PhD subject
Multi-task Learning for Text Normalization, Parsing and Machine Translation

Benjamin Muller

October 17, 2019

1 Introduction

The research that will be pursued within these PhD studies lies in the field of Natural Language Processing (NLP). Since 2011, the successes of Deep Learning (DL) models have spread out to most popular NLP tasks. Interestingly, the Deep Learning framework does not only lead to better performances for already existing pipelines, it also provides a very general framework for designing Multi-task learning systems.

These research will be developed in the context of two broader projects. The *SoSweet* project [Magué et al., 2016] which aims at modeling sociolinguistic aspects of social-media linked to language variability and the *Parsiti* project which aims at developing more robust and context aware models for parsing and machine translation, specifically for user generated content.

Recent progress in NLP are conditioned to two key elements. First, they require relatively large amount of annotated data for the given task. Second, the resulting system will perform very well only if the test data is "somehow" not too different from the training data. In other words, State-of-the-art (SOTA) NLP systems remain constrained in the domain, language level and genre on which the model has been trained. The path that will be taken to tackle this challenge is *Multi-task learning*. Indeed, by bringing together various tasks such as machine translation, syntactic parsing and text normalization, and by designing algorithms able to efficiently make use of their complementarity, we aim to build more robust NLP systems.

In order to tackle specifically the sensitivity of the designed pipeline to diversity in the data, we will follow two main paths. The first one focuses on the training procedure. By extending the *Adversarial Training* paradigm to a more linguistically grounded approach, we will aim at making our models more robust to noise. The second one tackles the model itself. By modeling more explicitly the context, whether it is linguistic or extra linguistic (other form of user generated content for instance), we will aim at making our model more adaptable and therefore more robust.

Before describing the research that will be pursued, we will start by defining a few key concepts useful for this proposal. We'll next present the recent striking publications on the subject. Finally, we will present a brief work-flow which describes how the research will be organized and prioritized.

2 Preliminary definitions

2.1 Modeling toolkit

We start by introducing the modeling toolkits and its specificities. As mentioned above DL systems have become very popular in NLP. What is Deep Learning and what is fundamentally different compared to shallow methods ? In a nutshell, a deep learning model is a cascade of simple non-linear computing functions trained in an end-to-end fashion. At its core, a deep learning model learns a continuous representation of the input variables in regard to the specific task we aim to solve. In other words, a deep learning model performs the features engineering which was traditionally done upstream in former NLP pipelines. This specificity paved the way for the recent successes of DL in NLP and other fields. As seen in [Collobert et al., 2011] the DL framework allows to design much more standard model able to solve diverse NLP tasks compared to task-specific models that were necessary before.

More specifically, regarding Machine Translation, this success is closely related to the emergence of a new framework : the encoder-decoder paradigm [Cho et al., 2014b]. Encoder-decoder allows to design tasks that takes sequential data both at inputs and outputs. In short, the encoder is a stack of computing units, usually a recurrent neural network, that learns a representation of the source variable. This representation is then fed to the decoder along with the target variable which is predicted in a sequential way. The key point here is that this architecture is trained in an end-to-end fashion. This paradigm quickly found applications outside of MT as in Question-Answering and Image-Captioning. More recently, it was used to tackle syntactic and semantic parsing as seen in [Liu and Zhang, 2017] and [Dong and Lapata, 2018].

2.2 Tasks

Let us now introduce three tasks at the core of the research that will be pursued.

2.2.1 Text Normalization

First, text normalization. Normalization consists in transforming a source sentence or document into a canonical form. We highlight two domains of NLP for which normalization is very helpful. The first one is historical document processing. Historical documents often requires a normalization step. Indeed, for most languages and for many periods in time (including the present for some languages and communities), there are no standardized spelling. This stands as a tough challenge for modern NLP systems as one of the first processing step is to assign a token to an unique index. Another domain of great interest for NLP is *User Generated Content* (UGC) (as Eric Schmidt highlighted, "every two days we produce more information from the dawn of civilization up until 2003", most of it being textual). As well as historical documents, most of textual UGC requires a robust normalization step. As seen in Table 1, processing this data stands a a real challenge for translation and syntactic parsing both at the lexical level and at the syntactic level.

Researchers have tackled normalization in various ways. The simplest approach is to see it as a spell checking problem and to assign unknown words to the closest one in a known vocabulary using the edit distance [Levenshtein, 1966]. Leveraging word embedding methods, [Sridhar, 2015] managed to design a purely unsupervised approach for learning a normalized lexicon as well as a normalization procedure.

Still, Table 1 illustrates that lexicon irregularities are not the only issue of UGC. Indeed, those irregularities might be seen as an overall dialect with its own regularities. Therefore, only a rich modeling of the context might be able to solve the extreme noisiness of UGC, whether it is for parsing or translation.

Following this necessity, a few attempts managed to apply the encoder-decoder paradigm for normalization. Intuitively, we can see normalization as the generation of a constrained sequence of words. As seen in [Bollmann and Søgaard, 2016], the encoder-decoder paradigm is a very promising approach for modeling context more efficiently and result in more robust normalizers. Shortly, their model consists in normalizing documents in German from many regions and periods of the past to standard modern German. This contribution highlights one of the challenges of applying encoder-decoder to normalization. Those architectures usually require lots of training samples. To overcome the issue, the authors designed their model as a multi-task learning problem. Indeed, they treated the transformation of a document written in a period and region to its "copy" in another region and period in the past as their auxiliary task. By doing so, the extended

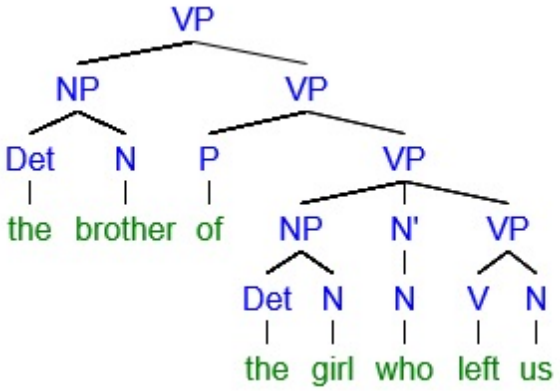

Figure 1: Constituency tree

drastically the amount of training data and manage to improve significantly the performance on the main normalization task.

Encoder-decoder are also now very useful in the field of Grammatical Error correction (GEC) a subfield of normalization. As described in [Tao Ge, 2018],the State-of-the-Art model in GEC (on the JFLEG dataset) is now an encoder-decoder architecture trained on a reinforcement learning policy based on the GLEU score. This contribution also demonstrates that switching to a sentence level objective function (here the GLEU score) might be a successful path.

Those successes demonstrates that the encoder-decoder paradigm might be well suited for dealing with extreme noise in the training data whether it is for a strict normalization task or for other objectives.

2.2.2 Syntactic Parsing

We then present syntactic parsing (or parsing). Parsing consists in extracting the grammatical relations of *syntax tokens* from raw sentences following a given grammatical formalism. We briefly describe two main formalisms.

First, dependency parsing which structures sentence as a directed binary grammatical relations (head-son relations) that link tokens. We can name two family of dependency parsing algorithms : graph-based algorithms and transition based algorithms. On one hand, graph-based algorithms model each sentence as a fully connected, oriented and weighted graph of tokens. Each weight represents the probability of token A being the head of token B. Based on this graph, a decoding algorithm finds the tree belonging to the graph that maximizes the global probability. On the other hand, transition-based algorithms model the parsing task as a sequential prediction of *transitions*. Each transition consists in a specific operation applied to the sentence (seen as a sequence of tokens) and an alternatively filled and emptied stack of tokens.

Interestingly, despite their very different nature,

(@rigolboche)

Original Source	Bing© Translation
→ T'as vu il l'a bien cherché wsh #AperoChezRicard	→ You have seen sought it wsh #AperoChezRicard
→ +10000, shah!	→ +10000, shah!
→ tabuz, lavé rien fé	→ tabuz, washed anything fe
→ ki ca ? le mec ou son chien ?	→ ki ca? the guy or his dog?
→ Wtf is wrong with him ? #PETA4EVER	→ Wtf is wrong with him ? #PETA4EVER
→ ki ca ? le chien ? looool	→ ki ca? the dog? looool

Table 1: Typical social media thread initiated by a seed photo and its automatic translation - *Inspired from a real conversation during the last Paris demonstration. Bing was used as it is the official MT engine for Twitter and Facebook.*

these algorithms compete closely in dependency parsing. As seen in [Daniel Zeman, 2017], among the first four teams to the CoNLL 2017 shared task ranked based on their averaged *Labeled Attachment Score*, one used only a graph-based model (the winning system), one used a strictly transition-based model while the two others used some kind of ensemble models that mix transition and graph approaches. Still, we highlight that the State-of-the-art in a multilingual setting is so far a graph model [Dozat et al., 2017] with a mixed character and word based model stacked to a standard recurrent encoder and a bi-affine output layer.

The second popular formalism is constituency parsing. As seen in [Jurafsky, 2000], the idea behind this formalism is that group of words within a sentence can form independent units, the so-called *constituents*. Constituency parsing aims to find these constituents, labeled them and organize them as a hierarchical structure or a tree. Figure 1 is a simple example of what a constituent tree is. Very recently, [Kitaev and Klein, 2018] reached state-of-the-art at constituency parsing on the Penn Tree-bank using an encoder-decoder architecture that includes a self-attention layer [Lin et al., 2017].

2.2.3 Machine Translation

The third task that we will developed during these research is Machine Translation (MT). Four years ago, MT embraced neural networks, leveraging the encoder-decoder paradigm. As seen in [Cho et al., 2014a], in a nutshell, Neural MT (NMT) successfully models the distribution of the target sequence (usually a sentence in language B) conditioned on the source sequence (usually a sentence in the language A). To do so, NMT includes attention mechanisms [Cho et al., 2014a], which allows to encode more efficiently the source space by focusing in a flexible way to the relevant tokens in the source sentence for generating the target tokens.

MT is challenging from a machine learning perspective for many reasons. As described in details in [Ott et al., 2018], this task faces inherent uncertainty. First, for its *one-to-many* nature. Indeed, several sentences in the target language might be satisfying translations of a given source sentence. Moreover, some language pairs might lead to an *underspecification* problem. Indeed, translating the pronoun *it* from English to French might be impossible if not enough context is provided. We highlight here that those challenges are, almost always, not taken into account in the standard encoder-decoder NMT model trained on parallel corpus of pairs of sentences. The other major challenge of machine translation, which is also related to its one-to-many nature, is evaluation. Indeed, being able to evaluate systematically with a human-like sense a translation remains an open problem. These last years and despite its inherent biases, the BLEU score became one of the most popular evaluation metric in the MT community [Papineni et al., 2002]. Based on this metric, we can highlight the best performing systems for given pairs of languages.

Very recently, [Chen et al., 2018] managed to analyze in details the strength and weaknesses of encoders and decoders of State-of-the-art systems. Based on this analysis, they designed an architecture that outperformed all former systems on English to French and English to Dutch (on WMT 2014 datasets). Briefly, they designed a new recurrent architecture called RNMT+ that they used within their encoder and as their decoder. RNMT+ is an upgrade of the former GNMT architecture [Wu et al., 2016]. They enhanced it with more parameters (more LSTM layers, most unidirectional LSTM were also replaced by bidi-

rectional LSTM), by stabilizing training using layer normalization [Ba et al., 2016], adaptive gradient clipping and label smoothing [Szegedy et al., 2016], and by adding residual connections. Then, they combined the RNMT+ with the Transformer encoder [Vaswani et al., 2017] in a cascading manner for French and by concatenating them together for Dutch. In details, they reached 41.0 in BLEU for translation to French and 28.5 for Dutch.

2.2.4 Multi-task learning

We now present a few striking trends in multi-task learning applied to NLP.

As described in [Caruana, 1997] in the context of neural networks, multi-task learning was first introduced in the 90s. More recently, [Luong et al., 2015] was the first to show that the encoder-decoder paradigm could leverage a multi-task learning approach. Indeed, the authors demonstrated that machine translation could benefit from auxiliary task such as image captioning and syntactic parsing. Using this approach, they also reached, at the time, SOTA in constituency parsing.

Since then, following this success, there has been an extensive literature in applying multi-task learning within the encoder-decoder paradigm. Those contributions usually follow one of the two following goals. On one side, reaching competitive performance on one or several of the tasks. As seen in [McCann et al., 2018], some contributors even attempted to perform 10 NLP tasks. On the other side, making use of this setting to learn richer representation of the input sequences, usually sentences. As seen in [Brunner et al., 2018] and [Subramanian et al., 2018], the idea is to learn some kind of "universal" representation of sentences. Before reaching the holy grail, those contributions are making interesting progress on how to actually evaluate representation at the sentence level, whether it is on syntax aspects, or semantic aspects.

3 Research paths

3.1 Scope

In Machine Learning, many experiments showed that multi-task learning does not only provide systems that can actually do multitasking, but in many cases, it was shown that multi-task learning can also improve the performance of one or several of the involved tasks. As described in [Caruana, 1997], this is partly due to the fact that making the model predict various and complementary outputs leads to a richer and less prone to over-fitting representation of the input variables.

Pushed by such an insight and by the flexibility of the encoder-decoder paradigm, the research will attempt to design more robust and adaptable multi-task learning systems regarding three, somehow complementary, tasks : Machine Translation, Syntactic Parsing and Normalization.

3.2 Resources

To pursue a such purpose, we first highlight the in-house available resources that will be available in addition to the existing open-source ones.

Within the SoSweet project, we have access to around 200 millions of tweets in French as well as a few millions of tweets in several other languages (mainly English). Moreover, we have access to 2 tree-banks (of around two thousands sentences each) (formated as CoNLL-U) of user generated content [Seddah et al., 2012]. Moreover, we have access to a few thousands of parsed trees of sentences in the *Arabizi* dialect. We also have the arabic transcriptions of a subsample of this data. Finally, we also will use a few thousands of English user-generated content extracted from on-line forums.

In addition, we will make use of the open source data. For machine translation, we will make use of the popular WMT 2014 datasets [Bojar et al., 2014]. For syntactic parsing, we will make use of the Universal Dependency project [Nivre et al., 2017] that aggregates treebanks of over 60 languages.

3.3 Architectures

Syntax for Machine Translation

When we come to think about using syntax for machine translation, many paths become possible.

We might want to dig into the syntax of the source language to help translation into the target language, as seen in [Hashimoto and Tsuruoka, 2017]. To paraphrase Leon Bottou, "We [humans] use structure in language to understand unknown things [not already known ones]". Making a model jointly learn syntax of the source language and translation to a target language could constrain the model to look for a syntactic structure in the source sentence. Intuitively, this should lead to translations more robust to rare or unknown words as the model could take advantage of the encoded syntax to infer their roles.

Another complementary approach would be to make the model predict the syntax of the target sentence along with the translation. This approach was recently successfully implemented in [Eriguchi et al., 2017] and [Aharoni and Goldberg, 2017]. Intuitively, a such approach makes model generate syntactically grounded predictions and therefore improve the quality of the translation. More specifically [Aharoni and Goldberg, 2017] were able to show that leveraging syntax with a *Recurrent Neural Network Grammar* [Dyer et al., 2016] was improving significantly the BLEU score for most language pairs.

Normalization as translation

Intuitively, normalization can be seen as a translation task, where the target language would be a canonical form of the source language. As mentioned in 2.2, recent contributions successfully attempted to develop such idea. Indeed, they manage to include normalization into an encoder-decoder architecture.

Normalize and translate with syntactic insights

Following the two ideas described above, we could design a single encoder-decoder with attention mechanisms and shared-weights that would translate, normalize and parse. A such architecture would learn, if properly trained, a common representation of the source sentence that would encode syntax. Implicitly constrained by the normalization task, this representation should also be robust to noise in the source language.

GAN with syntax

Another aspect of the research that will be pursued focuses on the objective function. As described above, MT is a very challenging task. One of its biggest challenges is the scoring step, or in other word, how to compare a predicted sentence and a gold translation. Recent models whether they use usual cross-entropy loss or directly optimize BLEU Scores with reinforcement learning algorithm [Ranzato et al., 2015] are only partially satisfying. One of the promising path to face this challenge is the Generative Adversarial Network framework (GAN). The GAN paradigm is a very general approach for doing simulation of data that lies in very complex spaces. In short, it consists in two networks trained (usually) simultaneously. A *Generator* which is trained to generate a variable lying in a given space and a *Decoder* which aims to distinguish if the generated data is real or generated. The overall system is trained in an end-to-end manner. GAN, as seen in [Wu et al., 2017], allows a more subtle objective for machine translation. Indeed, by letting the Discriminator to compare the predicted sequence and the gold prediction, we reach a richer evaluation procedure.

Following that direction and the wish of using syntactic information, we could enhance the Discriminator. A few paths seems open to do so. If we have a parse tree of the gold sentence as well as the prediction sentence, a solution would be to design a Discriminator that encodes both the surface form and the parsing tree. This would make the Discriminator syntax-aware.

3.4 Tackling Data Scarcity

One of the drawbacks of such an encoder-decoder architecture is that it requires tremendous amount of annotated data. To tackle this problem we describe five partially satisfying solutions that can help facing this data scarcity issue.

Unsupervised Representation Learning

We lack annotated data, but we have access to massive amount of raw textual data. Designing unsupervised methods that could help our downstream supervised models is one of the reasons why DL systems is so successful in NLP.

The concept of word embedding has become very popular. In short, it consists in learning a vectorial representation of words based on raw textual data. Such approaches usually model sentence level context and/or word morphology in some way to produce word vectors. The most popular one is the Skip-Gram model [Mikolov et al., 2013]. Briefly, a random vector is first associated with each word in a predefined vocabulary. Then, we train a simplified version of a shallow softmax regression - the so-called negative sampling objective - to predict context words given a focus word in the sentence. By training it on a lot of data, this model was shown to capture both syntactic and semantic properties of words and to provide improvements in downstream tasks.

More recently, the ELMo model [Peters et al., 2018] showed that using the hidden states of a character based language model could also generate very rich word embeddings. This contribution is striking in the way that a word is not associated to a fixed vector as it was done before, but it becomes a function of the nearby context. Moreover, thanks to its strict character-based nature, the ELMo does not require a predefine vocabulary. This approach is very promising and was shown to contribute to several progress in downstream tasks.

Recently, [Ficler and Goldberg, 2017] aimed to learn a style and content aware language model. They applied it to movie reviews. To do so , they simply conditioned their language model (a standard RNN language model) to a one-hot vector that encodes the writing style and the content, each described with a binary label, respectively [*Personal*,*Professional*] and [*Sentiment*,*Theme*]. The authors showed that this architecture was outperforming a standard unconditioned language model in terms of perplexity. The challenge of a such approach is to get the annotated data. This contribution is proposing a few simple ways using syntax and the presence of keywords to generate such annotations.

As mentioned above, the performances of encoder-decoder architectures might degrade drastically when we want to work with Out-of-Domain data. Following [Ficler and Goldberg, 2017], one of the path of research would be to adapt the condition language model paradigm to the ELMo model. This would produce language-level, content, and style aware representation learnt in a semi-supervised way. Seeing the extreme variability of User-Generated content, such robust representations might be well suited and help our multi-task architecture.

Using external lexicon

Another angle of research for tackling data scarcity, that we had the chance to develop at ALMANACH, is to leverage the large set amount of available external lexicons. Lexicons are collections of words associated with their possible morphological features and Part-of-Speech tags. As we show during our experiments, external lexicons significantly improves Part-of-Speech tagging and syntactic dependency parsing. Still, these attempts were done quite naively by encoding lexical information with straight n-hot encoded vectors or simple embedding vector. Following those attempts, learning dense representations, whether it is upstream or within our multi-task architecture, in a finer way should also help our model dealing with rare words and variability in the data.

Distant Supervision

The architecture we described in 3.3 raises lots of challenges from a data point of view. Indeed, training it would require at some point, cross-lingual parallel data for translation, normalized text aligned with its raw counter-part and parse trees, which is the most expensive to get.

Following [Eriguchi et al., 2017], we could attempt taking a *Distant Learning* approach. In short, instead of using annotated gold parsing trees, we could generate them automatically with some parser. We would then be able to work with parallel data for normalization or translation parsed in a noisy way.

Adversarial Training

Adversarial Training (AT), not to be mixed up with Generative Adversarial Network (GAN) [Goodfellow, 2016] is a regularization method that allows a model to be more robust to noise and adversarial examples. In short, an adversarial example, is a sample of data on which we perform a small perturbation. This perturbation is defined such that the resulting sample maximizes the loss (it's the worst possible perturbation near the given sample). Showing this perturbed sample to the model would lead to a dramatic false prediction. [Goodfellow et al., 2015] was the first to demonstrate, in the context of image processing, that designing an objective that includes adversarial examples could make the model more robust to noise and less sensitive to unknown adversarial examples.

Recently, [Yasunaga et al., 2017] showed that doing AT on the word and character embedding space could lead to more robust Part-Of-Speech Tagging (PoS). Still, we highlight here that applying AT to an image is quite different to applying it to an embedding space. Indeed, the point of [Goodfellow et al., 2015] was that perturbing an image slightly could lead to a wrong prediction while the resulting image was still the same for the human eye. In the context of language processing, modifying the samples at the embedding level doesn't lead to a same remark. Indeed, a modified word vectors is not really grounded to any meaningful structure.

Following this remark, we could think of designing a more linguistically grounded AT procedure for NLP. Indeed, instead of modifying the word vector only based on the "worst gradient direction", we could pick the word in its close neighborhood in the embedding space that leads to the worst perturbation. This would make AT for NLP more intelligible which could lead to better insights on why this procedure is actually successful.

4 Road-map

To summarize, this research aims to make use of the multi-task learning approach within the encoder-decoder paradigm for improving and making Machine Translation, Parsing and Normalization more robust. We now draw a brief road-map of how we conceive this research to be developed.

The first year of the PhD will be allocated to an extensive bibliographical work, the design and the training of the main multi-task architecture. The second year will be focused on experimenting with the models specifically on User Generating Content. We will aim at improving the performance and the robustness along several key axis, whether it is on learning better representation upstream, on the objective function itself or on the training procedure. The third year will aim at running the final experiments in regard to Out-of-Domain behaviors and writing the final thesis.

5 Conclusion

As we described, the recent successes of NLP systems remain inherently constrained to the context they have been trained on. To tackle this great challenge, we aim to leverage the existing multi-task learning approach using the very general encoder-decoder paradigm to improve and make our system generalize better. We will focus specifically on three tasks : Machine Translation, Text Normalization and Syntactic Parsing. We will specifically aim at improving the adaptability of our system to the variability of User Generated Content.

References

[Aharoni and Goldberg, 2017] Aharoni, R. and Goldberg, Y. (2017). Towards string-to-tree neural machine translation. *arXiv preprint arXiv:1704.04743*.

[Ba et al., 2016] Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

[Bojar et al., 2014] Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., and Specia, L., editors (2014). *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Baltimore, Maryland, USA.

[Bollmann and Søgaard, 2016] Bollmann, M. and Søgaard, A. (2016). Improving historical spelling normalization with bi-directional lstms and multi-task learning. *arXiv preprint arXiv:1610.07844*.

[Brunner et al., 2018] Brunner, G., Wang, Y., Wattenhofer, R., and Weigelt, M. (2018). Natural language multitasking: Analyzing and improving syntactic saliency of hidden representations. *arXiv preprint arXiv:1801.06024*.

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

[Chen et al., 2018] Chen, M. X., Firat, O., Bapna, A., Johnson, M., Macherey, W., Foster, G., Jones, L., Parmar, N., Schuster, M., Chen, Z., et al. (2018). The best of both worlds: Combining recent advances in neural machine translation. *arXiv preprint arXiv:1804.09849*.

[Cho et al., 2014a] Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014a). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

[Cho et al., 2014b] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014b). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

[Collobert et al., 2011] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

[Daniel Zeman, 2017] Daniel Zeman, Martin Popel, M. S. (2017). Proceedings of the conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. *CoNLL 2017 Shared Task*.

[Dong and Lapata, 2018] Dong, L. and Lapata, M. (2018). Coarse-to-fine decoding for neural semantic parsing. *arXiv preprint arXiv:1805.04793*.

[Dozat et al., 2017] Dozat, T., Qi, P., and Manning, C. D. (2017). Stanford's graph-based neural dependency parser at the conll 2017 shared task. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30.

[Dyer et al., 2016] Dyer, C., Kuncoro, A., Ballesteros, M., and Smith, N. A. (2016). Recurrent neural network grammars. *arXiv preprint arXiv:1602.07776*.

[Eriguchi et al., 2017] Eriguchi, A., Tsuruoka, Y., and Cho, K. (2017). Learning to parse and translate improves neural machine translation. *arXiv preprint arXiv:1702.03525*.

[Ficler and Goldberg, 2017] Ficler, J. and Goldberg, Y. (2017). Controlling linguistic style aspects in neural language generation. *arXiv preprint arXiv:1707.02633*.

[Goodfellow, 2016] Goodfellow (2016). In what way are adversarial networks related or different to adversarial training? https://www.quora.com/In-what-way-are-Adversarial-Networks-related-or-different-to-Adversarial-Training.

[Goodfellow et al., 2015] Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. *stat*, 1050:20.

[Hashimoto and Tsuruoka, 2017] Hashimoto, K. and Tsuruoka, Y. (2017). Neural machine translation with source-side latent graph parsing. *arXiv preprint arXiv:1702.02265*.

[Jurafsky, 2000] Jurafsky, D. (2000). *Speech & language processing*. Pearson Education India.

[Kitaev and Klein, 2018] Kitaev, N. and Klein, D. (2018). Constituency parsing with a self-attentive encoder. *arXiv preprint arXiv:1805.01052*.

[Levenshtein, 1966] Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

[Lin et al., 2017] Lin, Z., Feng, M., Santos, C. N. d., Yu, M., Xiang, B., Zhou, B., and Bengio, Y. (2017). A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

[Liu and Zhang, 2017] Liu, J. and Zhang, Y. (2017). Encoder-decoder shift-reduce syntactic parsing. *arXiv preprint arXiv:1706.07905*.

[Luong et al., 2015] Luong, M.-T., Le, Q. V., Sutskever, I., Vinyals, O., and Kaiser, L. (2015). Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

[Magué et al., 2016] Magué, J.-P., Karsaï, M., Seddah, D., and Chevrot, J.-P. (2016). A sociolinguistics of twitter.

[McCann et al., 2018] McCann, B., Keskar, N. S., Xiong, C., and Socher, R. (2018). The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

[Mikolov et al., 2013] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

[Nivre et al., 2017] Nivre, J., Agić, Ž., Ahrenberg, L., Antonsen, L., Aranzabe, M. J., Asahara, M., Ateyah, L., Attia, M., Atutxa, A., Badmaeva, E., Ballesteros, M., Banerjee, E., Bank, S., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bosco, C., Bouma, G., Bowman, S., Burchardt, A., Candito, M., Caron, G., Cebiroğlu Eryiğit, G., Celano, G. G. A., Cetin, S., Chalub, F., Choi, J., Cho, Y., Cinková, S., Çöltekin, Ç., Connor, M., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dobrovoljc, K., Dozat, T., Droganova, K., Eli, M., Elkahky, A., Erjavec, T., Farkas, R., Fernandez Alcalde, H., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D., Garcia, M., Ginter, F., Goenaga, I., Gojenola, K., Gökırmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Grūzītis, N., Guillaume, B., Habash, N., Hajič, J., Hajič jr., J., Hà Mỹ, L., Harris, K., Haug, D., Hladká, B., Hlaváčová, J., Hohle, P., Ion, R., Irimia, E., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kayadelen, T., Kettnerová, V., Kirchner, J., Kotsyba, N., Krek, S., Kwak, S., Laippala, V., Lambertino, L., Lando, T., Lê Hồng, P., Lenci, A., Lertpradit, S., Leung, H., Li, C. Y., Li, J., Ljubešić, N., Loginova, O., Lyashevskaya, O., Lynn, T., Macketanz, V., Makazhanov, A., Mandl, M., Manning, C., Manurung, R., Mărănduc, C., Mareček, D., Marheinecke, K., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Mendonça, G., Missilä, A., Mititelu, V., Miyao, Y., Montemagni, S., More, A., Moreno Romero, L., Mori, S., Moskalevskyi, B., Muischnek, K., Mustafina, N., Müürisep, K., Nainwani, P., Nedoluzhko, A., Nguyễn Thị, L., Nguyễn Thị Minh, H., Nikolaev, V., Nitisaroj, R., Nurmi, H., Ojala, S., Osenova, P., Øvrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Pitler, E., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Real, L., Reddy, S., Rehm, G., Rinaldi, L., Rituma, L., Rosa, R., Rovati, D., Saleh, S., Sanguinetti, M., Saulīte, B., Sawanakunanon, Y., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shakurova, L., Shen, M., Shimada, A., Shohibussirri, M., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Stella, A., Strnadová, J., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Trosterud, T., Trukhina, A., Tsarfaty, R., Tyers, F., Uematsu, S., Urešová, Z., Uria, L., Uszkoreit, H., van Noord, G., Varga, V., Vincze, V., Washington, J. N., Yu, Z., Žabokrtský, Z., Zeman, D., and Zhu, H. (2017). Universal dependencies 2.0 – CoNLL 2017 shared task development and test data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

[Ott et al., 2018] Ott, M., Auli, M., Granger, D., and Ranzato, M. (2018). Analyzing uncertainty in neural machine translation. *arXiv preprint arXiv:1803.00047*.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

[Peters et al., 2018] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

[Ranzato et al., 2015] Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2015). Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

[Seddah et al., 2012] Seddah, D., Sagot, B., Candito, M., Mouilleron, V., and Combet, V. (2012). The french social media bank: a treebank of noisy user generated content. In *COLING 2012-24th International Conference on Computational Linguistics*.

[Sridhar, 2015] Sridhar, V. K. R. (2015). Unsupervised text normalization using distributed representations of words and phrases. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 8–16.

[Subramanian et al., 2018] Subramanian, S., Trischler, A., Bengio, Y., and Pal, C. J. (2018). Learning general purpose distributed sentence representations via large scale multi-task learning. *arXiv preprint arXiv:1804.00079*.

[Szegedy et al., 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

[Tao Ge, 2018] Tao Ge, Furu Wei, M. Z. (2018). Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the ACL 2018*.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

[Wu et al., 2017] Wu, L., Xia, Y., Zhao, L., Tian, F., Qin, T., Lai, J., and Liu, T.-Y. (2017). Adversarial neural machine translation. *arXiv preprint arXiv:1704.06933*.

[Wu et al., 2016] Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

[Yasunaga et al., 2017] Yasunaga, M., Kasai, J., and Radev, D. (2017). Robust multilingual part-of-speech tagging via adversarial training. *arXiv preprint arXiv:1711.04903*.