Inverse Bitext Classification: Reverse Engineering of Sequences from Labels

Dor Lotan, Michal Shuvi

1 Introduction

Bitext classification is a class of NLP problems, in which the goal is to label pairs of sequences $\langle X, Y \rangle$. This includes a variety of tasks, such as yes/no question answering and many others. One such task is Natural Language Inference (NLI), which involves some premise P as an input, with the goal of generating a hypothesis $\hat{H}$ where P entails $\hat{H}$. A variant of this task is the Stanford NLI (SNLI) dataset , composed of samples of the form $\langle P, H, y \rangle$ where P is an image description which serves as premise, H is another image description which serves as hypothesis, and $y \in \{Entailment, Contradiction, Neutral\}$ is the logical relation between P and H. The task of SNLI is to generate $\hat{y}$ given $\langle P, H \rangle$. In other words: Find the logical relation between two sentences. We introduce the inverse task; Inverse Bitext Generation. Given the label and one element of the pair $\langle X, Y \rangle$, we attempt to generate the missing sequence. For example, Inverse NLI (INLI): Given a premise P and a label y, generate $\hat{H}$ such that $\hat{H}$ relates to P with the label y. To this end, we use T5: Text To Text Transfer Transformer (Raffel et al., 2019) in order to generate strings given an input string.

For example, consider the following premise: *Two men climbing a wooden scaffold.*
The hypotheses our model generates for this sentence are:
Entailment: *Two men are climbing a scaffold.*
Contradiction: *Two men are playing chess.*
Neutral: *Two men are climbing a scaffold to get to the top of the building.*
Note how the generated entailment follows directly from the premise, as any image described by *Two men climbing a wooden scaffold* is necessarily an image where *Two men are climbing a scaffold*. Contrast this with contradiction: an image described by the premise is necessarily not an image where *Two men are playing chess*. Finally, an image described by the premise could be an image where *Two men are climbing a scaffold to get to the top of the building*, or it could be not. Therefore, this hypothesis is neutral with regards to the premise.

A similar reasoning applies to the other examples of these labels: entailments tend to be generalizations of the premise, contradictions are a description of a different image, and neutral hypotheses tend to add on new details which do not contradict the premise, while not following directly from it. These two constraints would prove difficult for our models when inspecting its output.

We later introduce two additional Inverse Bitext Classification tasks. One is centered around yes/no question answering, where we generate questions about a text given their answers, and the other is generating comments on internet threads given their user scores as positive or negative.

2 Methods

Our tool for this task is a conditional language model. That is, a model that learns to predict token probabilities, given previous tokens and an input:

$$P(y_i | Y_{<i}, X)$$

In other words, given an input sequence $X = (x_1, ..., x_n)$ we predict the probability of an output sequence $Y = (y_1, ..., y_m)$, token by token. Training such a model is achieved with Cross Entropy:

$$-\frac{1}{|D|} \sum_{y_i \in D} \log P(y_i | Y_{<i}, X)$$

That is, the *log likelihood* of each next token given the last inputs, divided by the amount of tokens overall. We seek to minimize this over a set of input sequences, although the more common metric is Perplexity, which is the exponent of Cross

Entropy. Once the model predicts those probabilities, it chooses the next token based on some search algorithm. When the next token is generated, it is replaced by the correct token for the next iteration (*”Teacher Forcing”*).

Moreover, we take advantage of pre-training in order to incorporate meaning into our model. This is done by training the model on some task, such that the encodings it creates for each token in a given context are informative and meaningful, thus improving overall results.

3 Experiment Setup

For the purposes of this task, we use Text To Text Transfer Transformer, or T5. This is an encoder-decoder network, pre-trained for auto-regressive string generation. The training process is divided into two stages: pre-training and task specific training.

The pre-training phase is performed by masking sequences within every input sentence, and having the model generate the masked content. This method forces the model to create encodings that contain some information relevant to the reconstruction process, with the purpose of introducing semantic meaning into the encoding vectors.

Additionally, T5 uses Fine Tuning while training: the encoder and decoder keep updating after the pre-training phase. This allows the model to learn task-specific encodings. For our purposes, we use Huggingface’s Transformers library. This includes pre-trained T5 models of various sizes from which we can choose and train.

4 Datasets and Experiments

4.1 SNLI

For our primary experiments, we will use the Stanford Natural Language Inference dataset, or SNLI (Bowman et al., 2015). This dataset contains ∼500,000 examples of the form $\langle P, H, y\rangle$ where P is a premise sentence, H is a hypothesis sentence and y is the label describing their logical relation. We construct training examples in the following format:

```
INLI entailment: An old man with
a package poses in front of an
advertisement. </s>
```

`INLI` is the task marker for Inverse NLI. For each training example, the model generates some hypothesis $\hat{H}$, such that P and $\hat{H}$ should have a logical relation described by y. However, we also attempt to generalize this process to two more tasks, using two more datasets, as described below.

4.2 BoolQ

The first is BoolQ (Clark et al., 2019): A dataset comprised of ∼10,000 examples of the form $\langle T, Q, a\rangle$, where T is a passage of text, Q is a yes/no question about that text, and $a \in \{True, False\}$ is the answer to Q based on T. As per our inverse generation concept, we train a model to receive a text T and an answer a, and generate a question $\hat{Q}$ such that a is the answer to $\hat{Q}$ based on T (similar to the game show *Jeopardy*).

Note the similarity between question and hypothesis generation in INLI: If we take a question Q and change it to a premise, then $a = True$ is analogous to $y = entailment$, and $a = False$ is analogous to $y = contradiction$. For example, ”*Was George Washington the first U.S president?*” can be taken as ”*George Washington was the first U.S president*” which is either an entailment of T (some text about Washington), or a contradiction of T.

4.3 Reddit: r/jokes

The second supplementary dataset is based on the Reddit corpus. Reddit is a network of forums (a.k.a subreddits), in which users can submit posts, comment on posts or on other comments, and vote any post or comment up or down. Each post and comment have a numeral score based on user votes on them. Cornell University’s Conversational Analysis Toolkit library, or ConvoKit (Chang et al., 2020) contains a corpus of conversations on Reddit within each subreddit, which we use to construct a new dataset of 165,000 examples, each of the form $\langle P, C, y\rangle$ where P is a post on the Jokes subreddit, C is a direct comment to that post, and $y \in \{positive, negative, neutral\}$ is that comment’s label. Specifically, y is constructed from a score $s \in \mathbb{Z}$ such that: $y = \begin{cases} \text{positive} & s > 4 \\ \text{negative} & s < 1 \\ \text{neutral} & otherwise \end{cases}$

Each comment starts with a score of 1, hence the neutral label. Our model will learn to generate a comment $\hat{C}$ based on a post P and a target label y, such that $\hat{C}$ is a comment to P expected to receive a label of y. This task is not as clear cut as the first two, as the question of user scores is not as clearly predictable.

4.4 Parameters and Evaluation Methods

For every dataset, we will examine the model's loss over time, as well as a small sample of results which we will judge manually. Additionally, for INLI we will employ a pre-trained SNLI network to judge our generated sentences. In other words, from $\langle P, y \rangle$ we generate some $\hat{H}$. We then input $\langle P, \hat{H} \rangle$ to an SNLI network to obtain a new label $\hat{y}$. If our generated sequences are similar to those appearing in the SNLI dataset, then the network's accuracy on our sample (i.e zero-one loss) should match its' accuracy on SNLI, which can serve as a good evaluation of our results. The models we used were from Huggingface's pre-trained library. In terms of hyper-parameters, see Table 3.

Additionally, we used an Adam optimizer with betas $\langle 0.9, 0.999 \rangle$.

5 Results and Analysis

5.1 Convergence

All three models trained on their respective datasets' training samples, yield the graphs shown in figures 1, 2, and 3.

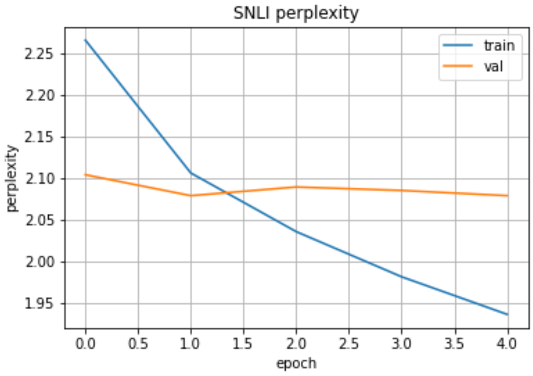

Figure 1: SNLI perplexity per epoch

As shown in figure 1, the INLI model learns the training data fairly quickly, yet still maintains a constant perplexity on the validation dataset. A similar process occurs in 2 and 3. This is explainable with regards to such models, as the validation dataset is not learned directly, and therefore might not serve as a reliable metric. While we aim to generate sequences which behave similarly to those in the validation dataset, we would not necessarily consider a deviation from the exact sequence a mistake by the learner. Despite this, the generated results do hold promise.

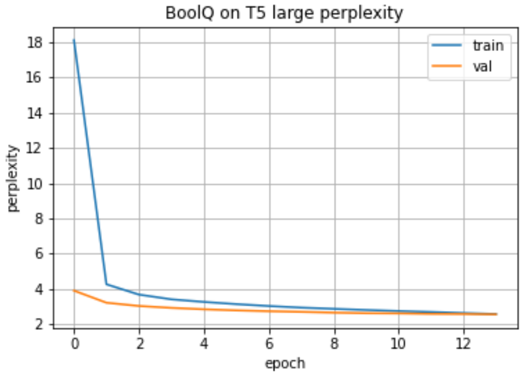

Figure 2: BoolQ perplexity per epoch

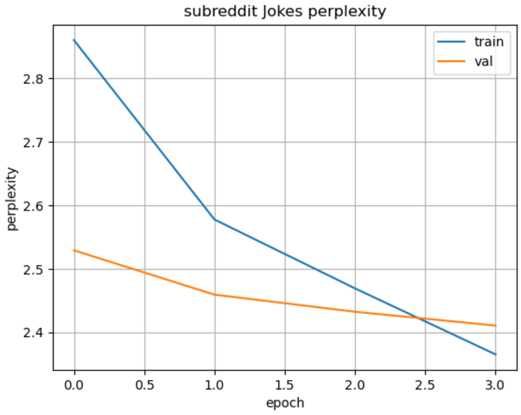

Figure 3: Reddit perplexity per epoch.

5.2 Manual Evaluation

A list of sample results was received from each model, for manual analysis. These samples are generated from the first 100 or 200 samples in the test datasets of each model.

For the task of INLI, a total of 198 samples were taken, of which 176 were judged to be accurate (88.8%). The same was done for BoolQ and Reddit, with results as described in tables 1 and 2.

Starting with our main task of INLI, it is clear that some labels are easier to generate hypotheses for; both neutral and contradiction obtain a score close to 85%. However, entailments are judged correct 95% of the time. A possible account for this is from *Creativity* and *Correctness*: to generate a neutral hypothesis, the model usually adds details to some part of the premise which were not present originally (creativity), while not contradicting it (correctness). Entailments require just correctness, while contradictions require just creativity. This will be further expanded upon with a detailed experiment in 5.4. A sample of the results can be

Label	Entail.	Contr.	Neut.
Total	69	65	64
Correct	66	55	55
% Correct	95.6%	84.6%	85.9%

Table 1: Hand-checked results for SNLI.

Label	True	False
Total	71	31
Correct	51	23
% Correct	72.8%	74.1%

Table 2: Hand-checked results for BoolQ.

seen in appendix D.1.

As for BoolQ, it generated 74 correct questions for of 102 texts and answers, resulting in 72.5% accuracy. Out of 28 mistakes, 17 were cases we consider to be ”Bad answers”: the model generated a valid question about T, but the answer was the opposite of a. That being the case, it’s safe to say the model learned, in general, how to ask questions about a text but not necessarily to be answer directed. There were also several ”Templates” the model might have picked up. For instance, many of the generated questions were of the form ”Is x the same as y”, with some cases of x=y. See appendix D.2 for a sample of the results.

Finally, manual inspection of the Reddit dataset is difficult to define. Since it is ambiguous, we are not capable of performing an exact evaluation. A possible method of evaluation could be a survey of Reddit users, in which they are asked to distinguish real comments from generated ones.

Despite that, appendix D.3 includes examples of results that seem good, and others that seem bad to us.

5.3 Automatic Evaluation

We have used a pre-trained SNLI network to evaluate the results from our INLI model. The model we chose was TextAttack’s (Morris et al., 2020) `bert-base-uncased-snli`, which has an accuracy of 90.48% on SNLI [1]. We had our INLI model generate hypotheses for the entirety of the test dataset, resulting in 9,824 labeled samples $\langle P, \hat{H}, y \rangle$. These were then fed into TextAttack’s model for evaluation. This model generated 8,826 correct predictions, i.e 89.84% accuracy. This implies our results are very similar in structure to SNLI, and indicates that our model’s hypotheses do relate to their premises as described by their labels, as the margin of error between our dataset and `snli_test` was less than 0.75%.

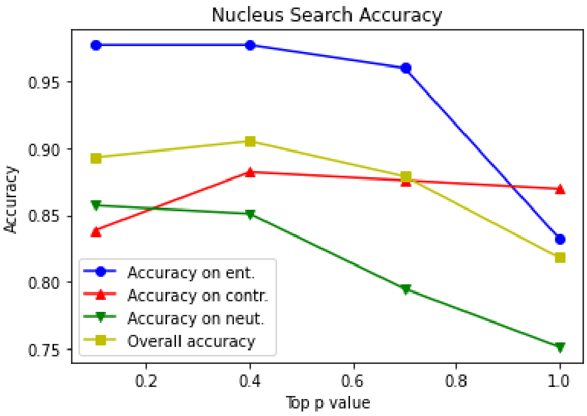

Figure 4: TextAttack’s Accuracies with respect to p

These conclusions are further supported by the confusion matrices of TextAttack’s model on the `snli_test` dataset 5, as opposed to our generated dataset 6. As shown, a majority of incorrect classifications tend to be $Entailment$, as our model tends to summarize or even copy the premise in its fail cases. Also, it is clear how the task of generating contradictions and neutral statements is harder for our model than entailments, while TextAttack’s model does have difficulties with neutral hypotheses.

5.4 Additional Experiment: Nucleus Search

As stated in 5.2, both contradictions an neutral statements require a degree of Creativity, and neutral statements require Correctness. To test this claim, we used our model to generate four datasets of 500 samples, using different p values for nucleus search. Testing those datasets on TextAttack’s model yields the accuracies described in 4. The labels requiring correctness ($Entailment$ and $Neutral$) suffer from an increase in randomness. Contrast this with $Contradiction$, which becomes more accurate as p increases, as it requires only creativity. In other words, randomness leads to creativity, but has an adverse effect on correctness. The optimal p value this experiment yields is 0.4, and so this was chosen for our final SNLI model.

[1] once the samples with label ”-” were removed.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference.

Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Z. Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. Convokit: A toolkit for the analysis of conversations.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. 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 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

John X. Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp.

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

A Source Code

Source code and results can be found at: `https://github.com/michalshuvi/INLI_project`

B Parameters per Dataset

See 3.

Dataset	SNLI	BoolQ	Reddit
Model	`base`	`large`	`base`
Batch Size	100	90	100
Epochs	5	14	4
Learn Rate	10^{-3}	10^{-5}	10^{-3}
Search Alg.	Nucleus	Greedy	Nucleus
p **value**	0.4	-	0.4

Table 3: Dataset Training Parameters.

C Confusion Matrices

See 5 and 6.

D Generated Examples

Included here are some examples of the sentences our models generate. Those marked with ✓were judged to be correct, and those marked with × were judged incorrect.

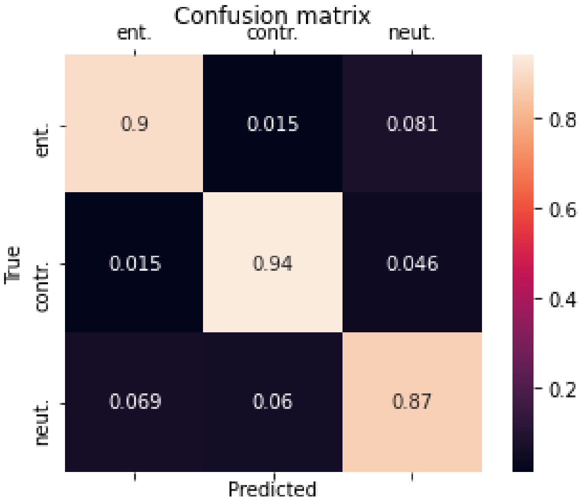

Figure 5: TextAttack's Confusion Matrix on snli-test

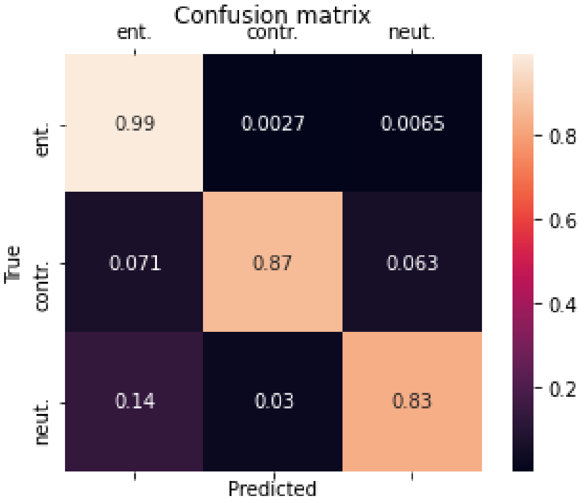

Figure 6: TextAttack's Confusion Matrix on the generated samples

D.1 SNLI

✓ *A woman wearing a ball cap squats down to touch the cracked earth.*
Entailment: *A woman is wearing a hat.*

× *A child wearing a red top is standing behind a blond headed child sitting in a wheelbarrow.*
Entailment: *Two children are sitting in a wheelbarrow.*

✓ *A man playing an electric guitar on stage.*
Contradiction: *A man is playing the piano.*

× *A dog jumping for a Frisbee in the snow.*
Contradiction: *A dog is playing in the snow.*

✓ *Three firefighter come out of subway station.*
Neutral: *Three firefighters are getting off the*

subway station to save a cat.

× *The man in the black wetsuit is walking out of the water.*
Neutral: *The man is going to get out of the water.*

D.2 BoolQ

✓ *The crank sensor can be used in combination with a similar camshaft position sensor to monitor the relationship between the pistons and valves in the engine, which is particularly important in engines with variable valve timing. This method is also used to "synchronise" a four stroke engine upon starting, allowing the management system to know when to inject the fuel. It is also commonly used as the primary source for the measurement of engine speed in revolutions per minute.*
True: *can a crank sensor be used to monitor engine speed*

× *Shower gels for men may contain the ingredient menthol, which gives a cooling and stimulating sensation on the skin, and some men's shower gels are also designed specifically for use on hair and body. Shower gels contain milder surfactant bases than shampoos, and some also contain gentle conditioning agents in the formula. This means that shower gels can also double as an effective and perfectly acceptable substitute to shampoo, even if they are not labelled as a hair and body wash. Washing hair with shower gel should give approximately the same result as using a moisturising shampoo.*
True: *can you use shower gel as a shampoo*

✓ *Barq's is an American soft drink. Its brand of root beer is notable for having caffeine. Barq's, created by Edward Barq and bottled since the turn of the 20th century, is owned by the Barq family but bottled by the Coca-Cola Company. It was known as Barq's Famous Olde Tyme Root Beer until 2012.*
False: *is barq's root beer the same as coca cola*

× *All biomass goes through at least some of these steps: it needs to be grown, collected, dried, fermented, distilled, and burned. All of these steps require resources and an infrastructure. The total amount of energy input into the process compared to the energy released by burning the resulting ethanol fuel is known as the energy balance (or "energy returned on energy invested"). Figures compiled in a 2007 report by National Geographic Magazine point to modest results for corn ethanol produced in the US: one unit of fossil-fuel energy is required to create 1.3 energy units from the resulting ethanol. The energy balance for sugarcane ethanol produced in Brazil is more favorable, with one unit of fossil-fuel energy required to create 8 from the ethanol. Energy balance estimates are not easily produced, thus numerous such reports have been generated that are contradictory. For instance, a separate survey reports that production of ethanol from sugarcane, which requires a tropical climate to grow productively, returns from 8 to 9 units of energy for each unit expended, as compared to corn, which only returns about 1.34 units of fuel energy for each unit of energy expended. A 2006 University of California Berkeley study, after analyzing six separate studies, concluded that producing ethanol from corn uses much less petroleum than producing gasoline.*
False: *does ethanol come from corn or sugarcane*

D.3 Reddit

✓ *i saw a cop pull over a u haul today.... looks like he was trying to bust a move.*
Positive: *i see what you did there*

× *how do you find will smith in a snow storm? you just look for fresh prints.*
Positive: *are you sure its not just hepburn?*

✓ *hedgehogs. why cant they just share the hedge?*
Negative: *is this a joke?*

× *sadly... ..some frogs have been known to kermit suicide*
Negative: *thats the frog, he if he wanted to.*