Comparing core concept categorisation models in geo-analytic questions

Author: Z.S. Wiersma (4237927)
Supervisors: S. Scheider & H. Xu

Master Applied Data Science, Utrecht University
01-07-2022

Abstract. Current question answering (QA) systems lack the ability to provide answers to geo-analytical questions. Geo-analytical questions must be interpreted to know what relevant data and geographical tools require to be used to provide an answer. This study focused on core concept categorisation, which is the first step in developing the aforementioned system. Named-entity recognition, in combination with transformer-based models BERT and RoBERTa, is applied to categorise core concepts in geo-analytical questions. Synonym replacement, a simple data augmentation technique, is applied to overcome data scarcity and results of both models are compared. RoBERTa has a better performance on the original data set and BERT has a better performance on the augmented data set. Both models presented significant improvements when applying synonym replacement. Results of this study can be applied to further develop a geo-analytical QA system.

Keywords: named-entity recognition; geo-analytical questions; core concepts categorisation; deep learning models; BERT; RoBERTa; data augmentation

1 Introduction

Being able to develop a system which can provide answers to geo-analytical questions has a large potential for (data) scientists. As an example, such tools can help health scientists in researching potential causes of people developing Parkinson's disease. A number of animal studies suggest an exposure to pesticides may increase the risk of developing Parkinson's disease (Wang et al., 2011). Health scientists might not be able to directly answer this research question without processing the available data with geographical tools. Useful maps, such as the location of addresses of people with Parkinson's disease and the location of agricultural fields, could be generated to support the hypothesis. However, the application of this

method can be difficult, as generating maps using geographical tools often requires highly-trained people (Goldin and Rudahl, 1997). In this example, health scientists would be helped tremendously if they had the opportunity to ask a system geo-analytical questions in a natural language.

Current QA systems already fulfil a similar need. However, they lack the ability to provide answers to geo-analytical questions (Scheider et al., 2021). Current QA systems construct their answers based on factoid knowledge and cannot provide answers to questions that require being analysed. For example, given question "Who is the prime minister of The Netherlands?", current QA systems will be able to extract a correct and concise answer from relevant documents or from a knowledge base. With a geo-analytical question, such as "What is the average distance between the location of addresses of people with Parkinson's disease and the closest agricultural field?", QA systems cannot directly extract an answer from external sources. In order to provide an answer to this geo-analytical question, a system must interpret the question and generate the requested answer by using relevant data and geographical tools.

Extracting relevant information from a question and converting that information to underlying concepts of spatial analysis is something analysts implicitly do. Previous studies call these underlying concepts core concepts of spatial information (Kuhn, 2012). Core concepts provide the ability to interpret the question and derive relevant data and geographical tools. To illustrate, in the previous example "location of addresses" can be categorised as core concept *toponymy*, "people with Parkinson's disease" can be categorised as core concept *object*, "agricultural field" can be categorised as core concept *nominal field*, and "average distance" can be categorised as core concept *content amount of field at ratio level*. By explicitly knowing the categorised core concepts in a question, a system will be able to find relevant data and geographical tools required to generate an answer.

This study focused on the first step of developing a system which can answer geo-analytical questions: core concept categorisation. There are several ways to assign a core concept to a phrase. One way to categorise phrases in questions is by creating a dictionary containing a phrase and the corresponding core concept. A dictionary-based method is not a favourable solution, as there are many ways to formulate a sentence and each variation must be added to the dictionary. Moreover, a dictionary-based method might falsely categorise certain phrases when they have multiple related meanings. A better approach to assigning core concepts to phrases is to use deep learning techniques with contextual information. This method captures the essence of an entire sentence in order to categorise a certain phrase. Furthermore, recent deep learning models are pre-trained on large data sets, which is beneficial for its performance (Qiu et al., 2020). For this reason, deep learning techniques are accurate and relatively easy to implement (Hestness et al., 2019).

In recent years, many deep learning techniques have achieved success in natural language processing (NLP) tasks. One of those deep learning techniques is Bidirectional Encoder Representations from Transformers

(BERT), a deep learning model developed by Google in 2018 (Devlin et al., 2018). BERT has become the baseline in NLP experiments due to its performance (Rogers et al., 2020). Also, many studies analysed BERT and proposed numerous improvements. An example of an improvement on the existing BERT model is Robustly optimized BERT approach (RoBERTa), developed by Facebook in 2019, due to BERT being significantly undertrained (Liu et al., 2019).

Both BERT and RoBERTa can be applied to train a core concept categorisation model, yet, there are no previous studies comparing the results of these two models in core concept categorisation (Liang et al., 2020). This study focused on building a core concept categorisation model using BERT and RoBERTa. The research question for this study is shown below.

How does the performance of transformer-based models BERT and RoBERTa compare in the context of core concept categorisation in geo-analytical questions?

A difficulty of this study is data scarcity. To overcome data limitations, synonym replacement, a data augmentation technique, was applied which boosts the performance of a model (Wei and Zou, 2019). A sub-question for this study is formulated below.

What is the effect on the results of transformer-based models BERT and RoBERTa when applying synonym replacement in the context of core concept categorisation in geo-analytical questions?

This paper is structured as follows: section 2 introduces related work, section 3 proposes the selected data, data preparation needed for analysis, and analysis procedure, section 4 provides the obtained results, section 5 discusses all major findings, and finally, section 6 provides a conclusion.

2 Related work

Categorisation of phrases is already an existing task of information extraction. It is mostly known as named-entity recognition (NER), but can also be referred to as entity extraction, entity chunking, or entity identification (Mohit, 2014). The following subsections provide more information about NER, how current deep learning techniques are used to understand context in natural language, and suggestions on improving deep learning models.

2.1 NER

NER is the process of identifying phrases in unstructured text and categorising the identified phrases into predefined categories. The categories can be generic, such as *person* or *toponymy*, but can be customised for specific applications as well, such as core concepts *object* and *field*.

NER identifies phrases in a sentence that belong together and categorises them. For example, in the sentence "James lives in The Netherlands.", "James" can be categorised as *person* and "The Netherlands" can be categorised as *toponymy*. This is similar to what is needed in categorising core concepts in geo-analytical questions. A study by Eftimov et al. (2017) suggested three different approaches to identifying phrases: dictionary-based, rule-based, or corpus-based methods. Dictionary-based methods match phrases to categories that exist in the dictionary. To improve the performance of a dictionary-based method, one could use several techniques by using variations of phrases. For example, the size of the dictionary will expand by using synonymy (different words have a similar meaning, i.e. "big" and "huge") or hyponymy (different words sharing the same generic supertype, i.e. "red" and "blue"). Despite having techniques to improve the performance of a dictionary-based method, it still requires maintenance and is not easily scalable. Rule-based methods use predefined rules to categorise phrases, for example, by using regular expressions, which can be used to find character combinations in phrases. Equally, this method is not preferred since developing each rule requires domain knowledge. Lastly, corpus-based methods are annotated corpora provided by domain experts. Having an extensive annotated corpora could arguably be the best solution to NER since each variation can be captured in this corpora. However, creating an annotated corpora encompassing all phrases is a time-consuming task for domain experts.

Recent advances in deep learning, such as the introduction of Convolutional Neural Networks, Recurrent Neural Networks, and Long Short Term Memory Networks, create the opportunity to improve NER using deep learning techniques (Minar and Naher, 2018). A study from 2019 concluded having a better performance on deep learning techniques compared to the time-consuming methods described above (Yadav and Bethard, 2019). Many studies suggest using a BERT-based approach for training application-specific categories, such as in the biomedical field or in cybersecurity (Hakala and Pyysalo, 2019; Tikhomirov et al., 2020). Other studies comparing several deep learning techniques in NER achieved the best performance by using a RoBERTa-based approach (Wang et al., 2020). Despite the (small) differences in performance among the deep learning techniques, all deep learning techniques achieved good results and should be considered for training NER-related tasks.

In a NER pipeline where deep learning techniques are used, phrases are often tagged in order to emphasise words that belong together in a phrase. The most popular format is beginning-inside-outside (BIO) tagging, also referred to as IOB tagging. A prefix of *B-* followed by a category indicates

a word being the beginning of a categorised phrase, a prefix of *I-* followed by a category indicates a word being inside a categorised phrase, and an *O* indicates a word not being categorised. An example of a BIO-tagged sentence is shown in table 1.

James	**lives**	**in**	**The**	**Netherlands**	**.**
B-person	*O*	*O*	*B-location*	*I-location*	*O*

Table 1: Example of a BIO-tagged sentence

By applying BIO tagging, a deep learning model can specifically be trained with words in a phrase belonging to one category, such as "The" and "Netherlands" belonging to the category *toponymy* in the example above.

2.2 Transformer

A transformer is a deep learning technique introduced in 2017 by Google (Vaswani et al., 2017) which is mostly applied in the field of NLP. This deep learning technique has an encoder-decoder architecture. In short, the encoder is responsible for understanding the input and the decoder is responsible for generating an output based on the understanding of the encoder. The encoder uses a number of encoding layers to computationally produce a numerical interpretation based on the given input. The decoder uses the same number of decoding layers to computationally produce an output based on the numerical interpretation of the encoder on each decoding layer. A schematic overview of this process is shown in figure 1.

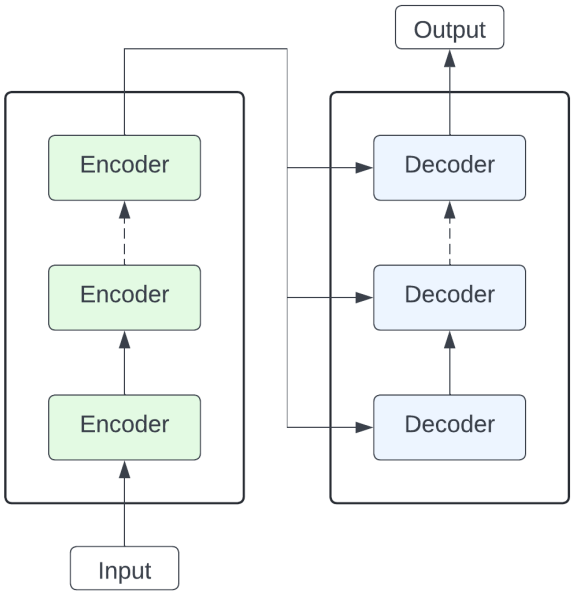


Figure 1: Schematic overview of a transformer's encoder-decoder architecture

A feature of a transformer is, for instance, the use of a self-attention mechanism. Both the encoding and the decoding layers make use of this mechanism. Self-attention is the process of understanding the relations between words in a sentence. Consider the sentence "The house collapsed because it was set on fire.". With a self-attention mechanism, a transformer is able to make an association between the words "it" and "house". Making associations between words is essential to understand the context of a sentence, since language by nature contains many ambiguities.

Another key feature of transformers is the ability to process the entire input simultaneously, making parallelisation possible when training a model. As a result, larger data sets can be used to train a model in a relatively short time frame.

2.3 BERT

After the introduction of transformers, language model BERT was developed to understand the representation of language by using the encoder of a transformer. In order to understand the statistical relationships of words, BERT was pre-trained on a large unlabelled text corpus: 800 million words from BooksCorpus and 2.500 million words from English Wikipedia, which took around four days to train and energy costs were estimated at $7.000 (Devlin et al., 2018; Schwartz et al., 2020). The pre-trained model is publicly available and can be fine-tuned to work for a specific task. Fine-tuning is done by providing labelled data and feeding it

to one additional output layer on top of the output layers of the pre-trained model (Devlin et al., 2018). Fine-tuning is considerably faster compared to the pre-training process, since fine-tuning can be done on a relatively small data set. Therefore, having access to the pre-trained BERT model reduces computation costs significantly.

The fine-tuning process consists of a few hyperparameters which are used to control the learning process. For example, batch size and number of epochs can be tuned with BERT. Batch size refers to the number of samples passed to the model simultaneously and epoch refers to one single pass of all samples to the model (Brownlee, 2018). For example, when having a batch size of 10 and a total of 1000 samples to be processed, one epoch has a total of 100 batches to process all samples at once.

Additionally, optimisers are used when fine-tuning deep learning models (Schneider et al., 2019). Optimisers are functions used to increase the accuracy of a model by changing attributes of a neural network, such as weights (Liu et al., 2021). The optimiser can be tuned with a hyperparameter called learning rate, which controls to what extent new information overwrites existing information (Murphy, 2012). Most studies suggest the use of the Adam optimiser, due to its performance and ease of implementation (Tato and Nkambou, 2018; Desai, 2020).

The optimal hyperparameter values are task-specific. However, it is suggested to use the following values when fine-tuning (Devlin et al., 2018):

- **Batch size:** 16, 32
- **Number of epochs:** between 2 and 4
- **Learning rate:** 5e-5, 3e-5, 2e-5

Larger data sets tend to be less sensitive to hyperparameter tuning compared to smaller data sets (Devlin et al., 2018). Due to low computational costs of fine-tuning, it is suggested to spend time on tuning hyperparameter values especially for smaller data sets.

2.4 RoBERTa

Facebook developed RoBERTa after the development of BERT and claims to achieve better results (Liu et al., 2019). The architecture of RoBERTa is comparable to the architecture of BERT. However, the process of pre-training is slightly different and the model was trained on a different unlabelled text corpus. The hyperparameter recommendations for RoBERTa are to some degree similar to those from the BERT model (Liu et al., 2019):

- **Batch size:** 16, 32
- **Number of epochs:** maximum of 10

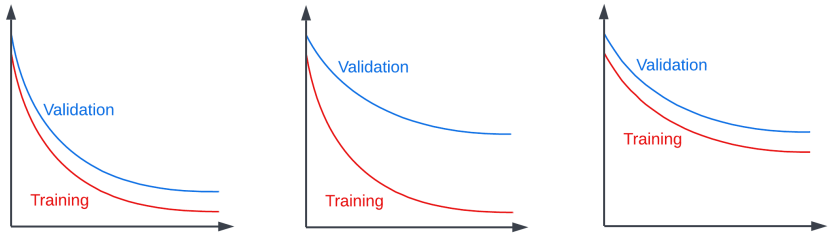
- **Learning rate:** 1e-5, 2e-5, 3e-5

A study about NER in 2020 achieved better performance using RoBERTa compared to other deep learning models, among which BERT (Wang et al., 2020). This study tuned its hyperparameters for all models to a batch size of 16, 2 epochs, and a learning rate of 5e-5.

2.5 Improving performance

Generalisation, which refers to a model's ability to understand unseen data, is one of the major challenges when building a deep learning model (Bansal et al., 2021). A model's ability to generalise is essential for its success, since a model is built to adapt and react appropriately to unseen data. Overfitting is a fundamental modelling issue which occurs when a model is trained too well on the provided training data, however, it lacks the ability to generalise appropriately on unseen data (Ying, 2019). On the contrary, underfitting occurs when a model is not capable of capturing the variability of the training data (Jabbar and Khan, 2015). As a result, the model does not perform well on both training data and unseen data.

An approach to discovering a model's fit is by plotting the training loss and validation loss at each epoch during training (Shorten and Khoshgoftaar, 2019). Training loss is a metric used to assess the performance of a model on training data and validation loss is a metric used to assess the performance of a model on unseen data. It is desired to have a low training and validation loss (Belkin et al., 2019). Figure 2 shows three plots with the number of epochs on the x-axis and the model's loss on the y-axis. A desired convergence is shown in (a) of figure 2: both training loss and validation loss decrease considerably overtime and stabilise after a number of epochs. Signs of overfitting is shown in (b) of figure 2: training loss continues to decrease considerably while validation loss only decreases slightly overtime. Lastly, signs of underfitting is shown in (c) of figure 2: both training loss and validation loss only decrease slightly overtime.



(a) Desired convergence (b) Signs of overfitting (c) Signs of underfitting

Figure 2: Examples of training and validation losses

Furthermore, within research of classification, models are evaluated using performance metrics precision and recall (Gehanno et al., 2009).

Precision indicates the ability of a model to only predict relevant instances, whereas recall indicates the ability of a model to predict all relevant instances. The use of precision and recall comes with a trade-off. When a model always predicts a relevant instance, recall is 100% and precision is extremely low. To account for these competing metrics, the F1-score is calculated using precision and recall to sum up the overall accuracy of a model (Crestani and van Rijsbergen, 1995). An F1-score can hold a value between 0 (either precision or recall is 0%) and 1 (precision and recall are 100%).

Several studies proposed data augmentation techniques to solve underfitting and overfitting, and improve performance (Shorten and Khoshgoftaar, 2019; Aquino et al., 2017). Data augmentation is the practice of artificially increasing training data by generating synthetic data (Summers and Dinneen, 2019). Currently, data augmentation in the field of computer vision (high-level understanding of digital images) requires less effort compared to data augmentation in NLP (Shorten et al., 2021). A data set of images can effortlessly be increased with label-preserving transformations on existing images. For example, an image of a house is still a house after flipping or rotating the image. Meanwhile, preserving the semantics in NLP is more complicated when applying similar transformations.

A study by Wei and Zou (2019) proposed a set of simple data augmentation techniques for NLP. The suggested techniques are synonym replacement, random insertion, random swap, and random deletion. An example of each simple data augmentation technique is shown in table 2.

Data augmentation technique	**Example**
None (original input)	A cat is born.
Synonym replacement	A kitten is born.
Random insertion	A cat is country born.
Random swap	A born is cat.
Random deletion	A is born.

Table 2: Schematic overview of used materials and methods

While these simple data augmentation techniques do not always preserve semantics, the study concluded that it has a large performance boost for small data sets (Wei and Zou, 2019). Larger data sets tend to be less sensitive to data augmentation techniques since models tend to already generalise properly on larger data sets without applying data augmentation techniques (Wei and Zou, 2019).

A study by Kobayashi (2018) proposed a more sophisticated technique: contextual augmentation. The proposed method is similar to synonym replacement. However, the difference is the consideration of contextual information. With contextual augmentation, inappropriate replacements, for example in the case of homonyms (same word, but different meanings), are less likely to happen. For example, sentence “He was mean to his sister.” can be replaced with “He was average to his sister.” when using synonym

replacement without contextual information. Despite the use of contextual information, only marginal improvements were observed compared to synonym replacement without contextual information (Kobayashi, 2018).

3 Materials and methods

This section describes the process of this research in more detail. First, data was augmented by using the provided training data and a core concept dictionary. Then, the augmented training data, the original training data, and test data were pre-processed. The fine-tuning process of BERT and RoBERTa used the pre-processed augmented training data and the pre-processed original training data separately. Finally, both models were evaluated using pre-processed test data. The schematic overview of this pipeline is shown in figure 3.

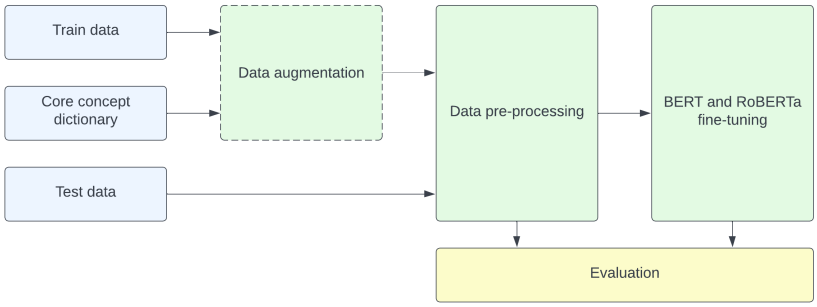


Figure 3: Schematic overview of used materials and methods

The blue blocks represent the used data sets, the green blocks represent the use of these data sets, and finally, the yellow block represents the evaluation of the fine-tuned models. The following subsections will explain all components in more detail.

3.1 Data sets

This study focused on core concept categorisation in geo-analytical questions. Several (digital) geographic information systems (GIS) textbooks were used to manually collect geo-analytical questions. The collected geo-analytical questions were applied as training and test data. Examples of used textbooks are *An Introduction to Geographical Information Systems*[1] and *GIS Tutorial 2: Spatial Analysis Workbook*[2]. 308 geo-analytical questions were collected from GIS textbooks and 134

[1] `https://abdn.pure.elsevier.com/en/publications/an-introduction-to-geographical-information-systems`

[2] `https://www.esri.com/news/releases/13-1qtr/gis-tutorial-2-spatial-analysis-workbook-101.html`

geo-analytical questions were collected from ArcGIS and QGIS online tutorials. A small subset of the data sets are shown in table 3.

Question
What is the percentage of residential areas inside 1 km area of the central station in Oleander
What is the Euclidean distance to tram stations in Amsterdam
What is the delivery cost from warehouse to stores within 10 minute driving time in Paris
...

Table 3: Small subset of input data

Experts in the domain of core concept categorisation manually created a core concept dictionary based on the input data. The dictionary contains phrases from the geo-analytical questions and their corresponding core concept. A small subset of this dictionary is shown in table 4.

Phrase	**Core concept**
percentage	PROPIR
residential areas	FLDN
central station	OBJ
...	...

Table 4: Small subset of core concept dictionary

The core concept dictionary contains a total value of 656 phrases and their corresponding categories. A total of 21 unique core concepts were identified.

3.2 Data augmentation

To increase the original data set, the core concept dictionary was enhanced with relevant noun-only synonyms using WordNet [3], a large lexical English database. The core concept of the original phrase was preserved for each synonym. For example, "share", a synonym of "percentage", was added to the core concept dictionary with core concept *PROPIR*. Similarly, the input data was increased with synonyms of each matched phrase from the original core concept dictionary. Permutations of multiple synonyms were not created. To illustrate, a small subset of the augmented input data, based on the first sentence of table 3, is shown in table 5.

[3] `https://wordnet.princeton.edu/`

Question
What is the percentage of residential areas inside 1 km area of the central station in Oleander
What is the share of residential areas inside 1 km area of the central station in Oleander
What is the portion of residential areas inside 1 km area of the central station in Oleander
...

Table 5: Small subset of augmented training data set

The number of questions in the augmented data set can be calculated by using the formula below, where n is the number of questions, s_i is the number of synonyms in question i, and S is the number of total variations.

$$\sum_{i=1}^{n} S = n + s_1 + s_2 + ... + s_i$$

After data augmentation, the data set contained a total of 4.180 questions (an increase of 3.872 questions compared to the original training data set). The original training data set and the augmented training data set were kept separate in order to evaluate the differences. Data augmentation was solely performed on the training data set.

3.3 Data pre-processing

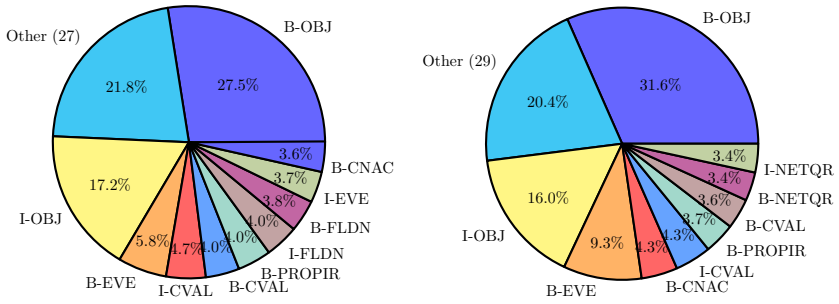
The input has to be pre-processed in order to satisfy the requirements of the fine-tuning process of BERT and RoBERTa. Both models expect a list of words and their corresponding BIO-tagged category. Categorisation of phrases is prioritised based on their word length. Some phrases might overlap with other phrases, such as "landscape conservation park" and "park" (both phrases exist in the core concept dictionary). A question containing phrase "landscape conservation park", the phrase will be categorised as core concept corresponding to "landscape conservation park" and not core concept corresponding to "park". Moreover, categorisation of locations is manually excluded from tagging as locations might contain phrases from the core concept dictionary ("United States" contains phrase "states").

The output of the first step of the pre-processing stage is shown in table 6. This table contains individual words from all questions and their corresponding category. A total of two tables were created: one table without augmented data and one table with augmented data. This allowed to build a total of four models (two BERT models and two RoBERTa models) and to evaluate each model separately.

Sentence	Word	Category
1	What	O
1	is	O
1	the	O
1	percentage	B-PROPIR
1	of	O
1	residential	B-FLDN
1	areas	I-FLDN
1	inside	O
...	...	...

Table 6: Small subset of the output of the pre-processing stage

Based on the information in the table above, a category distribution of both the original input data and the augmented input data was created and is shown in 4. Due to most words being categorised as *O*, this category was omitted to provide a better view on the distribution of core concepts.



(a) without data augmentation (b) with data augmentation

Figure 4: Category distribution on training data

The last step in the pre-processing stage is tokenising words and padding sentences. In transformer-based models BERT and RoBERTa, words are tokenised to keep the vocabulary size small. This is done by breaking words into smaller words. For example, "preferably" is split into "prefer" (a more common word) and "##ably" (a generic suffix). The double hash symbol indicates the word being a suffix of another word. Lastly, sentences must be padded since BERT and RoBERTa can only be fine-tuned when having an equal-sized matrix. To account for the differences in the number of words per sentence, meaningless words were added to shorter sentences to match the number of words in the longest sentence. The category assigned to these meaningless words is *PAD* (which stands for padding). An attention mask was applied to instruct BERT and RoBERTa to ignore meaningless words.

3.4 Fine-tuning BERT

A case-sensitive BERT model, consisting of 12 layers, 768 hidden layers, and 12 heads, was used for this study. The BERT model was tuned using a total of 3 epochs and a batch size of 16. Additionally, the Adam optimiser was applied with a learning rate of 5e-5.

3.5 Fine-tuning RoBERTa

A case-sensitive RoBERTa model, consisting of 12 layers, 768 hidden layers, and 12 heads, was used for this study. The RoBERTa model was tuned with the same hyperparameters as the BERT model and applied the same Adam optimiser.

3.6 Evaluation

Performance metrics precision (1), recall (2), and F1-score (3) were applied to evaluate each model. True positive (TP) refers to the model correctly matching a phrase to its category, false positive (FP) refers to the model incorrectly matching a phrase to a category, and false negative (FN) refers to the model not being able to match a phrase to its category. The F1-score was calculated using the harmonic mean of precision and recall, which made it suitable for evaluation despite the unevenly distributed data between each model.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

4 Results

This study built a total of four models in order to compare core concept categorisation results. First, a BERT and RoBERTa model were built without applying any data augmentation techniques. The training loss and validation loss of these two models are shown in figure 5. *PAD* categories were excluded from this plot to avoid distortion. Both models show comparable results: training loss decreases overtime and validation loss stabilises after the second epoch. According to the literature, both models show small indications of underfitting.

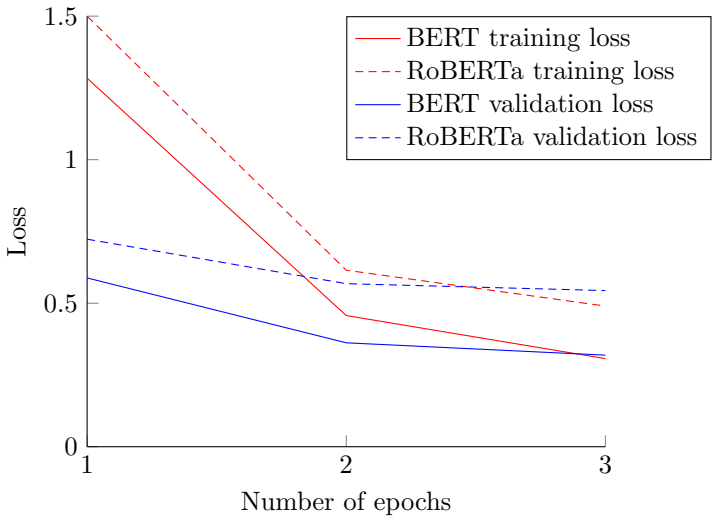


Figure 5: BERT and RoBERTa losses without data augmentation

After building the first two models, two additional models were built using data augmentation techniques. The training loss and validation loss of these two models are shown in figure 6. Similar to the models without any data augmentation, these two models show comparable results: training loss decreases to an extremely low number overtime and validation loss remains approximately the same. Due to a gap between training loss and validation loss in both models, there are strong indications of overfitting.

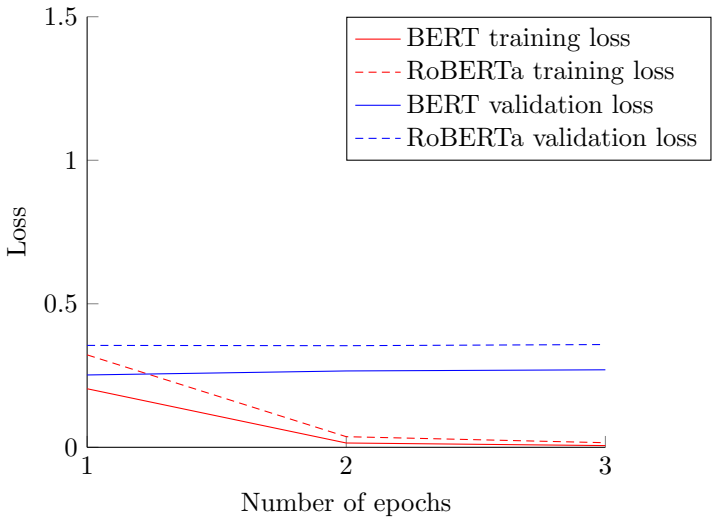


Figure 6: BERT and RoBERTa losses with data augmentation

Next, the performance metrics were calculated to evaluate the first two

models. Performance metrics for each category are shown in appendix A.1 (BERT) and in appendix B.1 (RoBERTa). Due to not having any TP values for some categories, performance metrics precision, recall, and F1-scores could not be determined for those categories and are denoted as *N/A*. To provide a better overview, all FP, FN, and TP values (including those with performance metrics *N/A*) were summed and added to table 7. Important to note, due to the high presence of irrelevant *O* and *PAD* categories in the results, these categories were excluded from the table in order to avoid distortion. Without applying any data augmentation techniques, RoBERTa achieved a substantially better F1-score compared to BERT. Despite the substantial difference, RoBERTa still predicts many categories incorrectly.

Model	**FP**	**FN**	**TP**	**Precision**	**Recall**	**F1**
BERT	90	1417	32	0.262	0.022	0.041
RoBERTa	809	1024	396	0.329	0.279	0.302

Table 7: Summed results of models without data augmentation

Lastly, the performance metrics were calculated to evaluate the last two models. All performance metrics are shown in appendix A.2 (BERT) and in Appendix B.2 (RoBERTa). Similar to the previous models, table 8 provides an overview with summed results of relevant categories. By applying data augmentation techniques, all performance metrics showed a significant increase compared to the models without data augmentation.

Model	**FP**	**FN**	**TP**	**Precision**	**Recall**	**F1**
BERT	422	706	923	0.686	0.567	0.621
RoBERTa	905	773	811	0.473	0.512	0.492

Table 8: Summed results of models with data augmentation

From the results in this study, it can be observed that BERT outperformed RoBERTa when applying data augmentation techniques and RoBERTa outperformed BERT when not applying data augmentation techniques.

5 Discussion

The results presented in this research are the first results giving answers to the performance comparison between BERT and RoBERTa in the context of core concept categorisation in geo-analytical questions. Moreover, it provides a performance comparison between a small data set and a larger augmented data set. BERT showed a better performance on training loss and validation loss compared to RoBERTa in the training process of both data sets. When looking at the performance metrics, RoBERTa built on the original data set performed significantly better compared to BERT

and BERT built on the augmented data set compared significantly better compared to RoBERTa. Nonetheless, both models presented a better performance when applying data augmentation techniques.

This study has several limitations and weaknesses. First and foremost, this study applied a data augmentation technique to overcome data scarcity, however, generating synthetic data without contextual information has the potential of generating semantically incorrect data. Moreover, this study only applied synonym replacement, while other data augmentation techniques can be considered to boost performance and prevent a model from overfitting. Future studies should focus on the performance of other data augmentation techniques and verify its result with domain experts which leads to more data, as well as improving data quality. To continue, a second limitation of this study is not evaluating each category separately. From the data, it can be observed that some categories performed significantly better compared to other categories, some even having an F1-score above 0.85 and many having an F1-score below 0.50. Future research can dive deeper into finding solutions to improve the performance of these categories. A third limitation is the absence of tuning hyperparameter values. Hyperparameter values were chosen based on previous work and are not tuned to work optimally with the provided data set. Future studies can put more focus on finding optimal hyperparameter values, such as number of epochs or batch size, in the context of core concept categorisation in geo-analytical questions. Furthermore, this study only compared two transformer-based models. Other deep learning models might produce better results. Future research should evaluate the potential of other deep learning models and compare results. Finally, results were compared using subjective judgment. Future research should put more thought in acceptable performance metric values by assessing previous work.

As discussed, the first step of developing a geo-analytical QA system is to accurately categorise core concepts in geo-analytical questions. This study created a foundation for future research by demonstrating significant performance boosts when applying data augmentation techniques to geo-analytical questions. Upcoming research on this topic can use the provided building blocks to further refine the implemented methods.

6 Conclusion

This study presented a comparison between transformer-based model BERT and RoBERTa on a small data set and a larger augmented data set. RoBERTa performs significantly better on a small data set compared to BERT and BERT performs significantly better on a large augmented data set compared to RoBERTa. When building a model categorising core concepts in geo-analytical questions, BERT in combination with data augmentation techniques should be applied.

All implemented methods in this study are released under an open

license and can be found at `https://github.com/ZWiersma/NER_BERT-and-RoBERTa`.

References

Aquino, N. R., Gutoski, M., Hattori, L. T., and Lopes, H. S. (2017). The effect of data augmentation on the performance of convolutional neural networks. *Braz. Soc. Comput. Intell.*

Bansal, M. A., Sharma, D. R., and Kathuria, D. M. (2021). A systematic review on data scarcity problem in deep learning: Solution and applications. *ACM Computing Surveys (CSUR).*

Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

Brownlee, J. (2018). What is the difference between a batch and an epoch in a neural network. *Machine Learning Mastery*, 20.

Crestani, F. and van Rijsbergen, C. J. (1995). Information retrieval by logical imaging. *Journal of Documentation.*

Desai, C. (2020). Comparative analysis of optimizers in deep neural networks. *International Journal of Innovative Science and Research Technology*, 5(10):959–962.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805.*

Eftimov, T., Koroušić Seljak, B., and Korošec, P. (2017). A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. *PloS one*, 12(6):e0179488.

Gehanno, J.-F., Rollin, L., Le Jean, T., Louvel, A., Darmoni, S., and Shaw, W. (2009). Precision and recall of search strategies for identifying studies on return-to-work in medline. *Journal of occupational rehabilitation*, 19(3):223–230.

Goldin, S. E. and Rudahl, K. T. (1997). Why is gis difficult. In *Proceedings of the 23rd Asian Conference on Remote Sensing. Kuala Lumpur, Malaysia.*

Hakala, K. and Pyysalo, S. (2019). Biomedical named entity recognition with multilingual bert. In *Proceedings of the 5th workshop on BioNLP open shared tasks*, pages 56–61.

Hestness, J., Ardalani, N., and Diamos, G. (2019). Beyond human-level accuracy: Computational challenges in deep learning. In *Proceedings of the 24th Symposium on Principles and Practice of Parallel Programming*, pages 1–14.

Jabbar, H. and Khan, R. Z. (2015). Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). *Computer Science, Communication and Instrumentation Devices*, 70.

Kobayashi, S. (2018). Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*.

Kuhn, W. (2012). Core concepts of spatial information for transdisciplinary research. *International Journal of Geographical Information Science*, 26(12):2267–2276.

Liang, C., Yu, Y., Jiang, H., Er, S., Wang, R., Zhao, T., and Zhang, C. (2020). Bond: Bert-assisted open-domain named entity recognition with distant supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1054–1064.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Z., Shen, Z., Li, S., Helwegen, K., Huang, D., and Cheng, K.-T. (2021). How do adam and training strategies help bnns optimization. In *International Conference on Machine Learning*, pages 6936–6946. PMLR.

Minar, M. R. and Naher, J. (2018). Recent advances in deep learning: An overview. *arXiv preprint arXiv:1807.08169*.

Mohit, B. (2014). Named entity recognition. In *Natural language processing of semitic languages*, pages 221–245. Springer.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., and Huang, X. (2020). Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

Rogers, A., Kovaleva, O., and Rumshisky, A. (2020). A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Scheider, S., Nyamsuren, E., Kruiger, H., and Xu, H. (2021). Geo-analytical question-answering with gis. *International Journal of Digital Earth*, 14(1):1–14.

Schneider, F., Balles, L., and Hennig, P. (2019). Deepobs: A deep learning optimizer benchmark suite. *arXiv preprint arXiv:1903.05499*.

Schwartz, R., Dodge, J., Smith, N. A., and Etzioni, O. (2020). Green ai. *Communications of the ACM*, 63(12):54–63.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48.

Shorten, C., Khoshgoftaar, T. M., and Furht, B. (2021). Text data augmentation for deep learning. *Journal of big Data*, 8(1):1–34.

Summers, C. and Dinneen, M. J. (2019). Improved mixed-example data augmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1262–1270. IEEE.

Tato, A. and Nkambou, R. (2018). Improving adam optimizer.

Tikhomirov, M., Loukachevitch, N., Sirotina, A., and Dobrov, B. (2020). Using bert and augmentation in named entity recognition for cybersecurity domain. In *International Conference on Applications of Natural Language to Information Systems*, pages 16–24. Springer.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, A., Costello, S., Cockburn, M., Zhang, X., Bronstein, J., and Ritz, B. (2011). Parkinson's disease risk from ambient exposure to pesticides. *European journal of epidemiology*, 26(7):547–555.

Wang, Y., Sun, Y., Ma, Z., Gao, L., Xu, Y., and Sun, T. (2020). Application of pre-training models in named entity recognition. In *2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, volume 1, pages 23–26. IEEE.

Wei, J. and Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Yadav, V. and Bethard, S. (2019). A survey on recent advances in named entity recognition from deep learning models. *arXiv preprint arXiv:1910.11470*.

Ying, X. (2019). An overview of overfitting and its solutions. In *Journal of physics: Conference series*, volume 1168, page 022022. IOP Publishing.

A BERT results

A.1 Without data augmentation

Category	FP	FN	TP	Precision	Recall	F1
B-CNAC	0	89	0	N/A	N/A	N/A
B-CNAER	0	3	0	N/A	N/A	N/A
B-CVAER	0	47	0	N/A	N/A	N/A
B-CVAL	0	43	0	N/A	N/A	N/A
B-EVE	0	55	0	N/A	N/A	N/A
B-EVEQR	0	4	0	N/A	N/A	N/A
B-FLDI	0	1	0	N/A	N/A	N/A
B-FLDN	0	25	0	N/A	N/A	N/A
B-FLDR	0	35	0	N/A	N/A	N/A
B-LOC	0	10	0	N/A	N/A	N/A
B-NET	0	10	0	N/A	N/A	N/A
B-NETQR	0	32	0	N/A	N/A	N/A
B-OBJ	85	364	26	0.234	0.067	0.104
B-OBJQO	0	1	0	N/A	N/A	N/A
B-OBJQR	0	9	0	N/A	N/A	N/A
B-PROPIR	0	22	0	N/A	N/A	N/A
I-CNAC	0	6	0	N/A	N/A	N/A
I-CNAER	0	4	0	N/A	N/A	N/A
I-CVAL	0	58	6	1.000	0.094	0.171
I-EVE	0	58	0	N/A	N/A	N/A
I-EVEQR	0	2	0	N/A	N/A	N/A
I-FLDN	0	47	0	N/A	N/A	N/A
I-FLDR	0	55	0	N/A	N/A	N/A
I-NET	0	7	0	N/A	N/A	N/A
I-NETQR	0	66	0	N/A	N/A	N/A
I-OBJ	5	299	0	N/A	N/A	N/A
I-OBJQR	0	17	0	N/A	N/A	N/A
I-PROPIR	0	48	0	N/A	N/A	N/A
O	954	7909	29756	0.969	0.790	0.870
PAD	8282	0	0	N/A	N/A	N/A

A.2 With data augmentation

Category	FP	FN	TP	Precision	Recall	F1
B-CNAC	7	7	82	0.921	0.921	0.921
B-CNAER	12	11	3	0.200	0.214	0.207
B-CVAER	3	94	1	0.250	0.011	0.020
B-CVAL	8	4	39	0.830	0.907	0.867
B-EVE	27	20	83	0.755	0.806	0.779
B-EVEQR	0	4	0	N/A	N/A	N/A
B-FLDI	0	1	0	N/A	N/A	N/A
B-FLDN	11	17	8	0.421	0.320	0.364
B-FLDO	1	0	0	N/A	N/A	N/A
B-FLDR	1	34	16	0.941	0.320	0.478
B-LOC	0	11	0	N/A	N/A	N/A
B-NET	5	10	6	0.545	0.375	0.444
B-NETQR	3	0	32	0.914	1.000	0.955
B-OBJ	104	151	267	0.720	0.639	0.677
B-OBJQI	16	0	16	0.500	1.000	0.667
B-OBJQO	0	1	0	N/A	N/A	N/A
B-OBJQR	1	9	0	N/A	N/A	N/A
B-PROPIR	32	12	17	0.347	0.586	0.436
I-CNAC	5	3	3	0.375	0.500	0.429
I-CNAER	0	4	0	N/A	N/A	N/A
I-CVAL	8	29	35	0.814	0.547	0.654
I-EVE	8	17	41	0.837	0.707	0.766
I-EVEQR	0	2	0	N/A	N/A	N/A
I-FLDN	16	36	11	0.407	0.234	0.297
I-FLDR	6	43	12	0.667	0.218	0.329
I-NET	0	7	0	N/A	N/A	N/A
I-NETQR	9	6	60	0.870	0.909	0.889
I-OBJ	114	143	156	0.578	0.522	0.548
I-OBJQR	1	17	0	N/A	N/A	N/A
I-PROPIR	24	13	35	0.593	0.729	0.654
O	446	162	37323	0.988	0.996	0.992

B RoBERTa results

B.1 Without data augmentation

Category	FP	FN	TP	Precision	Recall	F1
B-CNAC	0	89	0	N/A	N/A	N/A
B-CNAER	0	3	0	N/A	N/A	N/A
B-CVAER	0	47	0	N/A	N/A	N/A
B-CVAL	0	43	0	N/A	N/A	N/A
B-EVE	0	55	0	N/A	N/A	N/A
B-EVEQR	0	4	0	N/A	N/A	N/A
B-FLDI	0	1	0	N/A	N/A	N/A
B-FLDN	0	25	0	N/A	N/A	N/A
B-FLDR	0	37	0	N/A	N/A	N/A
B-LOC	0	10	0	N/A	N/A	N/A
B-NET	0	10	0	N/A	N/A	N/A
B-NETQR	0	32	0	N/A	N/A	N/A
B-OBJ	586	79	281	0.324	0.781	0.458
B-OBJQO	0	1	0	N/A	N/A	N/A
B-OBJQR	0	9	0	N/A	N/A	N/A
B-PROPIR	0	22	0	N/A	N/A	N/A
I-CNAC	0	6	0	N/A	N/A	N/A
I-CNAER	0	4	0	N/A	N/A	N/A
I-CVAL	0	64	0	N/A	N/A	N/A
I-EVE	0	58	0	N/A	N/A	N/A
I-EVEQR	0	2	0	N/A	N/A	N/A
I-FLDN	0	47	0	N/A	N/A	N/A
I-FLDR	0	56	0	N/A	N/A	N/A
I-NET	0	7	0	N/A	N/A	N/A
I-NETQR	0	66	0	N/A	N/A	N/A
I-OBJ	223	182	115	0.340	0.387	0.362
I-OBJQR	0	17	0	N/A	N/A	N/A
I-PROPIR	0	48	0	N/A	N/A	N/A
O	461	246	36808	0.988	0.993	0.990

B.2 With data augmentation

Category	FP	FN	TP	Precision	Recall	F1
B-CNAC	24	3	86	0.782	0.966	0.864
B-CNAER	9	11	3	0.250	0.214	0.231
B-CVAER	3	93	2	0.400	0.021	0.040
B-CVAL	7	38	5	0.417	0.116	0.182
B-EVE	14	58	29	0.674	0.333	0.446
B-EVEQR	0	4	0	N/A	N/A	N/A
B-FLDI	0	1	0	N/A	N/A	N/A
B-FLDN	8	18	7	0.467	0.280	0.350
B-FLDO	5	0	0	N/A	N/A	N/A
B-FLDR	4	27	25	0.862	0.481	0.617
B-LOC	0	11	0	N/A	N/A	N/A
B-NET	11	8	8	0.421	0.500	0.457
B-NETQR	6	0	32	0.842	1.000	0.914
B-OBJ	657	122	266	0.288	0.686	0.406
B-OBJQI	18	0	16	0.471	1.000	0.640
B-OBJQO	0	1	0	N/A	N/A	N/A
B-OBJQR	0	9	0	N/A	N/A	N/A
B-PROPIR	17	17	12	0.414	0.414	0.414
I-CNAC	0	6	0	N/A	N/A	N/A
I-CNAER	0	4	0	N/A	N/A	N/A
I-CVAL	4	55	9	0.692	0.141	0.234
I-EVE	7	5	53	0.883	0.914	0.898
I-EVEQR	0	2	0	N/A	N/A	N/A
I-FLDN	15	41	6	0.286	0.128	0.176
I-FLDR	9	41	15	0.625	0.268	0.375
I-NET	0	7	0	N/A	N/A	N/A
I-NETQR	11	6	60	0.845	0.909	0.876
I-OBJ	24	156	141	0.855	0.475	0.610
I-OBJQR	4	17	0	N/A	N/A	N/A
I-PROPIR	48	12	36	0.429	0.750	0.545
O	391	523	36367	0.989	0.986	0.988