask> *[Y]* 边。are found in Table 5.10. We find that most of the units used do not have much of an impact on the numbers predicted across each model, with most models defaulting to predicting 0. We did see that xlm-roberta-large, mt5-base (despite mt5-large, the larger model having heavy bias), and m-gpt did

deviate, which implies that they pay attention to their unit. For example, 岁(year) in xlm-roberta-large moves the model towards 1, 3, and 10 based on its sentence and unit.

	个(piece)	套(set)
mbert-base	1 (100.0)	1 (100.0)
xlm-roberta-base	1 (100.0)	1 (100.0)
xlm-roberta-large	2 (54.0), 3 (44.2)	1 (90.2)
mt5-small	1 (99.6)	1 (99.8)
mt5-base	1 (57.9), 0 (41.8)	0 (58.3), 1 (41.4)
mt5-large	1 (99.6)	1 (98.7)
mbart-large-cc25	4 (83.5), 0 (15.8)	0 (65.5), 4 (32.0)
gpt	4 (95.3)	2 (99.7)
	次(number)	岁(year)
mbert-base	1 (99.9)	1 (99.5)
xlm-roberta-base	1 (99.9)	2 (71.0), 3 (23.8)
xlm-roberta-large	3 (77.2), 2 (14.2)	1 (63.6), 3 (28.3)
mt5-small	1 (99.6)	1 (99.2)
mt5-base	1 (64.0), 0 (35.8)	0 (60.8), 1 (38.2)
mt5-large	1 (99.4)	1 (99.9)
mbart-large-cc25	0 (60.1), 4 (26.9), 3 (13.0)	4 (92.3)
gpt	2 (98.8)	3 (53.2), 2 (40.7)
	层(layer)	分(minute)
mbert-base	1 (100.0)	1 (60.5), 10 (35.8)
xlm-roberta-base	1 (91.6)	1 (97.0)
xlm-roberta-large	3 (67.1), 2 (32.7)	3 (47.9), 2 (27.3), 1 (12.5)
mt5-small	1 (99.7)	1 (99.8)
mt5-base	1 (72.9), 0 (26.7)	1 (79.4), 0 (20.2)
mt5-large	1 (99.8)	1 (98.9)
mbart-large-cc25	0 (69.5), 4 (23.8)	0 (49.6), 4 (38.7), 3 (11.4)
m-gpt	2 (71.2), 3 (28.8)	5 (45.5), 3 (18.4), 3 (15.8), 7 (11.1)
	月(month)	条(slip)
mbert-base	1 (92.0)	1 (100.0)
xlm-roberta-base	1 (100.0)	1 (100.0)
xlm-roberta-large	1 (65.4), 3 (32.1)	2 (57.7), 3 (41.3)
mt5-small	1 (99.6)	1 (99.3)
mt5-base	0 (76.6), 1 (23.1)	0 (74.7), 1 (25.1)
mt5-large	1 (91.1)	1 (98.0)
mbart-large-cc25	4 (90.6)	0 (81.8), 4 (16.9)
m-gpt	3 (76.7), 5 (14.6)	2 (99.3)

Table 5.9: Object bias for 所有*[X]* 都必须有<mask> *[Y]* 腿. where [X] is filled with one of 1000 random words and [Y] is filled with the unit of that column. Results are a list made of elements in the format: *predicted number (occurrence %)*.

	个(piece)	套(set)
mbert-base	1 (99.9)	1 (99.8)
xlm-roberta-base	1 (99.2)	1 (99.9)
xlm-roberta-large	3 (62.3), 2 (28.9)	1 (97.2)
mt5-small	1 (98.7)	1 (91.1)
mt5-base	0 (87.5), 1 (12.5)	0 (66.4), 1 (33.4)
mt5-large	1 (78.7), 0 (20.3)	1 (91.7)
mbart-large-cc25	3 (48.8), 4 (37.5), 0 (12.5)	3 (35.5), 4 (31.1), 0 (14.1)
gpt	4 (99.4)	1 (98.2)
	次(number)	岁(year)
mbert-base	2 (53.6), 1 (40.5)	1 (88.8)
xlm-roberta-base	1 (99.5)	1 (98.7)
xlm-roberta-large	3 (48.3), 1 (42.1)	10 (50.7), 1 (34.8)
mt5-small	1 (99.6)	1 (96.0)
mt5-base	1 (61.7), 0 (38.0)	0 (75.4), 1 (24.4)
mt5-large	1 (81.5), 0 (17.4)	1 (99.2)
mbart-large-cc25	3 (52.5), 4 (31.7), 0 (14.4)	4 (40.3), 3 (39.6), 0 (16.5)
gpt	1 (83.0), 2 (10.0)	1 (83.4), 3 (14.7)
	层(layer)	分(minute)
mbert-base	1 (99.9)	10 (60.9), 1 (16.5)
xlm-roberta-base	1 (99.4)	1 (99.4)
xlm-roberta-large	3 (68.4), 1 (26.9)	1 (74.5), 2 (20.8)
mt5-small	1 (93.7)	1 (98.4)
mt5-base	0 (74.6), 1 (25.4)	0 (57.1), 1 (42.7)
mt5-large	1 (77.5), 0 (21.8)	1 (91.9)
mbart-large-cc25	3 (73.9), 4 (16.9)	3 (68.1), 4 (18.4)
gpt	1 (97.5)	1 (86.5)
	月(month)	条(slip)
mbert-base	4 (37.5), 1 (33.1), 5 (12.8)	1 (100.0)
xlm-roberta-base	1 (100.0)	1 (99.5)
xlm-roberta-large	1 (89.9)	2 (49.8), 3 (35.0), 1 (13.9)
mt5-small	1 (98.2)	1 (84.8), 0 (15.1)
mt5-base	0 (86.2), 1 (13.6)	0 (86.2), 1 (13.8)
mt5-large	1 (84.4), 0 (14.1)	1 (78.5), 0 (20.5)
mbart-large-cc25	3 (61.1), 4 (26.3)	3 (53.7), 4 (32.1), 0 (11.7)
gpt	3 (41.5), 2 (52.4), 4 (14.1), 1 (10.5)	1 (96.4)

Table 5.10: Object bias for 所有*[X]* 都有<mask> *[Y]* 边 where [X] is filled with one of 1000 random words and [Y] is filled with the unit of that column. Results are a list made of elements in the format: *predicted number (occurrence %)*.

Chapter 6

Conclusions

In this chapter, we look at our contributions, a short overview of our results, and potential future work.

6.1 Contributions

The contributions to this project are as follows:

- Created a numerical reasoning dataset with over 22k examples.
- Evaluated and investigated performance on mBERT, xlm-roBERTa, mBART, mT5 and mGPT.
- Analysed the impact of plurality on model performance.
- Performed language-specific experiments on case declension in Russian, declension in Arabic, and Chinese word reliance.

6.2 Results overview

We have collected a high-quality numerical reasoning dataset of over 22k samples for Arabic, Russian, and Chinese. We then performed a range of experiments on the collection of these.

Our approach looked at a large variety of encoder-only (mBERT, xlm-RoBERTa) and auto-regressive PTLMs (mBART, mT5 and mGPT) and found that performance is poor on numerical reasoning across languages. We find this is due to attention often missing important parts of a sequence for reasoning and only ever predicting a small subset of possible answers. We also saw that our models suffered from object bias. We found that models are able to exploit Russian plurality but not Arabic. We also see that cross-lingual learning is beneficial in a numerical commonsense domain. We found declension performance strongly improved through finetuning, but models struggle with instrumental declension. We also found models nearly always knew how to predict no declension. We found declension performance in Arabic to be poor compared to original

performance when pretrained, but fairly close over finetuning. With the exception of this being T5, which often struggled to understand declension. Finally, we look at Chinese word reliance across two sentences, finding that most models suffer from object bias regardless of unit.

6.3 Potential future work

In the future, we would like to explore inter-annotator agreement (Passonneau et al., 2006) on our crowdsourced dataset. This is a more in-depth manner to evaluate the quality of a multilingual dataset. While expensive and time-consuming to do, it would result in a significantly better analysis of translation quality. We also would like to explore the performance of other models on our dataset, in particular, we want to see the capabilities of GPT-4 on our dataset as it has been pretrained on a significantly larger pre trained corpus than mGPT. We would also like to implement a local test set on the languages we performed to see if they're able to predict numerical facts that are culturally specific.

6.4 MInf extension

With the analysis done in this report, we will now try to improve performance. Chain-of-thought (Wei et al., 2022) has provided promising results for better reasoning. This is a method where you try to explicitly define the steps of inference in order to come to an appropriate answer. We believe that using such prompting may provide better results in numeric sense.

Bibliography

Behnke, Maximiliana, Miceli Barone, Antonio Valerio, Sennrich, Rico, Sosoni, Vilelmini, Naskos, Thanasis, Takoulidou, Eirini, Stasimioti, Maria, van Zaanen, Menno, Castilho, Sheila, Gaspari, Federico, Georgakopoulou, Panayota, Kordoni, Valia, Egg, Markus, and Kermanidis, Katia Lida. Improving machine translation of educational content via crowdsourcing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1528.

Bouraoui, Zied, Camacho-Collados, Jose, and Schockaert, Steven. Inducing relational knowledge from bert, 2019. URL https://arxiv.org/abs/1911.12753#.

Brown, Tom B, Mann, Benjamin, Ryder, Nick, Subbiah, Melanie, Kaplan, Jared, Dhariwal, Prafulla, Neelakantan, Arvind, Shyam, Pranav, Sastry, Girish, Askell, Amanda, Agarwal, Sandhini, Herbert-Voss, Ariel, Krueger, Gretchen, Henighan, Tom, Child, Rewon, Ramesh, Aditya, Ziegler, Daniel M, Wu, Jeffrey, Winter, Clemens, and Hesse, Christopher. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Cho, Kyunghyun, Merrienboer, van, Gulcehre, Caglar, Bahdanau, Dzmitry, Bougares, Fethi, Schwenk, Holger, and Bengio, Yoshua. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014. URL https://arxiv.org/abs/1406.1078.

Clark, Jonathan H., Choi, Eunsol, Collins, Michael, Garrette, Dan, Kwiatkowski, Tom, Nikolaev, Vitaly, and Palomaki, Jennimaria. Tydi QA: A benchmark for information-seeking question answering in typologically diverse languages. *CoRR*, abs/2003.05002, 2020. URL https://arxiv.org/abs/2003.05002.

Clark, Kevin, Khandelwal, Urvashi, Levy, Omer, and Manning, Christopher D. What does bert look at? an analysis of bert's attention. *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2019. doi: https://doi.org/10.18653/v1/w19-4828. URL https://aclanthology.org/W19-4828/.

Conneau, Alexis, Khandelwal, Kartikay, Goyal, Naman, Chaudhary, Vishrav, Wenzek, Guillaume, Guzmán, Francisco, Grave, Edouard, Ott, Myle, Zettlemoyer, Luke, and Stoyanov, Veselin. Unsupervised cross-lingual representation learning at scale, 2019. URL https://arxiv.org/abs/1911.02116.

Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL https://arxiv.org/abs/1810.04805.

Forbes, Maxwell and Choi, Yejin. Verb physics: Relative physical knowledge of actions and objects. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 266–276, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1025. URL https://aclanthology.org/P17-1025.

Goel, Pranav, Feng, Shi, and Boyd-Graber, Jordan. How pre-trained word representations capture commonsense physical comparisons. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pp. 130–135, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-6016. URL https://aclanthology.org/D19-6016.

Habernal, Ivan, Wachsmuth, Henning, Gurevych, Iryna, and Stein, Benno. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1930–1940, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1175. URL https://aclanthology.org/N18-1175.

Hara, Kotaro, Adams, Abi, Milland, Kristy, Savage, Saiph, Callison-Burch, Chris, and Bigham, Jeffrey P. A data-driven analysis of workers' earnings on amazon mechanical turk. *CoRR*, abs/1712.05796, 2017. URL http://arxiv.org/abs/1712.05796.

Hara, Kotaro, Adams, Abigail, Milland, Kristy, Savage, Saiph, Hanrahan, Benjamin V., Bigham, Jeffrey P., and Callison-Burch, Chris. Worker demographics and earnings on amazon mechanical turk: An exploratory analysis. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, pp. 1–6, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359719. doi: 10.1145/3290607.3312970. URL https://doi.org/10.1145/3290607.3312970.

Hu, Junjie, Ruder, Sebastian, Siddhant, Aditya, Neubig, Graham, Firat, Orhan, and Johnson, Melvin. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080, 2020. URL https://arxiv.org/abs/2003.11080.

Kudo, Taku and Richardson, John. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012.

Levesque, Hector, Davis, Ernest, and Morgenstern, Leora. *The Winograd Schema Challenge*. International Conference on Principles of Knowledge Representation and Reasoning, 2012. URL https://cdn.aaai.org/ocs/4492/4492-21843-1-PB.pdf.

Lewis, Martin W. and Wigen, Karen E., 1997. URL https://archive.nytimes.com/www.nytimes.com/books/first/l/lewis-myth.html?_r=1&oref=slogin.

Lewis, Mike, Liu, Yinhan, Goyal, Naman, Ghazvininejad, Marjan, Mohamed, Abdelrahman, Levy, Omer, Stoyanov, Ves, and Zettlemoyer, Luke. Bart: Denoising

sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019. URL https://arxiv.org/abs/1910.13461.

Liang, Yaobo, Duan, Nan, Gong, Yeyun, Wu, Ning, Guo, Fenfei, Qi, Weizhen, Gong, Ming, Shou, Linjun, Jiang, Daxin, Cao, Guihong, Fan, Xiaodong, Zhang, Ruofei, Agrawal, Rahul, Cui, Edward, Wei, Sining, Bharti, Taroon, Qiao, Ying, Chen, Jiun-Hung, Wu, Winnie, Liu, Shuguang, Yang, Fan, Campos, Daniel, Majumder, Rangan, and Zhou, Ming. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6008–6018, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.484. URL https://aclanthology.org/2020.emnlp-main.484.

Lin, Bill Yuchen, Lee, Seyeon, Khanna, Rahul, and Ren, Xiang. Birds have four legs?! numersense: Probing numerical commonsense knowledge of pre-trained language models, 2020. URL https://arxiv.org/abs/2005.00683.

Lin, Bill Yuchen, Lee, Seyeon, Qiao, Xiaoyang, and Ren, Xiang. Common sense beyond english: Evaluating and improving multilingual language models for commonsense reasoning. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021. doi: https://doi.org/10.18653/v1/2021.acl-long.102. URL https://aclanthology.org/2021.acl-long.102/.

Liu, Yinhan, Ott, Myle, Goyal, Naman, Du, Jingfei, Joshi, Mandar, Chen, Danqi, Levy, Omer, Lewis, Mike, Zettlemoyer, Luke, and Stoyanov, Veselin. Roberta: A robustly optimized bert pretraining approach, 2019. URL https://arxiv.org/abs/1907.11692.

Liu, Yinhan, Gu, Jiatao, Goyal, Naman, Li, Xian, Edunov, Sergey, Ghazvininejad, Marjan, Lewis, Mike, and Zettlemoyer, Luke. Multilingual denoising pre-training for neural machine translation, 2020. URL https://arxiv.org/abs/2001.08210.

Mikolov, Tomás, Le, Quoc V., and Sutskever, Ilya. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168, 2013. URL http://arxiv.org/abs/1309.4168.

Mishra, Swaroop, Mitra, Arindam, Varshney, Neeraj, Sachdeva, Bhavdeep, and Baral, Chitta. Towards question format independent numerical reasoning: A set of prerequisite tasks, 2020. URL https://arxiv.org/abs/2005.08516.

Niu, Jingcheng, Lu, Wenjie, and Penn, Gerald. Does BERT rediscover a classical NLP pipeline? In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 3143–3153, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.278.

Olah, Chris and Carter, Shan. Attention and augmented recurrent neural networks. *Distill*, 2016. doi: 10.23915/distill.00001. URL http://distill.pub/2016/augmented-rnns.

Papineni, Kishore, Roukos, Salim, Ward, Todd, and Zhu, Wei-Jing. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia,

Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Pascanu, Razvan, Mikolov, Tomas, and Bengio, Yoshua. On the difficulty of training recurrent neural networks, 2013. URL https://arxiv.org/abs/1211.5063.

Passonneau, Rebecca, Habash, Nizar, and Rambow, Owen. Inter-annotator agreement on a multilingual semantic annotation task. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy, May 2006. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2006/pdf/634_pdf.pdf.

Pavlick, Ellie, Post, Matt, Irvine, Ann, Kachaev, Dmitry, and Callison-Burch, Chris. The language demographics of Amazon Mechanical Turk. *Transactions of the Association for Computational Linguistics*, 2:79–92, 2014. doi: 10.1162/tacl_a_00167. URL https://aclanthology.org/Q14-1007.

Petroni, Fabio, Rocktäschel, Tim, Riedel, Sebastian, Lewis, Patrick, Bakhtin, Anton, Wu, Yuxiang, and Miller, Alexander. Language models as knowledge bases? *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. doi: https://doi.org/10.18653/v1/d19-1250. URL https://aclanthology.org/D19-1250/.

Ponti, Edoardo Maria, Glavaš, Goran, Majewska, Olga, Liu, Qianchu, Vulić, Ivan, and Korhonen, Anna. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2362–2376, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.185. URL https://aclanthology.org/2020.emnlp-main.185.

Raffel, Colin, Shazeer, Noam, Roberts, Adam, Lee, Katherine, Narang, Sharan, Matena, Michael, Zhou, Yanqi, Li, Wei, and Liu, Peter J. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019. URL https://arxiv.org/abs/1910.10683.

Schuster, Simon, 2014. URL https://www.simonandschuster.com/books/Who-Owns-the-Future/Jaron-Lanier/9781451654974.

Semuels, Alana. The atlantic, Jan 2018. URL https://www.theatlantic.com/business/archive/2018/01/amazon-mechanical-turk/551192/.

Shliazhko, Oleh, Fenogenova, Alena, Tikhonova, Maria, Mikhailov, Vladislav, Kozlova, Anastasia, and Shavrina, Tatiana. mgpt: Few-shot learners go multilingual, 2022. URL https://arxiv.org/abs/2204.07580.

Snover, Matthew, Dorr, Bonnie, Schwartz, Rich, Micciulla, Linnea, and Makhoul, John. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pp. 223–231, Cambridge, Massachusetts, USA, August 8-12 2006. Association for Machine Translation in the Americas. URL https://aclanthology.org/2006.amta-papers.25.

Talmor, Alon, Elazar, Yanai, Goldberg, Yoav, and Berant, Jonathan. olmpics - on what language model pre-training captures. *CoRR*, abs/1912.13283, 2019. URL http://arxiv.org/abs/1912.13283.

Tenney, Ian, Das, Dipanjan, and Pavlick, Ellie. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452.

Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N, Kaiser, Lukasz, and Polosukhin, Illia. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

Wallace, Eric, Wang, Yizhong, Li, Sujian, Singh, Sameer, and Gardner, Matt. Do NLP models know numbers? probing numeracy in embeddings. 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)*, pp. 5307–5315, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1534. URL https://aclanthology.org/D19-1534.

Wang, Cunxiang, Liang, Shuailong, Zhang, Yue, Li, Xiaonan, and Gao, Tian. Does it make sense? and why? a pilot study for sense making and explanation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4020–4026, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1393. URL https://aclanthology.org/P19-1393.

Wei, Jason, Wang, Xuezhi, Schuurmans, Dale, Bosma, Maarten, Chi, Ed H., Le, Quoc, and Zhou, Denny. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903, 2022. URL https://arxiv.org/abs/2201.11903.

Wu, Yonghui, Schuster, Mike, Chen, Zhifeng, Le, Quoc V., Norouzi, Mohammad, Macherey, Wolfgang, Krikun, Maxim, Cao, Yuan, Gao, Qin, Macherey, Klaus, Klingner, Jeff, Shah, Apurva, Johnson, Melvin, Liu, Xiaobing, Kaiser, Lukasz, Gouws, Stephan, Kato, Yoshikiyo, Kudo, Taku, Kazawa, Hideto, Stevens, Keith, Kurian, George, Patil, Nishant, Wang, Wei, Young, Cliff, Smith, Jason, Riesa, Jason, Rudnick, Alex, Vinyals, Oriol, Corrado, Greg, Hughes, Macduff, and Dean, Jeffrey. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL http://arxiv.org/abs/1609.08144.

Xue, Linting, Constant, Noah, Roberts, Adam, Kale, Mihir, Al-Rfou, Rami, Siddhant, Aditya, Barua, Aditya, and Raffel, Colin. mt5: A massively multilingual pre-trained text-to-text transformer, 2020. URL https://arxiv.org/abs/2010.11934.

Zaidan, Omar F. and Callison-Burch, Chris. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 1220–1229, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1122.

Zellers, Rowan, Bisk, Yonatan, Schwartz, Roy, and Choi, Yejin. SWAG: A large-scale adversarial dataset for grounded commonsense inference. *CoRR*, abs/1808.05326, 2018. URL http://arxiv.org/abs/1808.05326.

Zellers, Rowan, Holtzman, Ari, Bisk, Yonatan, Farhadi, Ali, and Choi, Yejin. Hellaswag: Can a machine really finish your sentence? *CoRR*, abs/1905.07830, 2019. URL http://arxiv.org/abs/1905.07830.

Zhou, Xuhui, Zhang, Yue, Cui, Leyang, and Huang, Dandan. Evaluating commonsense in pre-trained language models. *CoRR*, abs/1911.11931, 2019. URL http://arxiv.org/abs/1911.11931.

Appendix A

Models

A.1 Hyperparamaters

	learning rate	epochs	batch size
bert-base-multilingual-uncased	5e-5	3	16
xlm-roberta-base	5e-5	3	16
xlm-roberta-large	5e-5	3	16
mt5-small	3e-4	10	4
mt5-base	3e-4	10	4
mt5-large	3e-4	10	4
mbart-large-cc25	5e-5	3	4

A.2 Number list

English (all lists follow same order):

```
one,
two,
three,
four,
five,
six,
seven,
eight,
nine,
ten,
zero, no
```

Chinese:

```
一
二, 两
```

三,
四,
五,
六,
七,
八,
九,
十,
零, 无, 没, 没有, 不含, 毫无

Russian:
один, одно, одна, одни, одного, одной, одних, одному, одним, одну, одною, одними, одном
два, две, двух, двум, двумя, двое
три, трех, трем, тремя, трое, троих, троим, троими, тройки, тройку
четыре, четырех, четырьмя, четырем, четверо, четверых, четверым, четверыми, четвертом, четвертой, четырём
пять, пяти, пятью, пятеро, пятерых, пятерым, пятерыми, пятых
шесть, шести, шестью, шестеро, шестерых, шестерым, шестерыми
семь, семи, семью, семеро, семерых, семерым, семерыми
восемь, восьми, восьмью, восемью, восьмеро, восьмерых, восьмерым, восьмерыми
девять, девяти, девятью, девятеро, девятерых, девятерым, девятерыми
десять, десяти, десятью, десятеро, десятерых, десятерым, десятерыми,
никакие, нуля, ноль , нулю, ноля, нулевой, нулевым , нулевая, нулевую, нет,
не, без, ни, 'нуле', 'нулевое', 'нулевого', 'Никакие', 'никаких'

Arabic:
واحد، واحدة، واحدي، الواحد، الواحدة، الواحدي، وواحد، وواحدة، وواحدي، لواحد، لواحدة، لواحدي، واحدًا، بواحد، بواحدة، بواحدي، كواحد، كواحدة، كواحدي

ثنين، اثنان، اثنتين، اثنتان، اثنا ،اثنتا

ثلاث، ثلاثة، ثلاثي، الثلاث، الثلاثة، الثلاثي، وثلاث، وثلاثة، وثلاثي، لثلاث، لثلاثة، لثلاثي، بثلاث، بثلاثة، بثلاثي، كثلاث، كثلاثة، كثلاثي

أربع، أربعة، أربعي، الأربع، الأربعة، الأربعي، وأربع، وأربعة، وأربعي، لأربع، لأربعة، لاربعي، بأربع، بأربعة، بأربعي، كأربع، كأربعة، كأربعي

خمس، خمسة، خمسي، الخمس، الخمسة، الخمسي، وخمس، وخمسة، وخمسي، لخمس، لخمسة، لخمسي، بخمس، بخمسة، بخمسي، كخمس، كخمسة، كخمسي،

ست، ستة، ستي، الست، الستة، الستي، وست، وستة، وستي، لست، لستة، لستي، بست، بستة، بستي، كست، كستة، كستي

سبع، سبعة، سبعي، السبع، السبعة، السبعي، وسبع، وسبعة، وسبعي، لسبع، لسبعة، لسبعي، بسبع، بسبعة، بسبعي، كسبع، كسبعة، كسبعي

ثمان، ثمانية، ثماني، الثمان، الثمانية، الثماني، وثمان، وثمانية، وثماني، لثمان، لثمانية، لثماني، بثمان، بثمانية، بثماني، كثمان، كثمانية، كثماني

تسع، تسعة، تسعي، التسع، التسعة، التسعي، وتسع، وتسعة، وتسعي، لتسع، لتسعة، لتسعي، بتسع، بتسعة، بتسعي، كتسع، كتسعة، كتسعي

عشر، عشرة، عشري، العشر، العشرة، العشري، وعشر، وعشرة، وعشري، لعشر، لعشرة، لعشري، بعشر، بعشرة، بعشري، كعشر، كعشرة، كعشري،

لا، كلا، أي، بدون، ليس، عدم، صفر

A.3 Declension list

Nominal: ноль,нулевои, нулевая, нулевое, Никакие, один, одно, одна, одни, три, троики, трое, четыре, четверо, пять, пятеро, шесть, шестро, семь, семеро, восемь, восьмеро, девять, девятеро, десять, десятеро
Accusative: ноль, нулевую, один, одно, одну, одни, два, две, двое, три, троику, трое, четыре, четверо, пять, пятеро, шесть, шестеро, семь, семеро, восемь, восьмеро, девять, девятеро, десять, десятеро
Genitive: нуля, ноля, Нулевого, нулевои, никаких, одного, однои, одних, двух, трех, троики, троих, четырех, четверых, четвертои, пяти, пятерых, пятых, шести, шестерых, семи, семерых, восьми, восьмерых, девяти, девятерых, десяти, десятерых
Dative нулю, нулевои, одному однои, одним, двум, трем, троим, четырем, четырем, четверым, четвертои, пяти, пятерым, шести, шестерым, семи, семерым, восьми, восьмерым девяти, девятерым, десяти, десятерым
Instrumental: нулевои, нулевым, одним, однои, одною, одними, двумя, тремя, троими, четырьмя, четверыми, четвертои, пятью, пятерыми, шестью, шестерыми, семью, семерыми, восемью, восьмью, восьмерыми, десятью, девятерыми,

десятью, десятерыми
Prepositional: нуле, нулевои, никаких, одном, однои, одних, двух, трех, троих, четырех, четверых, четвертом, четвертои, пяти, пятерых, пятых, шести, шестерых, семи, семерых, восьми, восьмерых, девяти, девятерых, десяти, десятерых
Other: нет, не, без, ни

Appendix B

MTurk

B.1 Crowdsourcing instructions

```
    <head>
  <meta charset="UTF-8">
</head>

<p><strong>Eligibility</strong>:</p>
<ul>
    <li>You must be a<span style="color: #ff0000;"> native speaker of Russian</
    <li>You must be<span style="color: #ff0000;"> proficient in English</span>
</ul>

<p><strong>Guidance:</strong></p>

<ul>
    <li>Sentences will have bracketing, when you translate, this should be main
    <li>Please translate the sentence into a Russian sentence which is close to
    <li>When given a number in written form, please translate it into its writt
</ul>

<p><strong>Informed consent</strong>:</p>
<p>This is a linguistic experiment performed at the University of Edinburgh. If
<p><strong>Feedback</strong>:</p>
<p>We are happy to receive feedback and improve this job accordingly. Feel free
```

B.2 MTurk participants' consent form

V1 07/06/21

STUDY NAME: English to Arabic Human Translation with Native Speakers

WHAT IS THE PURPOSE OF THIS STUDY AND WHAT WILL I BE ASKED TO DO?

This study is being run by researchers at the University of Edinburgh. The purpose of the study is to transl of statements from English into Arabic. You've been invited to take part because you are located in Egypt, Morroco or the United States of America.

If you decide to take part, you will see 16 sentences in English and you will need to write equivalent sente translated into Arabic. This task should take approximately 5 minutes and you will be compensated throug Amazon Mechanical Turk platform.

There are no anticipated risks associated with participation.

USE OF YOUR DATA

In addition to your responses, you will be asked to provide information about your language background. your age, country and the number of years speaking the relevant language. Worker IDs will also be stored compliance with GDPR, no personal data which could be used to identify you will be collected.

The anonymised data will be publicly released for research purposes.

WHAT IF I WANT TO WITHDRAW FROM THE STUDY?

You can leave the study at any time through the Mechanical Turk platform or through contacting the ema this case, all your data from this study will be deleted.

WHO CAN I CONTACT WITH QUESTIONS OR CONCERNS?

If you have questions about the study, please contact the lead researcher, Dayyán O'Brien by emailing D.O 1@sms.ed.ac.uk Please note that this may expose your personal email address to the research team. In co with GDPR, all emails from participants will be deleted following the end of the study. If you wish to make about the study, please contact Professor Mirella Lapata by email: mlap@inf.ed.ac.uk. If you have any cor research team cannot resolve to your satisfaction, please contact inf-ethics@inf.ed.ac.uk, giving the study

I understand that my anonymised data will be publicly released.

<select box> Yes/No

I understand that I can withdraw from the study at any point without
giving a reason.

<select box> Yes/No

If you understand the task and wish to participate in the study, please select "Yes, I will participate"; if not not participate."

<select box> Yes I will participate / No I will not participate

This study was certified according to the Informatics Research Ethics Process, RT number 6800

Appendix C

In-house

C.1 Crowdsourcing instructions

```
    <h2><span style="color: rgb(0, 0, 0);">Translate Statements from English to

<p>Translate <strong>all sentences into Russian. There will be a maximum of 16 s

<p>You must be a<span style="color: #ff0000;"> native speaker of</span> <span st

<p>Please attempt to translate every word into Russian. If this is difficult <st

<p><strong>Guidance:</strong></p>

<ul>
<li>You will be given a sentence in English, and its equivalent in Google Transl
<li>Sentences will have bracketing, when you translate, this should be maintaine
<li>Please translate the sentence into a Russian sentence which is close to how
<li>When given a number in written form, please translate it into its <strong>wr
</ul>

<div style="color:blue">
<h3>Example Translations</h3>
</div>

<table style="border: none; width:90%;">
<thead>
<tr>
<th style="width:90%;">
<div style="color:blue">
<h4>Source sentence in English (EN) and translation into Russian (RU)</h4>
</div>
</th>
</tr>
```

```
</thead>
<tbody>
<tr>
<td id="a" style="border: none; width:75%; text-align:justify;">EN1 Goats are [
<td style="border: none; width:11%;"> </td>
</tr>
<tr>
<td id="a" style="border: none; width:75%; text-align:justify;color:blue">RU1
<td style="border: none; width:11%;"> </td>
</tr>
<tr>
<td id="b" style="border: none; width:75%; text-align:justify;">EN2 Roses have
<td style="border: none; width:11%;"> </td>
</tr>
<tr>
<td id="a" style="border: none; width:75%; text-align:justify;color:blue">RU2
<td style="border: none; width:11%;"> </td>
</tr>
<tr>
<td id="c" style="border: none; width:75%; text-align:justify;">EN3 A cat has [
<td style="border: none; width:11%x;"> </td>
</tr>
<tr>
<td id="a" style="border: none; width:75%; text-align:justify;color:blue">RU3
<td style="border: none; width:11%;"> </td>
</tr>
</tbody>
</table>
</div>

<p><strong>This study was certified according to the Informatics Research Ethic
</div>
<span class="init-display-hidden" id="keybinding-info">Press &quot;Click to beg
```

C.2 Participants' consent form

V1 07/06/21

STUDY NAME: English to Arabic Human Translation with Native Speakers

WHAT IS THE PURPOSE OF THIS STUDY AND WHAT WILL I BE ASKED TO DO?

This study is being run by researchers at the University of Edinburgh. The purpose of the study is to transl of statements from English into Arabic. You've been invited to take part because you are located in Egypt, Morroco or the United States of America.

If you decide to take part, you will see 16 sentences in English and you will need to write equivalent sente translated into Arabic. This task should take approximately 5 minutes and you will be compensated throug Amazon Mechanical Turk platform.

There are no anticipated risks associated with participation.

USE OF YOUR DATA

In addition to your responses, you will be asked to provide information about your language background. your age, country and the number of years speaking the relevant language. Worker IDs will also be stored compliance with GDPR, no personal data which could be used to identify you will be collected.

The anonymised data will be publicly released for research purposes.

WHAT IF I WANT TO WITHDRAW FROM THE STUDY?

You can leave the study at any time through the Mechanical Turk platform or through contacting the ema this case, all your data from this study will be deleted.

WHO CAN I CONTACT WITH QUESTIONS OR CONCERNS?

If you have questions about the study, please contact the lead researcher, Dayyán O'Brien by emailing D.O 1@sms.ed.ac.uk Please note that this may expose your personal email address to the research team. In c with GDPR, all emails from participants will be deleted following the end of the study. If you wish to make about the study, please contact Professor Mirella Lapata by email: mlap@inf.ed.ac.uk. If you have any cor research team cannot resolve to your satisfaction, please contact inf-ethics@inf.ed.ac.uk, giving the study

I understand that my anonymised data will be publicly released.

<select box> Yes/No

I understand that I can withdraw from the study at any point without
giving a reason.

<select box> Yes/No

If you understand the task and wish to participate in the study, please select "Yes, I will participate"; if not not participate."

<select box> Yes I will participate / No I will not participate

This study was certified according to the Informatics Research Ethics Process, RT number 6800