Distinguishing Fact from Fiction in Contemporary Language Modelling

by

Jingcheng Niu

A research paper submitted in conformity with the requirements for the degree of Master of Science
Graduate Department of Computer Science
University of Toronto

© Copyright 2019 by Jingcheng Niu
Abstract

Distinguishing Fact from Fiction in Contemporary Language Modelling

Jingcheng Niu
Master of Science
Graduate Department of Computer Science
University of Toronto
2019

Statistical language modelling is the essence of all the progress behind natural language processing (NLP). Modern language models can learn a function mapping a sequence of words to a probability, and a vectorized representation of each token (word embedding) simultaneously. Both the probability and the word embedding are proven to be useful for a large variety of NLP tasks, and that triggers many speculations that language models have the ability for some unconventional tasks that require some understanding of the innate knowledge of the human language. However, many of those speculations are based on very limited evidence. Therefore, in this paper, we will exam those speculations rigorously to find out if those speculations are already achieved as facts, or remain as fiction.

For the applications using sentences’ probabilities, Pereira (2000) proposes a connection between the probability of a sequence of words and its grammaticality by showing Chomsky’s (1957) grammatical example being \( \sim 20,000 \) times more probable than its ungrammatical inverse. However this experiment has only two samples and thus lacks statistical significance. We expanded Pereira’s (2000) experiment and found the connection vanishes. Instead, all language models show a strong ability to perceive sensicality, a sentence’s “real-world” utility.

For NLP applications built upon word embeddings and neural classifiers, we primarily investigated the endeavour towards purely end-to-end goal-oriented dialogue systems. We decided to reappraise current neural spoken language understanding (SLU) systems, a critical component in the traditional goal-oriented dialogue system pipeline. SLU systems are most commonly evaluated on the ATIS (Hemphill et al. 1990) benchmark and achieves near-perfect performances. The reappraisal revealed a serious overfitting problem causing the performance of neural ATIS-based neural SLU systems to drop to the same range with an outdated rule-based system. Such a severe issue hinders the readiness for a complete end-to-end goal-oriented dialogue system.

Nevertheless, despite finding that probabilities of sentences are not reliable reflections of their grammaticality and neural slot-filling architectures developed on the most common SDS research benchmark have a serious overfitting issue, we discovered some capabilities of neural language modelling that has been overlooked. Those capabilities can provide significant insights for future research.
Chapter 1

Introduction

Statistical language modelling is the backbone of current NLP systems. Bengio et al. (2003) introduced neural networks to language modelling, allowing them to learn a “probability function for word sequences” and “a distributed representation for each word” (word embedding) simultaneously. Probabilities have been used (since the beginning of statistical language modelling) for generative algorithms to choose the best next word in a sequence. Some other tasks that use language model probabilities include natural language generation (Langkilde-Geary, 2002; Wen et al., 2013; Chen et al., 2015), machine translation (Brown et al., 1990; Brants et al., 2007) and automatic speech recognition (Povey et al., 2011, 2016). Word embeddings are often used to “preprocess” raw text into somewhat meaningful vector representations to allow neural classifiers to recognize patterns easily. Sentence classification algorithms in areas such as sentiment analysis (Agarwal et al., 2011), and slot filling tasks, including spoken language understanding (SLU) (Mesnil et al., 2013; Yao et al., 2014; Zhu and Yu, 2018) and temporal information extraction (TIE) (Chang and Manning, 2012; Bethard, 2013) often apply word embeddings for better results.

With all the progress in the aforementioned areas of application, there has been a great deal of speculation about the potential abilities of statistical language models. In this paper, we analyse two of those speculations. The first is that the probability of a sentence is to some extent a reflection of its grammaticality. The second is that, because of the high performance boost brought by applying word embeddings to the traditional spoken dialogue system (SDS) pipeline, a purely end-to-end SDS can be achieved.

Chomsky (1957) once claimed that no statistical language models can reliably provide grammaticality judgements, at least in the narrow sense of syntactic well-formedness. He argued that all statistical language models would inevitably consider the two phrases “colorless green ideas sleep furiously” (CGISF) and its inverse “furiously sleep ideas green colorless” (FSIGC) equally impossible as they both never appear in any English discourse. However, this particular challenge was met by Pereira (2000) when he showed CGISF is 20,000 times more probable than FSIGC using an aggregative Markov model (Saul and Pereira, 1997). This result inspired several computational linguists to delve further into the relation between language modelling performance and grammaticality, and the result of this research is often adduced as evidence to debate whether the nature of grammaticality itself is probabilistic or categorical (Clark et al., 2013; Lau et al., 2014, 2015, 2017; Sprouse et al., 2015, 2018). During the debate, Lau et al. (2014, 2015, 2017) were the first to claim that language models can model acceptability, a judgement
related to grammaticality that can be elicited from native speakers with no special training in linguistics. Since grammaticality is one of the key factors underlying acceptability, Lau et al. (2017) seem to have concluded that the probabilistic output of language models must be a reflection of grammaticality.

Lau et al. (2014, 2015, 2017) used the empirically measurable concept of acceptability instead of grammaticality because the latter is a theoretical concept that is “not directly accessible to observation or measurement” (Lau et al., 2017). This is a valid reason to prefer acceptability over grammaticality; however, one needs to be very careful when designing experiments around acceptability. As pointed out by Lau et al. (2017) themselves, grammaticality is not the sole underlying factor behind the performance-related concept of acceptability. Other factors such as semantic plausibility and various types of processing difficulties can also affect people’s acceptability judgements. But Lau et al. (2017) did not address this problem in their experimental design. Their way of generating ungrammatical sentences by round trip machine translation might actually create sentences that are grammatical but not semantically plausible. My own study, presented in Chapter 2, attempts to tease apart grammaticality from what I shall call sensicality, which relates to a sentence’s “real-world” interpretability.

Lau et al.’s (2017) decision to sample acceptability rather than grammaticality may not in fact even be necessary. Although it is impossible to observe and measure grammaticality from test subjects, it is actually possible to determine some sentences’ grammaticality. While it may be difficult to reach consensus on the grammaticality of some sentences that only violate some minor linguistic phenomena and are generally interpretable, a great many strings are clearly ungrammatical without consideration of their interpretability, generally on the basis of functional or agreement violations, e.g., “he eat a apple” or “me want pizza.” Clearly grammatical sentences can also be identified by trained linguists on the basis of distributional evidence in spite of their uninterpretability. For example, CGISF is definitely grammatical, while FSIGC is not.

Our investigation of grammaticality and language modelling is not aimed to weigh in on the debate between the categorical and gradient view of grammaticality, nor provide evidence for either side. In observance of the assumptions underlying Chomsky (1957) and Pereira (2000), we will assume that it is categorical. We will instead focus on investigating which of grammaticality or sensicality is learned by unsupervised language models.

Spoken dialogue modelling (or, more commonly spoken dialogue systems, SDS) is another area deserving of investigation. Traditionally, a user’s input is processed by a pipeline of SLU, dialogue state tracking, policy management, and natural language generation (Wen et al., 2017). With all the progress mentioned above on word embedding and neural learning architectures, all the current top-performing components in the SDS pipeline are neural-network-based, which makes the SDS pipeline end-to-end trainable (Wen et al., 2017). Their success has in turn triggered an interest by the community to look into end-to-end, goal-oriented dialogue systems (Bordes et al., 2017; Li et al., 2017; Wen et al., 2018), which requires only an unlabelled dialogue history, without several levels of manually annotation or highly complicated ontologies. The usual representation of the current state of the art is that, while traditional pipelines of neural components work well, end-to-end approaches either surreptitiously still rely on the traditional pipeline (Li et al., 2017) or suffer significant performance drops (Bordes et al., 2017; Wen et al., 2018).

We have instead opted to look at the trade-off between portability and coverage in neural implementations of one component in the traditional pipeline: slot filling. In particular, ATIS-based neural slot-filling systems are a good starting point, because their impressive performance
has been cited as a reason that the ATIS travel reservation domain should be entirely abandoned as “too shallow” Béchet and Raymond (2018). At the same time, we note that Zhu and Yu (2018) discovered a severe overfitting issue causing a 15% absolute performance drop on their attention-based neural model (?) with the very same dataset. In fact, as I will demonstrate in Chapter 3, that overfitting problem is pervasive across neural architectures.

This paper is organized to answer the two questions above: (1) do language models have the ability to provide grammaticality judgements (Chapter 2), and (2) can neural slot-filling models robustly classify users’ inputs (Chapter 3). Finally, in Chapter 4 we will provide a clearer picture of the capabilities and incapacities of contemporary neural language models.
Chapter 2

Language Modelling for Grammaticality Judgement

2.1 Pereira (2000) and Mikolov (2012)

The famous “Colorless green ideas sleep furiously” (CGISF) example of Chomsky (1957) posited a seemingly irreconcilable divide between formal linguistics and statistical language modelling, arguing that every sequence of words not attested in the collective memory of a language’s use would be considered equally “remote” by a putative instance of the latter, regardless of whether the sequence was grammatical (CGISF) or ungrammatical. The example was presented briefly and informally in order to reject statistical language modelling as an alternative approach to the one advocated and developed in greater detail by Chomsky (1957). It was only presented with one other example, the reverse of the sentence, i.e., “Furiously sleep ideas green colorless”, in order to draw a contrast between two nonsensical sequences, only one of which (CGISF) is grammatical.

Pereira (2000) provides an attempt at a refutation by constructing a statistical language model based upon an agglomerative Markov process (Saul and Pereira, 1997), and then observing that CGISF is assigned a probability by the model which is roughly 200 000 times greater than the probability assigned to its reversal. What is perhaps even more stunning than the magnitude of this factor is that, in spite of the far greater quantity of computational resources and electronically encoded text available in the year 2000 relative to 1957, Pereira (2000) was content to evaluate his model on only those two word-sequences. Twelve years later, Mikolov (2012) follows suit by again evaluating a neural language model on only those two inputs, and arrives at a similar conclusion. Neural models had beaten the test.

More recently, a small cottage industry of research has emerged, in which language models, particularly neural language models, have been evaluated for their ability to detect various features of individual grammaticality constraints, such as agreement between non-adjacent subjects and their verbs, finding in many cases near-human levels of performance (Linzen et al., 2016; Bernardy and Lappin, 2017; Gulordava et al., 2018).

There has nevertheless been some scepticism expressed about the ensuing euphoria among computer scientists — mainly by linguists. Sprouse et al. (2015) notes that the trigram model from Lau et al. (2015) assigns different rankings to 10 different permutations of CGISF, depending on the training corpus (e.g., the Wall Street Journal corpus versus an example training corpus taken from Lau et al. (2015)).
Chapter 2. Language Modelling for Grammaticality Judgement

<table>
<thead>
<tr>
<th>Model</th>
<th>BNC</th>
<th>WT103</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pereira</td>
<td>362.19</td>
<td>460.62</td>
</tr>
<tr>
<td>Mikolov</td>
<td>332.04</td>
<td>185.82</td>
</tr>
<tr>
<td>QRNN (regular)</td>
<td>173.57</td>
<td>96.65</td>
</tr>
<tr>
<td>QRNN (TAS)</td>
<td>92.85</td>
<td>34.98</td>
</tr>
</tbody>
</table>

Table 2.1: Perplexity achieved on test sets

Can the scores assigned to these sequences be reliably construed as a regression scale of grammaticality (or perhaps acceptability), if they are so fickle? Chowdhury and Zamparelli (2018) also express concern about the ability of neural language models to generalize to more abstract grammatical phenomena than subject-verb agreement.

2.2 Language Models

We investigated three different types of language models: Pereira’s (2000) original aggregative Markov model (Saul and Pereira, 1997), Mikolov’s (2012) original RNN language model (Mikolov, 2012), and a QRNN-based language model (Merity et al., 2018) that we take to be representative of contemporary models, because it is the best performing model without using any pretrained knowledge. Mikolov’s model is also used by Clark et al. (2013); Lau et al. (2015, 2017); Sprouse et al. (2018) on their research about gradient acceptability. We have excluded language models such as BERT (Devlin et al., 2018), GPT-2 (Radford et al., 2019), and XLNet (Yang et al., 2019), that require pre-training on large sums of internet data, in which the sentence “colorless green ideas sleep furiously” can be found, and is no longer unequivocally nonsensical.

We obtained a publicly available implementation of each of the three language models. The implementation of the tied adaptive softmax (TAS) method used the unusual approach of applying softmax on already softmaxed values. For this reason, we also experiment on QRNN models trained using regular cross-entropy loss functions.

All three models are trained on the BNC (BNC Consortium, 2007) and WikiText-103 (WT103) (Merity et al., 2016). We used the hyperparameters described in (Pereira, 2000) to train Pereira’s model, the hyperparameters described in (Lau et al., 2017; Sprouse et al., 2018) to train Mikolov’s model, and the hyperparameters suggested by the official implementation of the SalesForce model for the QRNN model. Better performance would likely be achieved through more extensive hyperparameter optimization, but our results as shown in Table 2.1 are already comparable to the performance reported in the respective original publications.

2.3 Experimental Design

Previous research on grammaticality/acceptability and language models mainly designs experiments using naturally occurring English sentences, and modifies those sentences based on various individual linguistic phenomena to manually introduce a specific source of ungrammaticality into the sentences. Notable exceptions are The Corpus of Linguistic Acceptability (CoLA) and the Linguistic Inquiry corpus.

---

1 For Mikolov’s model, we used the implementation of https://github.com/yandex/faster-rnnlm that is also used by Lau et al. (2017) and Sprouse et al. (2018).  
2 https://github.com/salesforce/awd-lstm-lm
of grammatical and ungrammatical sentences collected by Sprouse et al. (2013) and Sprouse and Almeida (2012), and used in Sprouse et al. (2018), which are based upon examples found in linguistics publications.

Lau et al. (2014, 2015, 2017) create ungrammatical sentences by round-trip translating natural English sentences. All of those studies observe acceptability, the empirical measurement of how native speakers describe a sentence to be “acceptable”, instead of grammaticality, the theoretical concept defined in the Chomskyan fashion. The result of measuring acceptability and using modified natural English sentences in corpora is that the research subjects creating the acceptability judgements are very likely to be misled by whether a string makes sense. Without proper linguistics training, average native speakers do not consider the difference between not being acceptable because an input does not make common sense, and not acceptable because the input is syntactically ill-formed. To address this problem we briefly return to the CGISF example of Chomsky (1957) to separate grammaticality and what we shall call sensicality.

Given the distinction between grammaticality and sensicality, we can divide input strings into four categories: grammatical and sensical, grammatical and nonsensical, ungrammatical and sensical, and ungrammatical and nonsensical. For our present purposes, there will not be any point to considering sensical but ungrammatical input. Among the remaining three categories, we designed two sets of experiments: a grammaticality test that attempts to distinguish grammatical from ungrammatical input that are both nonsensical, and a sensicality test between sensical and nonsensical input that are both grammatical.

2.3.1 Point-Biserial Correlation Test

Again for consistency with Chomsky’s original conjecture, we have chosen to follow the Chomskyan view of grammaticality as a binary judgement, and so the correct correlation test between probabilities assigned by a statistical model and a discrete, binary judgement is the point-biserial correlation, which has also been used by Sprouse et al. (2018). Like most correlation scores, this figure ranges from +1 (perfect correlation) to -1 (perfect anti-correlation). The midpoint, 0, indicates perfect independence or absence of correlation. A large absolute value would therefore show that either the output probability $p$, of a statistical model, or the value $1 - p$ would be a suitable regression scale through which one could find a separating threshold value in order to reconstruct a binary grammaticality classification.

The point-biserial test assumes balanced positive and negative sample sizes, but the data that will be described in the following section and the data used in the categorical metric experiments in Sprouse et al. (2018) are not balanced. Therefore as argued by Becker (1986), correction is required for “unequal sample size when populations represented by the samples can be assumed to be equally numerous.” The populations of grammatical and ungrammatical inputs satisfy that notion of “equally numerous” and therefore the point-biserial correlation should be corrected. Becker (1986) suggests a formula 2.1 for correction where $r$ is the point-biserial correlation obtained without correction and $p$ and $q$ are the proportions of the two sample sizes:

$$r_c = r / \sqrt{r^2 + 4pq(1 - r)^2}. \quad (2.1)$$

While this correction is intended to preserve the value of the point-biserial correlation when the two samples are of equal size, this clearly is not the case, and so we speculate that the formula 2.2 had been
intended for unbalanced sample size correction:

\[ r_c = r / \sqrt{r^2 + 4pq(1 - r^2)} \]  

(2.2)

### 2.3.2 Grammaticality Experiments

**Colorless Green Ideas Sleep Furiously** Of particular interest to us is the set of all 120 (= 5!) permutations of CGISF. These were independently classified as grammatical or ungrammatical by two linguists who are native speakers of North American English. They then reconciled their judgements where they differed (\( \kappa = 0.568 \)), with reference to the tests provided in [Huddleston and Pullum (2005)](HuddlestonPullum2005), leaving a corpus of 120 assignments of grammaticality (27) and ungrammaticality (93). Other evaluation sets, e.g., that of [Sprouse et al. (2018)](Sprouseetal2018), have been collected using crowdsourced judgements, but this leaves the door open to potential differences between acceptability and grammaticality. We have chosen instead to use trained linguists who are qualified to evaluate grammaticality.

Our first grammaticality experiment is then to differentially assess the grammatical versus ungrammatical CGISF permutations in a given language model. Even with the correction of [Becker (1986)](Becker1986), we cannot deny that the CGISF test is biased because of its unbalanced and limited sample size. We provide it here purely for historical interest, given its centrality to the arguments of [Pereira (2000)](Pereira2000) and [Mikolov (2012)](Mikolov2012).

**Nonsensical Projections of CGISF** Following a method that is akin to an *exquisite corpse* [Sorkin (1991)](Sorkin1991), we are able to obtain a larger set of equal numbers of grammatical and ungrammatical strings that are both nonsensical. First, we obtained the POS sequences of the CGISF permutations that were labelled as grammatical and ungrammatical, respectively. Because there are two adjectives (“colorless” and “green”) in CGISF, we only consider a sequence grammatical or ungrammatical if both adjective substitutions received the same grammaticality judgement from the linguists. Then, we randomly generated strings based on these POS sequences, by inserting vocabulary items that had been independently assigned these POS tags by the Penn Treebank (PTB) [Marcus et al. (1994)](Marcusetal1994), but only those vocabulary items that were also attested in the BNC and WT103. To ensure that these strings were nonsensical, we enforce the constraint that every bigram in the sampled strings does not appear in the Google \( n \)-gram corpus. Using this procedure, we obtained 200 ungrammatical and 200 grammatical strings, which we name CGISF projections (0BG).

Our second grammaticality experiment is then to differentially assess these grammatical versus ungrammatical CGISF projections in a given language model. They are larger samples, balanced and lexically more diverse.

We also sampled another set of 200 grammatical and 200 ungrammatical CGISF projections using a similar procedure, in which the bigram constraint is inverted so that every bigram must exist in the Google \( n \)-gram corpus. We conjecture that these alternative samples, CGISF projections (BG), must at any rate appear to be more sensical to a language model than the samples from the original procedure. We differentially assess them as well.

### 2.3.3 Sensicality Experiments

We obtained a sample of grammatical and sensical strings directly from the BNC validation and test sets that have the same POS tag sequence as one of the grammatical CGISF permutations above. We
Table 2.2: 10-fold cross-validation accuracies by training set on the CoLA in-domain development set.

<table>
<thead>
<tr>
<th>Model</th>
<th>BNC</th>
<th>WT103</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pereira</td>
<td>0.691</td>
<td>0.687</td>
</tr>
<tr>
<td>Mikolov</td>
<td>0.704</td>
<td>0.693</td>
</tr>
<tr>
<td>QRNN (regular)</td>
<td>0.716</td>
<td>0.699</td>
</tr>
<tr>
<td>QRNN (TAS)</td>
<td>0.716</td>
<td>0.699</td>
</tr>
</tbody>
</table>

used ClausIE [Del Corro and Gemulla 2013] to filter out strings that crossed clause boundaries, and then manually excluded strings that remained either because of ClausIE errors, or because of poor BNC POS labelling (most determiners, for example, are labelled as adjectives). This yielded a total of 197 strings, 123 of which come from the validation set.

When assessing this against the 27 grammatical and nonsensical CGISF permutations from Section 2.3.2 we use the first 36 of these that match the CGISF POS sequence exactly. When assessing against either the 200 grammatical CGISF projections (0BG) or the 200 grammatical CGISF projections (BG) from Section 2.3.2 we use the entire set.

2.3.4 CoLA

The Corpus of Linguistic Acceptability (CoLA) [Warstadt et al., 2018] is a collection of 10,657 example sentences from linguistics publications with their grammaticality judgements. It forms an integral part of the General Language Understanding Evaluation (GLUE) benchmark [Wang et al., 2018a]. It must be noted, however, that their linguistic acceptability task is supervised (CoLA was divided into a training set (8551), a development set (1043), and a test set (1063)), with both positive and negative samples. The ungrammatical strings in CoLA have generally been devised to illustrate a specific grammatical defect, and are often but not always sensical. Recent systems trained on these labelled data, e.g., [Liu et al., 2019; Lan et al., 2019; Raffel et al., 2019], are able to attain a reported roughly 0.7 Matthews correlation coefficient (MCC) [Matthews, 1975]. Some of these incorporate language models, but they do not simply use language model probabilities. They are classifiers, purpose-trained for grammaticality.

The performance of Mikolov’s [2012] model, on the other hand, has been reported in CoLA studies as a baseline [Warstadt et al., 2018; Lau et al., 2017]. Warstadt et al. [2018] did a 10-fold cross-validation with it on the CoLA test set, by fitting an optimum decision threshold to the softmax output of each fold in order to assign grammaticality labels. They obtained a 65.2% in-domain accuracy and a 71.1% out-of-domain accuracy. This figure has been cited as a standard for assessing the ability of statistical language models to learn grammar-related patterns in an unsupervised fashion [Lappin and Lau, 2018]. We repeated their 10-fold cross-validation protocol using GLUE’s CoLA in-domain development set and computed the accuracies shown in Table 2.2.

Accuracies are a poor predictor of either point-biserial correlations or MCCs, their discrete analogue, but we reproduce our accuracies here along the way to noting that 71.3% of the in-domain strings and 70.5% of the out-of-domain strings in GLUE’s CoLA development set are labelled positively. These systems perform no better than always guessing positive from the standpoint of classifier accuracy, even though guessing positive with a constant probability (including 0) asymptotically produces a Matthews correlation of zero. Warstadt et al. [2018], to their credit, actually do report an MCC for their study, but it is only 0.253.
The use of either MCC or thresholds casts a veil over the actual floating-point scores that are assigned by the underlying model, however. Recall that Pereira (2000) actually cites the magnitude of difference between CGISF and its reversal when refuting Chomsky’s (1957) claim. Can this view of a model’s performance cast a more favourable light upon them?

2.4 Experiments and Results

Every experiment described in Sections 2.3.2 and 2.3.3 was conducted upon by every language model described in section 2.2. In total, each string receives eight log probabilities from each of the four language models (Pereira, Mikolov, regular QRNN, TAS QRNN), each trained on two different corpora (BNC, WT103).

To address the problem that different language models use different normalization methods, we report the correlations using three different normalization methods: the exponent of the log likelihood (EXP) $e^\ell$ (often considered the sentence’s probability), the exponent of the log likelihood after being normalized by length of the sentence (NORM) $e^{\ell/N}$, and the log likelihood output by the model (LOG) $\ell$.

2.4.1 Grammaticality Experimental Results

Table 2.3 shows the correlations between the grammatical and ungrammatical CGISF sentences. Due to the spelling difference between UK and US English, we evaluated the probabilities of both spellings: colorless and colourless for models trained on the BNC. All models show weak to no point-biserial correlations, with the QRNN model trained with its unusual TAS implementation being the only exception, having a moderate correlation.

<table>
<thead>
<tr>
<th>Model</th>
<th>Score</th>
<th>BNC colorless</th>
<th>BNC colourless</th>
<th>WT103</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LOG</td>
<td>0.35</td>
<td>0.36</td>
<td>0.32</td>
</tr>
<tr>
<td>Pereira</td>
<td>EXP</td>
<td>0.0098</td>
<td>0.034</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.22</td>
<td>0.30</td>
<td>0.34</td>
</tr>
<tr>
<td>Mikolov</td>
<td>LOG</td>
<td>0.025</td>
<td>0.14</td>
<td>-0.11</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.22</td>
<td>0.24</td>
<td>-0.11</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.097</td>
<td>0.16</td>
<td>-0.12</td>
</tr>
<tr>
<td>QRNN (regular)</td>
<td>LOG</td>
<td>0.020</td>
<td>-0.31</td>
<td>-0.045</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>-0.13</td>
<td>-0.21</td>
<td>-0.19</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>-0.027</td>
<td>-0.30</td>
<td>-0.10</td>
</tr>
<tr>
<td>QRNN (TAS)</td>
<td>LOG</td>
<td>0.20</td>
<td>0.42</td>
<td>-0.14</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.18</td>
<td>0.44</td>
<td>-0.14</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.21</td>
<td>0.49</td>
<td>-0.18</td>
</tr>
</tbody>
</table>

Table 2.3: Grammaticality Test: grammatical vs. ungrammatical CGISF permutations

Table 2.4 shows the correlations of the grammaticality test between the grammatical and ungrammatical CGISF projections. As discussed in section 2.3.2, we evaluated CGISF projections with two alternative constraints, one (0BG) that forces every bigram in the projection not to appear in the Google ngram Corpus, another (BG) that by forces every bigram in the projection to appear. As we can observe, no model shows a correlation between the probabilities assigned to the input, and its grammaticality label.
Table 2.4: Grammaticality test on grammatical vs. ungrammatical CGISF projections

<table>
<thead>
<tr>
<th>Model</th>
<th>Score</th>
<th>0BG</th>
<th>0BG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>BNC</td>
<td>WT103</td>
</tr>
<tr>
<td>Pereira</td>
<td>LOG</td>
<td>0.063</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.050</td>
<td>0.073</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.13</td>
<td>0.21</td>
</tr>
<tr>
<td>Mikolov</td>
<td>LOG</td>
<td>0.13</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.085</td>
<td>0.070</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.16</td>
<td>0.15</td>
</tr>
<tr>
<td>QRNN (regular)</td>
<td>LOG</td>
<td>-0.024</td>
<td>0.016</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.052</td>
<td>-0.039</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.022</td>
<td>0.022</td>
</tr>
<tr>
<td>QRNN (TAS)</td>
<td>LOG</td>
<td>-0.034</td>
<td>0.085</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.043</td>
<td>0.0043</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.012</td>
<td>0.064</td>
</tr>
</tbody>
</table>

Table 2.5: Sensicality test on grammatical CGISF sentences vs. BNC

<table>
<thead>
<tr>
<th>Model</th>
<th>Score</th>
<th>BNC</th>
<th>BNC</th>
<th>WT103</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>colorless</td>
<td>colourless</td>
<td></td>
</tr>
<tr>
<td>Pereira</td>
<td>LOG</td>
<td>0.83</td>
<td>0.79</td>
<td>0.80</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.13</td>
<td>0.13</td>
<td>0.11</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.46</td>
<td>0.46</td>
<td>0.23</td>
</tr>
<tr>
<td>Mikolov</td>
<td>LOG</td>
<td>0.83</td>
<td>0.81</td>
<td>0.90</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.17</td>
<td>0.17</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.59</td>
<td>0.58</td>
<td>0.73</td>
</tr>
<tr>
<td>QRNN (regular)</td>
<td>LOG</td>
<td>0.79</td>
<td>0.38</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>EXP</td>
<td>0.14</td>
<td>0.15</td>
<td>0.27</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.52</td>
<td>0.47</td>
<td>0.56</td>
</tr>
<tr>
<td>QRNN (TAS)</td>
<td>EXP</td>
<td>0.71</td>
<td>0.11</td>
<td>-0.39</td>
</tr>
<tr>
<td></td>
<td>NORM</td>
<td>0.21</td>
<td>0.20</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td>LOG</td>
<td>0.59</td>
<td>0.36</td>
<td>-0.18</td>
</tr>
</tbody>
</table>

Table 2.6: Sensicality test between grammatical CGISF projections and all 197 of the grammatical BNC strings. Except for the QRNN (TAS) model, all other models show very strong or, in the case of the regular QRNN model, at least moderate correlations between sensicality and the probability emitted by the language models using the (0BG) constraint. With the (BG) constraint, all language models except for QRNN (TAS) show less impressive correlations. This difference further confirms our hypothesis, as it is easier to tell totally nonsensical sentences apart from sensical ones, than to tell moderately nonsensical sentences apart from sensical ones.
### 2.5 Chomsky (1957) Might be Right

As instruments for testing grammaticality, neither of the original models of Pereira (2000) and Mikolov (2012), nor the more recent neural improvements that we tested, show much promise. Far better, however, is their ability to test for sensicality, which we can describe as a very loose approximation of interpretability. Indeed, statistical language models are often used in computational linguistics, e.g., in parsing, as proxies for semantic representations in order to provide useful disambiguating information. It is this at which they excel, not as indicators of grammaticality.

It is entirely possible that a different class of statistical model could be created in order to test for grammaticality effectively. We do not believe that the evidence provided to date therefore proves Chomsky’s (1957) original claim. This task, however, seems to be at odds with the state of the art in language modelling.
Chapter 3

Neural Spoken Language Understanding Architectures

3.1 The Task of Spoken Language Understanding

Spoken Language Understanding is typically characterized as a sequence labeling problem in which certain tokens are identified as fillers that contribute argument values to a meaning representation through “slot” positions in the utterance. Wang et al. (2011) first used conditional random fields (CRF) for slot filling. A few years later, inspired by the success of recurrent neural networks (RNN) in language modelling, Mikolov (2012) and Mesnil et al. (2013) developed the first RNN slot filler that achieved a relative error reduction of 14%. Subsequently, different variations of RNN such as LSTM (Yao et al., 2014) were developed for slot filling, followed by encoder-decoder models that could utilize information from the entire sentence (Kurata et al., 2016), both of which avail themselves of an attention mechanism (Li et al., 2018). As recently as Wang et al. (2018b), Deep Reinforcement Learning (DRL) has been proposed as a way to refine encoder-decoder models on sparsely distributed tags; this has achieved the highest reported performance so far.

This development has taken place in parallel, however, with work that has used qualitative error analyses to cast doubt on the continued use of ATIS as a benchmark for progress in slot filling. Most recently, Béchet and Raymond (2018) conclude that ATIS is simply too “shallow” to offer anything of additional substance for DNN-based architectures to achieve, formulating a three-way taxonomy of errors in the reference annotation for the ATIS corpus that account for roughly half of the remaining errors still faced by state-of-the-art slot filling models. Even prior to the recent popularity of neural architectures, Tur et al. (2010) cited a problem with earlier n-gram-based modeling approaches, which tended to fit every utterance into a known sample without regard to domain knowledge or aspects of global context that could override local n-gram contexts.

3.2 The ATIS Dataset

The ATIS Spoken Language Systems Pilot Corpus (Hemphill et al. 1990) contains utterances of users asking flight-related questions that could be answered by a relational query search from the ATIS
database. For the task of slot filling, only the text part of the corpus is used. Generally, 4978 Class A utterances in the ATIS-2 and ATIS-3 corpora are used as the training set, and 893 utterances from ATIS-3 Nov93 and Dec94 are selected as the testing set. Developers may randomly split the 4978 utterances into a training set (for us, 90%) and a development test set (10%).

The text data are converted to the format suitable for the slot filling task. Each token of an utterance is considered to be a potential slot, and each slot should contain a tag, with an optional Concept part and a mandatory Named Entity (NE) part, in the In/Out/Begin (IOB) format. Mesnil et al. (2013) converted the relational queries into that format using an automatic process. Figure 3.1 is an annotated example. The entire dataset contains 9 distinct concepts and 44 NEs that yield 127 total possible tags.

For ease of reference, we number both the training and test sets in lexicographical order here, starting from 0.

3.2.1 Errors in ATIS

Béchet and Raymond (2018) identify three sources of error: annotations missing slots entirely or transposing labels, for example, between departure and arrival cities; determinately reading an utterance that is naturally ambiguous (no system should be penalized for having guessed another valid reading); and labeling only the first of several instances of the same NE in the same utterance (systems that label more than one are penalized). 1.14% of the slots in the training set are incorrectly labeled overall, as are 2.29% of those in the test set. These percentages are significant, given that state-of-the-art systems commonly report error rates of between 1.2% to 6%. Note that there are almost twice as many errors in the test set as in the training set on a percentage basis. About half of these are ambiguous slots arising from the use of “UNK” for hapax legomena. In these 46 cases, the slot cannot be determined without knowledge of what the word formerly was. Most egregiously, five of utterances 785 - 791 are “What is UNK?” and the other two are “What is a UNK?”.

The test set is unique in other respects. Six of its slot labels (B-booking_class, B-stoploc.airport_code, B-flight, I-state.name, I-flight_number and B-compartment) are not found in the training set. Except for B-stoploc.airport_code, the other five are NE annotation errors. The test set also handles the word noon differently: four instances are treated as a period of day, whereas all occurrences of noon in the training set are treated as a time.

3.2.2 Taxonomy

We have created our own error classification. Not all of these classes map onto one of the three in Béchet and Raymond (2018). The taxonomy and errors were labelled independently by two annotators, who
Table 3.1: Annotation Mistakes by Taxonomic Class.

<table>
<thead>
<tr>
<th>Split</th>
<th>Train utterances</th>
<th>Test utterances</th>
<th>Train instances</th>
<th>Test instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOB</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Selection</td>
<td>22</td>
<td>22</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Missing</td>
<td>29</td>
<td>30</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Concept</td>
<td>72</td>
<td>120</td>
<td>28</td>
<td>46</td>
</tr>
<tr>
<td>NE</td>
<td>12</td>
<td>13</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>UNK</td>
<td>46</td>
<td>46</td>
<td>46</td>
<td>46</td>
</tr>
</tbody>
</table>

were then forced to reconcile where they disagreed.

- **Incorrect IOB Segmentation**

  In the test set, 309: “List airports in Arizona, Nevada and California please.” unifies the two states Arizona and Nevada into one slot, and was annotated as B-state_name and I-state_name. Corrected.

- **Wrong Word Selection**

  Some slots select the wrong words. Utterance 1374: “I need information on ground transportation between airport and downtown in the city of Boston.” labels the whole phrase city of Boston as toloc.city_name, whereas elsewhere only Boston is labeled. Chose dominant word sequence.

- **Missing Labels**

  Words that should be annotated are not (equivalent to label, O, i.e. outside of any slot). For example, in 29: “All am flights departing Pittsburgh arriving Denver.”, the abbreviation “am” should have been labeled B-depart_time.period_of_day, but was not annotated. Annotation added.

- **Concept Mistakes**

  These are the most prevalent annotation error. For example, “Denver” in 40: “All flights before 10 am Boston Denver.” was annotated as B-fromloc.city_name, where it should have been toloc. Includes ambiguities that are not consistently annotated (we chose the dominant annotation) as well as unambiguous fillers that bear more than one concept role (which the annotation standard does not permit; these were discarded).

- **NE Mistakes**

  These appear in both the training and the test set. For example, in utterance 29: “Flights from Denver to Westchester county New York weekdays.”, New York means the state of New York, not New York City, but its NE was labeled as a city name instead of state name. Corrected.

- **Out-of-Vocabulary (UNK)**

  These are found in the training set (e.g., 4394: “What is UNK?”) and the test set, as discussed above. Discarded the utterance.
3.3 Experiments

3.3.1 Rule-based Grammar

In addition to repairing the ATIS annotations, we developed a rule-based grammar for use as a baseline and domain-specific knowledge source, particularly of time and location phrases. We used the Attribute Logic Engine (ALE) (Carpenter and Penn, 1994), a grammar development system and logic programming language based upon typed feature structures. ALE compiles grammars into an all-paths chart parser that produces phrase structure forests. We use the logic programming extension to project words into individual IOB slots, given a parsing chart.

The grammar does not generate a spanning parse for utterances with multiple sentences (e.g., 3612: “US air 269 leaving Boston at 428. What is the arrival time in Baltimore?”). These, as well as single sentences for which no spanning edge is found, are instead projected using a covering of edges that is selected with the greedy algorithm shown in Algorithm 1. This algorithm prefers longer spans to shorter spans and breaks ties by selecting one edge uniformly at random.

Algorithm 1

\[
\text{Greedy}(\text{edges})
\]

\[
\begin{align*}
\text{long} & \leftarrow \text{a longest edge in edges} \\
L & \leftarrow \text{edges finish before long} \\
R & \leftarrow \text{edges start after long} \\
\text{return Greedy}(L) + \text{long} + \text{Greedy}(R)
\end{align*}
\]

The grammar uses 601 lexical entries (one or more for each of the 573 word types in ATIS), 643 feature structure types, 22 features and 330 phrase structure rules. The feature structure types that we defined were for two major purposes: 168 syntactic types that label the nodes of a parse tree, and 475 types that declare appropriate values for features. Every syntactic node label has features that refer to a list of slot fillers (TAGS) and a list of tokens (WORDS) in the subtree at which it is rooted.

Among the 330 grammar rules, 65 rules are used to capture multi-word expressions (MWE), which ALE does not otherwise support. Only 161 rules are designed specifically for ATIS, with the remaining 104 being general rules of English grammar. Nouns are further divided into different ATIS-specific slot values such as cities, states and airlines. Verb semantics are categorized based on their indication of direction. “Directional” verbs such as ‘depart’ and ‘land’ are distinguished from the others. Prepositions are further split into timerelated, direction-related, location-related, costrelated, and other special functions.

3.3.2 Neural ATIS Systems

<table>
<thead>
<tr>
<th>Model</th>
<th>Reported F1 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNN (Mesnil et al., 2013)</td>
<td>93.98</td>
</tr>
<tr>
<td>LSTM (Yao et al., 2014)</td>
<td>95.08</td>
</tr>
<tr>
<td>Encoder-Decoder (Kurata et al., 2016)</td>
<td>95.66</td>
</tr>
<tr>
<td>Encoder-Decoder with focus (?)</td>
<td>95.79</td>
</tr>
<tr>
<td>Self-attentive BiLSTM3 (Li et al., 2018)</td>
<td>96.35</td>
</tr>
<tr>
<td>Encoder-Decoder DRL (Wang et al., 2018b)</td>
<td>97.86</td>
</tr>
</tbody>
</table>

Table 3.2: Reported Performance of Neural SLU Models.
## Table 3.3: Experimental Results.

We reimplemented or, in one case, obtained from the authors code for the models mentioned in Table 3.2, which also shows the F1-scores reported there. The hyperparameters were set to those that are reported in the papers as having the best performance. Each model was trained for 100 epochs, and then the epoch with the highest development test set performance was chosen to evaluate on the ATIS test set. We were unable to reproduce comparable figures for the DRL scheme of Wang et al. (2018b) and so it has been excluded from our analysis.

### 3.3.3 Experimental Results

**Reappraisal of ATIS-based Systems**  Our own results are reported in Table 3.3. The column, Test, reports results on the original ATIS test set. Fixed reports on the ATIS test set after all of the repairs mentioned in Section 3.2.1 were fixed. UNK reports on the ATIS test, with all repairs except the exclusion of utterances with ambiguous occurrences of UNK. Finally, X reports on a corpus, which, similar to the Corpus X test set presented in Zhu and Yu (2018), modified the ATIS test set by replacing every NE with a different NE from the same epistemic class in a travel domain ontology defined by them, such that the new NE has never occurred with the same concept. For example, the city “Toronto” appears as a fromloc.city.name and toloc.city.name, but never as a stoploc.city.name in ATIS. So “Toronto” is used in Corpus X wherever the reference annotation requires a stoploc.city.name. Zhu and Yu (2018) did this in order to experiment with a neural architecture that trains first on a coarse classification and then fine-tunes to the ATIS reference annotation in a later step, but the F1 drops on Corpus X are a result of overfitting in which the model effectively learns that Toronto is never a stopover.city. Our Corpus X differs from their Corpus X test set only in that we first corrected their ontology in light of our taxonomy of annotation errors.

Because the rule-based parser uses an all-paths algorithm, its F1-score is reported in three ways. Rand(om) uses the greedy Algorithm 1 in which ties are broken at random. Scep(tical) only counts successes that every member of a tie produces. Cred(ulous) counts successes that any member of a tie produces. The sceptical and credulous scores bracket the possible parse selection strategies. Full Parse
restricts the evaluation to those utterances (the percentage of which appears in the final row) for which one or more complete parses was found by the rule-based grammar.

**Entity-only Evaluation** To locate the source of the overfitting problem, we evaluated the RNN and the LSTM model again for NER only. We removed all the concepts in ATIS, keeping only the named entities. For example, a `B-from_loc.city_name` will be simplified into a `B-city_name`. This procedure was repeated in the Corpus\_X set. The two models are trained on the “no concept” ATIS training set and then evaluated on both the “no concept” ATIS test set and the “no concept” Corpus\_X set. To demonstrate the trend of the overfitting issue, we reported the test set F1 score after every epoch of training. As a control group, the performance of the two models on the unmodified ATIS corpus is also evaluated. Particularly, both models are trained on the original ATIS training set; the ATIS test set and Corpus\_X set performances are reported after each epoch of training.

Figure 3.2 shows the result of this experiment. The overfitting issue is still obvious, as on both models the performance on the original Corpus\_X set is consistently worse than the performance on the original ATIS test set. However, as soon as concepts are removed, the overfitting issue disappears. The performance on the “no concept” ATIS test set and the “no concept” Corpus\_X set are indistinguishable.

Also, one may argue that the modification of the complexity of the task could be the contributing factor to the disappearance of the overfitting issue, rather than NER or the absence of a conceptual distinction per se. To address this concern, we perform another set of entity-only evaluations that does not modify any of the corpus annotations, but ignores the conceptual part of the tag during evaluation. For example, if a token labelled as `B-fromloc.city_name` should be a `B-toloc.city_name` or `B-city_name`, we consider the labelling of the token to be correct. In this way, the task still retains its complexity as the structure of the tags is unchanged.
Figure 3.3: The differences between the F1 scores evaluated on ATIS Test and Corpus_X

Figure 3.3 shows the result of the experiment. As we can observe, except in the beginning of the training period (\(\sim 10\) epochs for RNN and \(\sim 20\) epochs for LSTM) and a few outliers, the difference between ATIS Test and Corpus_X when evaluated without concepts is always smaller than the difference when evaluated with concepts. Since a difference between ATIS Test and Corpus_X can be seen as a sign of overfitting, this experiment corroborates our hypothesis that the overfitting issue is reduced when only take name entities are taken into consideration, even without changing the complexity of the task.

### 3.4 Result Analysis

One might expect that recent neural approaches could use their word vector representations to generalize better to out-of-domain utterances than the earlier models that Tur et al. (2010) referred to. In fact, the results of the previous section on Corpus_X clearly indicate that these recent architectures overfit their language models to filler content itself, overshadowing any potential gain from better contextual inference. ATIS is “shallow” in that it offers only a small amount of training data and an overall lack of lexical and syntactic variety. What is even more telling is that the performance of these recent architectures on Corpus_X is so bad that it falls within the F1 range of our rule-based grammar. When used for NER, as opposed to slot-filling, these architectures show both a promising lack of overfitting, and a near perfect performance as NERs. The advantages promised by nascent statistical approaches to natural language understanding when rule-based grammars were still in vogue were primarily centred around: (1) portability and (2) coverage. As to portability, recent neural approaches to a corpus as small as ATIS necessarily surrender a certain amount of it for the sake of jointly modeling knowledge of language and domain-specific knowledge - a laudable goal on substantially larger training sets. Our experience with industrial partners suggests, however, that extensibility, in which developers wish to
roll out the same domain but to a further extent, such as with more cities, more airports etc. in the

case of the ATIS corpus, is of qual importance to them as portability to different domains. There, a

rule-based grammar would only be the preferred option if augmenting the filler vocabulary were all that

was at stake. It would not be the preferred option if the extension were in the direction of much greater

syntactic variety.

That brings us to coverage. The relative error reduction observed after fixing the ATIS annotation
generally fails to attain the 50% predicted by Béchet and Raymond (2018). Nevertheless, those repairs
put the neural models close to the rule-based grammar’s range on utterances for which it generates a

full syntactic parse. Our greedy parse selection approach is necessitated by the mere ∼ 80% coverage

of the ATIS domain with our rule-based grammar. Neural parsing architectures do exist, and already

provide better coverage than 80%.

These arguments taken together suggest that, while there may be very little remaining reward to

addressing the slot-filling problem with ATIS, there is still a very perceptible parsing problem, even

on a corpus of ATIS’s size and lack of syntactic variety. ATIS is not syntactically annotated; to our

knowledge, no syntactically annotated corpus in the travel reservation domain exists. The development

of such a corpus, the transfer of learning between parsers on different domains of this size, and the

appropriation of such a portable parser to slot filling, remain the most promising direction of further

research for slot filling, in our view. In this endeavour, ATIS may still play a very prominent role.
Chapter 4

Discussion

Disappointingly, a sentence’s probability does not reliably reflect its grammaticality in contemporary language models, as we showed in section 2.4, and neural slot-filling architectures, across the board, have a very grave overfitting issue on one of the most commonly used benchmarks in SDS research. While in the former case, it seems fairly clear that a language model is simply the wrong tool for the job, in the latter case, neural slot-filling, and by extension, end-to-end neural dialogue systems, may very possibly emerge as the asymptotically preferred approach on an ever-expanding scale of available annotated dialogue-domain resources, and yet the fact remains that no dialogue domain with a sufficiently large training set currently exists for neural models to demonstrate this hypothetical potential. SNIPS (Coucke et al., 2018), the largest publicly available annotated dataset as of this writing, is only about three times larger than ATIS, and its domain is very heterogeneous. Even against a much older rule-based approach on even a small but crucial component of a traditional dialogue pipeline, neural slot-filling has not, by our measurements, clearly pulled into the lead position. What is even more surprising is that both of these fictions have been sustained by an almost inexplicable unwillingness on the part of the research community to conduct proper evaluations.

On the other hand, our experiments have revealed some capabilities of neural language models that have been overlooked. While not being correlated to grammaticality, the probabilities generated by language models appear to be sharp reflections of a sentence’s sensicality. This ability is more practical because the theoretical grammaticality and the empirically measured acceptability are not always useful for real-world applications. And that does seem to have improved with successive generations of models.

We also discovered that neural slot-filling systems are robust and accurate in the category of NER systems, but suffer the overfitting issue when it comes to learning domain-sensitive rules of inference. Traditional rule-based systems, on the other hand, are unsurprisingly less robust and are proven by our experiments to still be competitive. Rule-based systems are highly interpretable and customizable; both of these remain desirable for industrial applications, as suggested by industrial partners that I have collaborated with. Therefore, combining the neural NER with a rule-based logic module can result in SLU systems that are both robust and accurate. The concept of logic-embedded neural networks [Li and Srikumar, 2019] is still emerging, but is attracting a great deal of interest.

We are also interested in examining the overfitting problem in other areas of research such as TIE. The task of TIE typically features a separation of NER and logic. The task of events and time expression (TIMEX) extraction is more NER focused, while the task of relation extraction is purely logic-based.
The performance difference between the two tasks further confirms our hypothesis: in the TempEval-3 shared task [UzZaman et al. (2013)], state-of-the-art event and TIMEX extraction systems have $\sim 80\%$ and $\sim 90\%$ F1 scores respectively, while state-of-the-art relation extraction only achieves a $36.26\%$ F1 score for identification and a $56.45\%$ F1 score for classification.
Bibliography


