WITS UNIVERSITY

School of Computer Science and Applied Mathematics
UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

Deep Learning with Free-Text Rationale for Cryptic Crosswords

Literature Review

William Hill (2115261)
Supervised by Dr Steven James (RAIL)

May 20, 2022

Contents

1 Introduction

Cryptic Crosswords are puzzles, appearing in many major newspapers such as the *Times*, whereby a natural language clue is given, which has a single answer comprising of one or more words (i.e. a word or phrase). "Cryptic" refers to the fact that the clue's surface-level meaning is equivocal and requires one or more clever wordplays and world knowledge in order to discover the true answer. Wordplay denotes any valid natural language disambiguation task, including phonetic, syntactic and character-level manipulation (e.g. homophones, anagrams, synonyms).

Consequently, any Machine Learning model aiming to solve these clues requires robust knowledge of the English language and its nuances (as any human attempting this challenge would).

A significant amount of research has focused on storing and retrieving knowledge within the parameters of a Language Model via pretraining and achieving state-of-the-art results on common NLP tasks (Petroni *et al.*, 2019; Talmor *et al.*, 2020; Roberts *et al.*, 2020). The intricacies of the required knowledge for solving Cryptic Crosswords necessitate additional techniques, such as Curriculum Learning (Rozner *et al.*, 2021), in order to acquire said knowledge more reliably.

However, solutions using current best practices (using a pretrained T5 model (Raffel *et al.*, 2020), fine-tuned on Cryptic Crossword datasets) yielded inadequate state-of-the-art results for predicting the answers (Efrat *et al.*, 2021; Rozner *et al.*, 2021), thus further work is needed on complex disambiguation tasks.

Previous work has shown that when additionally predicting the rationale behind a model's predicted label, the state-of-the-art label accuracy was improved upon or matched for common NLP tasks (Narang *et al.*, 2020; Kumar and Talukdar, 2020), suggesting that simultaneously predicting rationale can improve label prediction. This train of thought has not been attempted when considering Cryptic Crosswords.

Consequently, this review focuses on and discusses content relating to solving Cryptic Crosswords using Deep Learning Language Models which simultaneously predict a label with a corresponding Natural Language Rationale.

This review is structured as follows: Section 2 will introduce some concepts that will provide context to the content of later sections; Section 3 will address the various types of rationale, why they are useful, and which one best suits Cryptic Crosswords; Section 4 formulates the architecture of our neural network model such that training and testing it is as efficient as possible, given the format of the problem and structure of available data; Section 5 discusses various ways our model can acquire world knowledge and an understanding of the intricacies of natural language; Section 6 raises the concepts central to learning to decipher cryptic clues; and Section 7 introduces methods to evaluate the rationale produced and methods to better indicate the model's ability to generalise.

2 Background

The following are concepts required to give the correct context of points in this review.

Language Model A *language model* is a probability distribution over sequences of words, which gives the ability to predict those words (i.e. construct meaningful sentences and bodies of text). Specifically, we consider *neural language models*, which use Neural Networks to capture the complexities of natural language.

Masked Language Modelling (MLM) *MLM* is a technique whereby spans of input text are corrupted (e.g. replaced with "[MASK]") and training the model to "fill in the gaps" (i.e. the model is generative).

Meta-linguistics Chomsky and others (1976) define *meta-linguistics* as "the knowledge of the characteristics and structures of language", which refers to the phonology, morphology, syntax and semantics of language. For example, *meta-linguistic awareness* is recognising that *break* is the root of the word *unbreakable* and *un-* is a prefix for negation.

Multitask Learning (MTL) MTL "is an inductive transfer mechanism whose principle goal is to improve generalization performance" (Caruana, 1997), where a single model optimises multiple loss functions for *similar* tasks separately but simultaneously. Mechanically, it uses a shared portion of a neural network to learn a shared representation for the common elements of the tasks being learnt. If MTL is used for non-related tasks, a shared representation doesn't exist and so can't be learnt, hence the training diverges. Notably, MTL is not a single algorithm, but rather a collection of ideas and techniques (Caruana, 1997).

Natural Language Inference (NLI) NLI is a common NLP task where a Language Model is given a "premise" (i.e. a statement to establish context) and a "hypothesis" (i.e. another statement concerning content of the premise) and has to determine the validity of the hypothesis with respect to the premise: the hypothesis is either true (entailment), false (contradiction), or undetermined (neutral), where the term in brackets is the jargon of the task.

Natural Language Processing (NLP) NLP refers to the field of Artificial Intelligence which aims to produce machine learning models capable of understanding and producing language (e.g. text, speech) with a similar ability to humans.

Rationale Rationale is defined (by the Oxford Dictionary of English) as "a set of reasons or a logical basis for a course of action or belief". In the context of NLP, rationale is some explanation the model offers for why it produced the output it did, and gives some insight as to the internal state of the model.

Transfer Learning Transfer Learning is a technique whereby a model is first pretrained on a data-rich task and then fine-tuned on a downstream task. In NLP, the "data-rich task" is generally unsupervised learning on a large volume of text (e.g. scraped from the Internet) and the "downstream task" is the domain-specific task being investigated. The knowledge gained in pretraining can be applied to a downstream task, even if the pretraining task(s) and the downstream task and their domains are not related (Pan and Yang, 2010).

3 Free-Text Rationale

One of the main issues with Neural Networks is that they operate as black boxes: we know what the given input is and what the produced output is, but we have no way of knowing how the network arrived at the answer. Comparably, this is also how humans operate: if you ask someone a question and they give you an answer, you have no idea how he/she arrived at that answer; the only way to gain understanding is to ask the person to explain his/her thought process. Therefore, it is a natural extension to ask the model to explain itself as well.

Multiple streams of research have focused on making Neural Language Models more interpretable: for example, interpretability by construction, using extractive rationales, and using free-text[1] rationales. Interpretability *by construction* is performed by introducing architectural modifications such that intermediate steps in the model's decision making process can be observed and thus provide some insight into the final decision (Andreas *et al.*, 2016; Jiang *et al.*, 2019). For example, multiple neural networks may be used to solve parts of the problem and then combined with another network to produce the final answer. Interpretability *using extractive rationales* is performed by requiring the model to additionally output the span (substring) of the input text which motivated its answer, thus revealing what the model perceived as the most informative part(s) of the input (Lei *et al.*, 2016; DeYoung *et al.*, 2020). Interpretability *using free-text rationales* is performed by requiring that the model generate a free-form, natural language explanation for its answer. The advantage of free-text rationales is that it allows the model to produce an explanation where its decision making process includes information not immediately present in the input (and so extractive rationales would be insufficient) (Camburu *et al.*, 2018; Rajani *et al.*, 2019).

In addition to providing insight to a model's decision making process, rationales have also been shown to improve the accuracy of the label prediction itself for common NLP tasks, such as Commonsense Question Answering (Rajani *et al.*, 2019) and NLI (Kumar and Talukdar, 2020). Owing to the recency of work concerning Cryptic Crosswords, no investigations into the effect of rationales on the answer accuracy have taken place and, since accuracy has been improved for other tasks, it is a feasible avenue to explore.

Furthermore, owing to the ambiguous and complex nature of Cryptic Crosswords, free-text rationales will be the form of interpretation and explanation required to provide reasoning over-and-above information found in the clue itself. A logical step would be to consider first generating a number of explanations based on different meta-linguistics, and use these explanations to inform the answer (i.e. use a pipeline - made of two separate models, one mapping inputs to rationales , and another independently mapping the rationales to outputs). However, Wiegreffe *et al.* (2021) demonstrate that pipelines are insufficient for free-text rationales: the final

[1] "Free-Text" and "Natural Language" are used interchangeably to describe free-form text sequences

label predictions suffer from compounding errors; and rationales alone are subject to a loss of context so, therefore, label predictions suffer as a result (free-text rationales do not provide enough information alone to predict a label with sufficient accuracy). Instead, they present a *self-rationalising* model where the label and rationale are predicted simultaneously: this way, the model is forced to reason about the answer as it is producing the answer, which remedies both the issues raised.

4 Representation

Cryptic Crosswords have existed since the 1920s and have become a popular enigma. As such, there are magnitudes of clue-answer samples to use as data sets, such as those collated in Efrat *et al.* (2021) and Rozner *et al.* (2021), but there are far fewer explanations given for why an answer is in fact the answer.

Because we are going to simultaneously predict rationale and the answer, ideally we would want data where each clue-answer pair is explained which, practically, is not the case. However, we still want to use all the available data for training, to gain a more robust model. Fortunately, Narang *et al.* (2020) demonstrated that good-quality rationales can still be produced with a limited number of explained examples, although the quality does increase with more explained examples.

Consequently, we require a model that is able to predict a label either with or without an associated rationale. Raffel *et al.* (2020) presented a Unified Text-To-Text Transformer, T5, which can be used to accomplish this, because it takes, as input, a sequence of text, and produces, as output, another sequence of text. They argue that any problem can be formulated in this format: for example, instead of a classification problem producing an ID for the class, it will produce the class name itself. Consequently, T5 can be used to solve any task. Following Narang *et al.* (2020), prompts can be used to differentiate the task we want the model to perform: if we want the model to produce a rationale, we will prepend *explain* to the clue for the input text, and append *explanation* and the given explanation to the label for the target output text; if we do not want the model to predict a rationale because, for example, the sample does not have an associated explanation, then we do not include the prompts and an explanation. This allows us to utilise the entire collated data sets to train a single model which can simultaneously produce the label and a rationale. Importantly, this allows us to utilise an existing architecture proven to be excellent (Raffel *et al.*, 2020; Narang *et al.*, 2020; Sanh *et al.*, 2022), without having to add any special architectural modifications, which is complex and may or may not work with the same efficacy.

5 Acquiring Knowledge

One of the main attractions of Cryptic Crosswords is the requirement for the solver to possess crystallized knowledge[2], an acute understanding and appreciation for the nuances of language (English to be exact), and high fluid intelligence[3] (Friedlander and Fine, 2016), something at which many cryptics experts (like Kathryn

[2] the accumulation of knowledge and skills through experience
[3] the ability to think and reason abstractly

Friedlander) believe is extremely difficult for an artificial intelligence to excel (BBC, 2021).

Accordingly, the main starting point for building a Neural Language Model to solve Cryptic Crosswords is going to be storing as much knowledge as possible in its parameters: this allows the model to gain general-purpose knowledge that allows it to intrinsically understand language and apply it to a downstream task (Roberts *et al.*, 2020; Raffel *et al.*, 2020). The currently-accepted technique for doing so is pretraining the model, which in NLP is done on a data-rich task, generally unsupervised learning on unlabelled data, and has led to state-of-the-art results on many NLP benchmarks (Raffel *et al.*, 2020; Devlin *et al.*, 2019; Yang *et al.*, 2019; Dong *et al.*, 2019; Liu *et al.*, 2019; Lan *et al.*, 2020).

Raffel *et al.* (2020) collated the "Colossal Clean Crawled Corpus" (C4), which consists of hundreds of gigabytes of English language data scraped from the Web, and used this to pretrain their T5 model (the parameters for various-sized models have been released). These pretrained parameters are used as the initialisation for the model when fine-tuning on downstream tasks. Furthermore, updating all of the pretrained parameters when fine-tuning produces better performance (Raffel *et al.*, 2020).

This form of pretraining (on the entirety of C4) requires a large model to be truly representative of the knowledge contained in it: both Raffel *et al.* (2020) and Roberts *et al.* (2020) found that the larger the model, the better the downstream performance. However, training such a large model is slow and expensive and alternative modifications have been suggested.

Firstly, Guu *et al.* (2020) proposed Retrieval-Augmented Language Model pretraining (REALM), whereby a *knowledge retriever* is used to first retrieve relevant documents from a corpus (such as C4) and then uses those documents to make the prediction. This *knowledge retriever* will be trained as part of the learning process. An additional advantage to REALM is that it allows the model to store knowledge in a more interpretable and modular manner, meaning we can explicitly add or remove knowledge from the model as opposed to the implicit knowledge storing of conventional pretraining, where it is difficult to determine *what* knowledge the model possesses and attempting to get the model to remember certain information is hit-or-miss. All-in-all, REALM significantly improves on previous state-of-the-art results, by ensuring the model learnt more relevant information. Since Cryptic Crossword solvers require world knowledge, Salient Span Masking (SSM) (Guu *et al.*, 2020) is an important addition to guide the *knowledge retriever*. SSM masks content such as named entities (e.g. "United Kingdom") and dates (e.g. "January 1970"), which is relevant in a global context as opposed to a local one: for example, for a masked sample sentence, instead of the mask requiring local context (i.e. within the sentence itself), it requires global context (i.e. from a different section of the document, or an different document altogether). Roberts *et al.* (2020) also found that using the SSM strategy instead of traditional MLM improved downstream performance, although they did so without REALM.

Secondly, in the process of performing MLM, the input text sequence is corrupted and the model taught to generate the missing segment. Clark *et al.* (2020) recognised that the corrupted information is lost, which restricts the model's exposure to information. Instead, they suggest a method called *Replaced Token Detection*, whereby instead of entirely removing segments of the input, they are replaced with a plausible alternative which is generated by a separate network, and the model is trained to distinguish those segments of input sequences which are model-generated and those which are not (i.e. are unmodified). *Replaced Token Detection* ensures the model is exposed to all the input data instead of a subset of it, and scales better than MLM given the same computation resources, whilst still outperforming MLM.

A promising path would be to combine SSM (possibly with REALM) with Replaced Token Detection as the pretraining strategy, before fine-tuning on a downstream task where, potentially, all the pretrained parameters are updated. This would most likely lead to more efficient and represented learning, and a corresponding better performance.

6 Fine-Tuning

Using pretraining to acquire a model, based on T5, with encoded language knowledge and world knowledge is only the first part of transfer learning. The second, and last, part is to train the model to solve the specific task at hand, namely to simultaneously predict cryptic crossword answers and associated free-text rationales. This is conventionally called "fine-tuning" to a downstream task.

6.1 Multitask Learning

Fine-tuning can be performed in a number of ways, depending on the task. If there are multiple tasks, or the task can be broken down into multiple subproblems, then multitask learning is required, which can be performed either implicitly or explicitly. Implicit multitask learning is performed when the model learns how to perform smaller tasks as a byproduct of learning to perform a more complex task (this generally occurs during unsupervised learning) (Radford *et al.*, 2019). This type of learning is exploited during our pretraining process to teach the model to perform language-general tasks (of which there are too many to explicitly specify). For example, if the pretraining corpus includes content from a forum, such as StackOverflow, then in addition to the content learnt, the model may also understand how to provide and correctly format an answer when given a question. Explicit multitask learning is performed when an instruction of the task to perform is included in the input (this generally occurs during supervised learning). In NLP, this instruction is generally a prompt added to the input text (Sanh *et al.*, 2022). For example, given the input text *"The artwork appeared overnight. It seems to be signed by someone named Banksy. I must say, it is rather exquisite"*, we can perform Sentiment Analysis by prepending *"sentiment: "* to the text, or we can summarise the text by prepending *"summary: "* to the text. Sanh *et al.* (2022) also demonstrated that enough variability in natural language prompts (i.e. if the prompt is included in a manner similar to how a human would ask another person to perform a specific task) makes the model robust to variations in held-out prompts, although the need for resilience to this variability should not be present in our Cryptic Crossword domain.

Sanh *et al.* (2022) showed that explicit multitask learning is the more effective of the two if we require our model to perform multiple tasks. In our Cryptic Crossword setting, we can utilise explicit multitask learning and prompting to differentiate when our model needs to predict a rationale with the answer or not in order to gain the performance boost of explicit multitask learning.

6.2 Curriculum Learning

Similarly to how we considered human behaviour when deciding to ask a neural network model to explain its label prediction, we need to consider the parallels between how a human would learn to perform a complex task and how our model could learn to perform a complex task. Humans generally start with simpler concepts and move onto progressively harder ones at a later stage, and typically use the simpler concepts to aid the learning of the advanced ones. For example, children advance through grades at school, with each grade teaching them progressively more difficult content and often builds on the work learnt in previous grades, or, to follow the adage, *you need to walk before you can run*. Simply asking a person, or our model, to learn the complex task outright leads to degraded performance since trying to learn too many new things at once hinders the ability to learn them (Rozner *et al.*, 2021) (this is similar to the effect of explicit multitask learning improving the ability to generalise over implicit multitask learning).

Bengio *et al.* (2009) introduced the idea of applying this *Curriculum Learning* to machine learning models. A survey by Soviany *et al.* (2022) provides multiple examples where Curriculum Learning outperforms conventional training methods (i.e. attains better accuracy), and Rozner *et al.* (2021) demonstrate that Curriculum Learning in a Cryptic Crossword context dramatically improves results (over those attained via conventional training techniques in Efrat *et al.* (2021)) and use explicit multitask learning to do so: first train the model on progressively harder subproblems (e.g. definition lookup, then word descrambling etc.), and finally train the model on the problem of Cryptic Crossword clue deciphering. Soviany *et al.* (2022) also recognise that Curriculum Learning speeds up convergence during training.

Two major problems exist with Curriculum Learning: how do we decide which tasks are simpler and which are more complex, and the occurrence of catastrophic forgetting.

For the first one, Soviany *et al.* (2022) considered multiple different methods for sorting tasks by difficulty: rule-based methods (Bengio *et al.*, 2009), self-paced learning (where the difficulty of examples is determined based on the model's own performance), manual annotation by domain experts, and using domain-dependent difficulty measures (e.g. in many NLP tasks, the difficulty of an example can be determined by the length of the input text) etc.. Soviany *et al.* (2022) conjecture that the best method for designing a curriculum is one where we model our own human learning experience, a method which Rozner *et al.* (2021) used with favourable results. Additionally, when separating the examples by difficulty, we need to ensure that each difficulty level has enough diversity (in terms of labels) to ensure we are still able to generalise effectively.

For the second one, catastrophic forgetting refers to the phenomenon where a cognitive system has previous information completely erased as a result of learning new information (French, 1999). Typically, all natural cognitive systems gradually forget old information as new information is learnt. However, the better the system is able to generalise, the more likely it is to catastrophically forget (French, 1999). Neural Networks are an example of such a cognitive system (French (1999) calls them "connectionist networks"): the very features which allow the model to generalise are what will cause the model to forget the "simpler" tasks when it learns the more advanced tasks. Rozner *et al.* (2021) deals with issue by periodically feeding the model an earlier task (although this only decreases the liklihood of catastrphic forgetting occurring).

6.3 Parameter Updates

There are multiple ways the parameters (weights) of the model can be updated. If additional layers were added to the model to, for example, allow a downstream task to utilise knowledge acquired during pretraining but still use task-specific modifications (e.g. to output the answer in a specific format etc.), then an option is to only update the added layers' parameters. However, regardless if modifications to the architecture were made, all the parameters, including those learnt during pretraining, may also be updated when fine-tuning. Raffel *et al.* (2020) found that the latter is the optimal method since it yields better performance.

7 Evaluation

The last major point to consider in this review is how to evaluate the accuracy of our predictions and quality of out produced rationales. Traditionally, accuracy is measured by the percentage of examples for which correct labels were produced. The difficult part is measuring the quality of the produced rationales. It is worth noting that the research question is focused entirely on the effect of rationales on label accuracy. Although, it could be useful to investigate the quality of the rationale and whether it is faithful (i.e. if it is an accurate explanation of the model's decision making process). Additionally, even if simultaneously predicting rationale does not improve label accuracy, then a faithful rationale will give us insight into how the model works and may provide avenues for future work to explore.

7.1 Rationale Quality

The quality of a natural language rationale can be difficult to evaluate, owing to the many nuances of language itself. Roberts *et al.* (2020) found that simply comparing the overlap of the produced rationale to the ground-truth rationale is insufficient: a free-form answer has linguistic variations that don't alter the correctness of the answer, but impact how much of the answer overlaps with the expected answer ("overlap" is a notion where words and phrases, for example, are matched). Wiegreffe *et al.* (2021) suggested a more robust, but more computationally expensive, alternative: an automated simulatability score.

This simulatability score is an approximation of human simulatability (Doshi-Velez and Kim, 2017), which Wiegreffe *et al.* (2021) recognise is a reliable measure of rationale quality. It measures the additional predictive ability of the rationale over the input, and is measured by the performance difference between runs where the rationale is, and is not, included with input (when predicting the label). Consequently, the rationale will first have to be produced, and then two models used to establish this performance difference: one model takes only the input to produce the label; the other model takes both the input and our predicted rationale to produce the label.

The simulatability scores have a sufficiently high correlation with human simulatability scores (Hase *et al.*, 2020) and can thus be used as a reliable measure of rationale quality.

7.2 Rationale Faithfulness

Faithfulness concerns the idea that the produced rationale is representative of the model's decision making process, and is not just a plausible-sounding alternative. A faithful rationale will not only correlate to the predicted label, but will also offer insight into the model's behaviour.

Wiegreffe *et al.* (2021) offer two metrics to measure the faithfulness of free-text rationales:

- *Robustness Equivalence:* the labels and rationales are similarly stable/unstable when Gaussian noise (with a mean of zero and various variance values, σ^2) is added to the input. This is measured by the change in the label accuracy, the number of labels that change (to any other value), and the change in rationale quality as provided by the simulatability score. "Stable" refers to a small change, and "unstable" refers to a large change. In practice, the change for the labels and rationales needs to be compared: if they are similar, then we can say they are equivalently robust; if they are not similar, then we can say that they are not equivalently robust. Wiegreffe *et al.* (2021) concluded that *self-rationalising* models are equivalently robust with respect to labels and free-text rationales.
- *Feature Importance Agreement:* The idea is that if the label and corresponding rationale are associated, then the input tokens important for the label prediction should also be important for the rationale production. This is measured using Gradient Attribution, with respect to a predicted class p:

$$a(\mathbf{x}^{(i)}; l_p) = f(\nabla_{\mathbf{x}^{(i)}} l_p) \in \mathbb{R},$$

where f is a function which reduces the gradient of the predicted class' logit $l_p = \sigma^{-1}(p)$ with respect to an input token embedding $\mathbf{x}^{(i)} \in \mathbb{R}^d$, to a scalar. f is chosen to be the L_1 norm, which outperforms other function choices (Wiegreffe *et al.*, 2021). We can compute the sum of a over m logits to arrive at the attribution for a sequence of tokens (i.e. text):

$$a(\mathbf{x}^{(i)}) = a(\mathbf{x}^{(i)}; \{l_p^{(k)}\}_k) = \Sigma_{k=1}^{m} a(\mathbf{x}^{(i)}; l_p^{(k)}) \in \mathbb{R}.$$

The attribution of a sequence of n input token embeddings, $\mathbf{X} \in \mathbb{R}^{n \times d}$ is a vector

$$a(\mathbf{X}) = [a(\mathbf{x}^{(1)}), ..., a(\mathbf{x}^{(n)})] \in \mathbb{R}^n$$

By choosing the values of k to refer to label logits $\mathscr{L}$ or rationale logits $\mathscr{R}$, we arrive at attribution scores for the labels and rationales, respectively:

$$a(\mathbf{X})_{\mathscr{L}} \quad \textit{and} \quad a(\mathbf{X})_{\mathscr{R}}.$$

Using these attribution scores, we remove the top $k\%$ ($k \in \{10\%, 20\%, 30\%\}$) of attributed tokens with respect to either the labels or rationales for each data point and measure performance with respect to the other (i.e. if we remove the tokens important for label prediction, then we measure the effect on rationale quality and vice versa). If either of these tests is notably different from a test where a random $k\%$ of tokens are removed, then the model exhibits feature importance agreement, which Wiegreffe *et al.* (2021) concluded a *self-rationalising* model does.

Consequently, if these two metrics return that the labels and rationales produced are associated, then we can reasonably state that the rationales are faithful and provide insight into the label prediction and the model's behaviour.

7.3 Generalisability

The ability to solve a problem for held-out (unseen) samples is of utmost importance. We are not guaranteed to be exposed to all possible clues, or even types of clues, and so our model needs to be able to learn the principles of solving Cryptic Crosswords so it can apply the techniques in general.

Both Efrat *et al.* (2021) and Rozner *et al.* (2021) recognised this, and further recognised that there are similarities between the answers of clues which prevent the model from learning how to solve the clue: it rather exploits lexical similarity between clues with the same answer (it becomes a case of "these two clues are similar, so their answers must be similar" instead of solving each clue outright).

Rozner *et al.* (2021) went on to formulate a way to split the data such that the lexical similarity cannot be exploited. They use two types of splits: a naive disjoint split and a word-initial disjoint split. The naive disjoint split simply ensures that all clue-answer pairs with the same answer only appear in one of the training, validation or test data sets (this was also used in Efrat *et al.* (2021)). The word-initial disjoint split further ensures that all clue-answer pairs with answers that start with the same two letters appear in the same split. This is done since derivatives of words are lexically similar to each other, so much so that our model could perceive them as the "same" (i.e. the difference between them could be negligible). This prevents the model from merely finding a shortcut to minimise the cost function by memorising paraphrases and their answers, and ensures the model truly learns the problem.

Performance was more degraded for the word-initial disjoint split than for the naive disjoint split, supporting the argument that the model was exploiting lexical similarity. Thus, the test set generated using the word-initial disjoint split gives a better indication of the model's ability to generalise to held-out input.

8 Conclusion

Cryptic Crosswords are puzzles characterised by needing to overcome extreme ambiguity, to the point where even humans struggle to solve them. Various attempts have been made to build a model to solve them, but all failed to effectively generalise, suggesting there is plenty of room for further work.

Owing to the recency of work concerning Cryptic Crosswords, no investigations into the effect of rationales on the answer accuracy have taken place and, since accuracy has been improved for other tasks, it is a feasible avenue to explore. The most suitable form of rationales for Cryptic Crossword reasoning are free-text rationales since they can be simultaneously predicted with a label and, more importantly, allow the model to provide an explanation where its decision making process includes information not immediately present in the input, which is the main restriction for other forms of rationale.

In addition to the scarcity of explanations of answers in Cryptic Crossword datasets, this simultaneous prediction requirement necessitates a model able to solve multiple tasks (predict a label with or without a rationale). A T5-based model, as a text-to-text transformer, is ideally suited to address this problem format.

A main requirement for solving Cryptic Crosswords is the acquisition of world knowledge and a native understanding of natural language. Normally, this has been addressed by pretraining using MLM on a large language corpus, although it has been demonstrated that using a discriminative method (like *Replaced Token Detection*) instead of a generative method (like MLM) improves pretraining efficacy dramatically. Further, using Salient Span Masking (SSM) to more intelligently choose what information to remember also improves performance.

Consequently, a promising path would be to combine SSM (possibly with REALM) with *Replaced Token Detection* as the pretraining strategy, before fine-tuning on a downstream task where all the pretrained parameters are updated. Fine-tuning should be done using a Curriculum Learning approach with explicit multitask learning, where subtasks (focusing on learning meta-linguistics) are ranked accordingly to a human estimation of difficulty.

Although rationale quality and faithfulness are not paramount to our research question, they can be used to offer insight into the model's decision making process and may offer further avenues to explore in future work.

References

[Andreas *et al.* 2016] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 39–48, 2016.

[BBC 2021] BBC. *Cryptic crosswords: A puzzling British obsession*, 2021. Retrieved 2022-05-12, from https://www.bbc.com/culture/article/20210302-cryptic-crosswords-a-puzzling-british-obsession

[Bengio *et al.* 2009] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA, 2009. Association for Computing Machinery.

[Camburu *et al.* 2018] Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. e-snli: Natural language inference with natural language explanations. *Advances in Neural Information Processing Systems*, 31, 2018.

[Caruana 1997] Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[Chomsky and others 1976] Noam Chomsky et al. *Reflections on language*. Temple Smith London, 1976.

[Clark *et al.* 2020] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020.

[Devlin *et al.* 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[DeYoung *et al.* 2020] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, July 2020. Association for Computational Linguistics.

[Dong *et al.* 2019] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, M. Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. *ArXiv*, abs/1905.03197, 2019.

[Doshi-Velez and Kim 2017] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv: Machine Learning*, 2017.

[Efrat *et al.* 2021] Avia Efrat, Uri Shaham, Dan Kilman, and Omer Levy. Cryptonite: A Cryptic Crossword Benchmark for Extreme Ambiguity in Language. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4186–4192, Online and Punta Cana, Dominican Republic, Nov 2021. Association for Computational Linguistics.

[French 1999] R M French. Catastrophic forgetting in connectionist networks. *Trends Cogn. Sci.*, 3(4):128–135, April 1999.

[Friedlander and Fine 2016] Kathryn J. Friedlander and Philip A. Fine. The grounded expertise components approach in the novel area of cryptic crossword solving. *Frontiers in Psychology*, 7, 2016.

[Guu *et al.* 2020] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training. *ArXiv*, abs/2002.08909, 2020.

[Hase *et al.* 2020] Peter Hase, Shiyue Zhang, Harry Xie, and Mohit Bansal. Leakage-adjusted simulatability: Can models generate non-trivial explanations of their behavior in natural language? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4351–4367, Online, November 2020. Association for Computational Linguistics.

[Jiang *et al.* 2019] Yichen Jiang, Nitish Joshi, Yen-Chun Chen, and Mohit Bansal. Explore, propose, and assemble: An interpretable model for multi-hop reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2714–2725, Florence, Italy, July 2019. Association for Computational Linguistics.

[Kumar and Talukdar 2020] Sawan Kumar and Partha Talukdar. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online, July 2020. Association for Computational Linguistics.

[Lan *et al.* 2020] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *ArXiv*, abs/1909.11942, 2020.

[Lei *et al.* 2016] Tao Lei, Regina Barzilay, and Tommi Jaakkola. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas, November 2016. Association for Computational Linguistics.

[Liu *et al.* 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

[Narang *et al.* 2020] Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. WT5?! Training Text-to-Text Models to Explain their Predictions. In *arXiv preprint arXiv:2004.14546*, 2020.

[Pan and Yang 2010] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[Petroni *et al.* 2019] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics.

[Radford *et al.* 2019] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[Raffel *et al.* 2020] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683, 2020.

[Rajani *et al.* 2019] Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy, July 2019. Association for Computational Linguistics.

[Roberts *et al.* 2020] Adam Roberts, Colin Raffel, and Noam Shazeer. How Much Knowledge Can You Pack Into the Parameters of a Language Model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online, Nov 2020. Association for Computational Linguistics.

[Rozner *et al.* 2021] Josh Rozner, Christopher Potts, and Kyle Mahowald. Decrypting Cryptic Crosswords: Semantically Complex Wordplay Puzzles as a Target for NLP. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

[Sanh *et al.* 2022] Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask Prompted

Training Enables Zero-Shot Task Generalization. In *The Tenth International Conference on Learning Representations*, 2022.

[Soviany *et al.* 2022] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International Journal of Computer Vision*, 2022.

[Talmor *et al.* 2020] Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. oLMpics-On What Language Model Pre-training Captures. *Transactions of the Association for Computational Linguistics*, 8:743–758, 12 2020.

[Wiegreffe *et al.* 2021] Sarah Wiegreffe, Ana Marasović, and Noah A Smith. Measuring Association Between Labels and Free-Text Rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284, Online and Punta Cana, Dominican Republic, Nov 2021. Association for Computational Linguistics.

[Yang *et al.* 2019] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, 2019.