A Syntax-Aware Edit-based System for Text Simplification

Oscar M. Cumbicus-Pineda,
Ixa group and Carrera de Computación
UPV/EHU and UNL
ocumbicus001@ikasle.ehu.es

Itziar Gonzalez-Dios, Aitor Soroa
Ixa group, HiTZ center
University of the Basque Country (UPV/EHU)
{itziar.gonzalezd,a.soroa}@ehu.eus

Abstract

Edit-based text simplification systems have attained much attention in recent years due to their ability to produce simplification solutions that are interpretable, as well as requiring less training examples compared to traditional seq2seq systems. Edit-based systems learn edit operations at a word level, but it is well known that many of the operations performed when simplifying text are of a syntactic nature. In this paper we propose to add syntactic information into a well known edit-based system. We extend the system with a graph convolutional network module that mimics the dependency structure of the sentence, thus giving the model an explicit representation of syntax. We perform a series of experiments in English, Spanish and Italian, and report improvements of the state of the art in four out of five datasets. Further analysis shows that syntactic information is always beneficial, and suggest that syntax is more helpful in complex sentences.

1 Introduction

Automatic Text Simplification (ATS) aims to reduce the linguistic complexity of a text for a certain target audience. ATS is useful for people learning foreign languages, suffering cognitive disabilities, or with low literacy level. The area of application of ATS is however not restricted to humans, as it has also been used to improve automatic NLP tasks such as parsing (Chandrasekar et al., 1996a), summarization (Beigman Klebanov et al., 2004; Silveira and Branco, 2012), semantic role labeling (Vickrey and Koller, 2008; Woodsend and Lapata, 2017), information extraction (Evans and Orasan, 2019) and machine translation (Gerber and Hovy, 1998; Štajner and Popovic, 2016; Hasler et al., 2017), among others.

ATS is often cast as a machine translation task, where the system receives complex sentences as input, and produces sentences that are simpler yet maintaining the original meaning. While early ATS systems were rule-based, statistical or hybrid (Saggion, 2017), in the last years neural network based ATS approaches have also been proposed (Alva-Manchego et al., 2020). In particular, sequence-to-sequence (seq2seq) neural models (Sutskever et al., 2014; Nisioi et al., 2017; Zhang and Lapata, 2017a) have shown to obtain state-of-the-art results. Such systems are trained on parallel corpora comprising pairs of complex/simple sentences, and implicitly learn the simplification rewrites needed to convert complex sentences into simpler ones.

Neural seq2seq systems are usually black boxes that are trained on an end-to-end fashion. As a consequence, built models are usually very difficult to interpret, and offer little control or hints that explain why a particular input word sequence has been rephrased. Edit-based ATS systems try to overcome this limitation by learning the transformations required to convert complex sentences into their simpler counterparts. The set of transformations is limited and known beforehand, and usually comprise edit operations such as delete, removal or lexical substitution (Alva-Manchego et al., 2017; Dong et al., 2019; Kumar et al., 2020). Because the set of allowed operations is restricted, the search space is considerably reduced. As a consequence, edit-based models are usually sample efficient and require less training examples compared to traditional seq2seq systems (Mallinson et al., 2020; Omelianchuk et al., 2021).

Edit-based ATS systems learn edit transformations at a word level, but often those operations are applied to whole phrases. Besides, systems need to capture long range relations among words, such as syntactic and phrase structures. For example, in the English sentence presented in Table 1 there is a long subject "Dry air wrapping around the southern periphery of the cyclone" that causes the

Proceedings of Recent Advances in Natural Language Processing, pages 329–339
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_039

Version	Sentence
Orig. (EN)	Dry air wrapping around the southern periphery of the cyclone eroded most of the deep convection by early on September 12.
Aut.	Dry air wrapping around the southern periphery of the cyclone.
Orig. (ES)	*En este sentido, el presidente de la Oficina Nacional de la Caza, Juan Antonio Sarasketa, aseguró que este proyecto de reglamento imposibilita la posesión y uso de armas, (...)* In this sense, the president of the National Hunting Office, Juan Antonio Sarasketa, assured that this regulation project makes it impossible to possess and use weapons, (...)
Aut.	*el presidente de la oficina nacional de la caza , juan antonio sarasketa , aseguró que el gobierno de cine de emigrantes* The president of the National Hunting Office, Juan Antonio Sarasketa, assured that the goverment of immnigrants cinema

Table 1: Sentences with long sentences (English) or subordination clauses (Spanish) cause ATS to fail.

ATS to produce a simplification that is ungrammatical. The Spanish sentence in the Table 1 contains completive clause introduced by *que*. However, completive clauses require a verb which has not been added in the simplified sentence.

The examples above show that word-based edit systems often fail to capture syntactic relations regarding the syntactic functions, phrase-level syntax and subordination. Following this intuition, in this paper we propose leveraging syntactic information derived from dependency trees into a well known edit-based ATS system. We present a syntax aware edit-based system for ATS which uses a graph convolutional network (GCN) layer (Kipf and Welling, 2017) to represent the dependency trees. In the training process, the GCN learns to refine the representation of input sentence words according to their structural relations in the dependency graph. These syntax augmented representations are combined with the encoder outputs using a residual connection, and passed to the decoding stage.

Our experiments confirm the effectiveness of our approach, outperforming previous ATS systems and improving the state-of-the-art results in several datasets.

The contributions of this paper are the following:

- We present a method to integrate syntactic information into an edit-based seq2seq system.

- The results confirm our intuition that syntactic information is useful for ATS systems. The new system surpasses previous state of the art in four out of five datasets. We conduct an ablation study that shows the effect of removing the syntactic information from the system.

- We obtain the best results reported so far for Spanish and Italian.

This paper is structured as follows: in Section 2 we present the work done with dependencies, graphs and recent ATS systems; in Section 3 we describe our approach, in Section 4 our experiments and in Section 5 the results, we conclude and outline the future work in Section 7.

2 Related Work

Sentence simplification (SS) aims to reduce a sentence's reading complexity while preserving its meaning. Recently, systems based on neural networks are gaining attention in ATS. For more information about previous works in ATS the interested reader is referred to these works (Shardlow, 2014; Siddharthan, 2014; Saggion, 2017; Alva-Manchego et al., 2020; Sikka et al., 2020). In this section, we focus on work that are based on dependencies and on one of the recently developed techniques, which is based on editing to simplify the text.

Dependency based simplification has proven to be effective in rule based and statistical simplification to analyze the texts and induct rules in the rule-based and hybrid architectures of ATS systems (indeed, the first module was a parser and many works have relied on dependency parsers) (Chandrasekar et al., 1996b; Woodsend and Lapata, 2011; Siddharthan, 2011; Klerke and Søgaard, 2013; Aranzabe et al., 2013; Saggion et al., 2015) as well as to perform tree transformations together with statistical machine translation (Zhu et al., 2010).

However, in the last years neural approaches have gained interest in ATS and SS research. The most popular framework has been the seq2seq models, which mainly rely on RNN and transformer based architectures (Nisioi et al., 2017; Zhang and Lapata, 2017b). While the majority of ATS systems are supervised, some works have obtained good results using unsupervised approaches (Surya et al., 2019; Martin et al., 2020b).

One of the main drawback of the aforementioned approaches is the lack of interpretability, i.e., the extend to which a system can explain in a meaningful way why certain actions have been performed. Edit-based systems (Alva-Manchego et al., 2017; Dong et al., 2019; Kumar et al., 2020) offer a solution to the interpretability problem, as they directly learn a sequence of edit operations that convert complex sentences into simpler ones. Alva-Manchego et al. (2017) present a machine translation system that predicts three explicit simplification operations (keep, replace and delete) in aligned pairs of complex-simple sentences. Dong et al. (2019) propose EditNTS, a neural programmer/interpreter that learns to generate edit operations (add, keep, and delete) in a sequential fashion. Kumar et al. (2020) design a scoring function that measures the quality of a candidate sentence based on the fluency, simplicity, and meaning preservation and generate the simplified candidate sentences by iteratively editing the given complex sentence. The operations they take into account are removal, extraction, reordering and substitution.

Syntactic information has been previously used in rule-based systems, for instance, as an indicator to identify the complexity of sentences (Evans and Orasan, 2018). On deep learning systems, graph convolutional networks (GCN) over dependency trees is a usual method that leverages syntactic information into the models, and captures long-range syntactic relations among words. GCNs generalize the convolution operation usually applied in images to arbitrary graphs, and allow to refine information associated to nodes according to the information in the neighbor nodes (Kipf and Welling, 2017). They have been successfully used in NLP tasks such as semantic role labelling (Marcheggiani and Titov, 2017), information extraction (Zhang et al., 2018) and aspect-based sentiment analysis (Wang et al., 2020). Contemporaneous to this work, Zhe Lin (2021) use semantic information in seq2seq systems by including in the graph of the source sentence the predicate-argument relations between content words in a sentence.

3 A Syntax-Aware ATS System

Our system is based on EditNTS, an edit-based system that has obtained state of the art results on many datasets. We start by briefly describing EditNTS. Then, we describe the graph convolutional network that leverages syntactic information derived from dependency trees. Finally, we describe how to integrate the syntactic module into EditNTS.

3.1 EditNTS

We start by briefly describing the EditNTS system, and refer the reader to Dong et al. (2019) for a more detailed description. Let $x = x_1, \ldots, x_{|x|}$ be a complex input sentence and $y = y_1, \ldots, y_{|y|}$ its simplified version. EditNTS learns to produce a series of edit operations $z = z_1, \ldots, z_N$ which, applied over the input sentence x, produces y. Each edit operation z_i is one of $\{\mathrm{ADD}(w), \mathrm{KEEP}, \mathrm{DELETE}\}$. EditNTS contains an encoder, decoder and interpreter modules, which are described as follows:

Encoder

The encoder transforms the input sequence x into a sequence of output and hidden representations ($\mathbf{o}_i$ and $\mathbf{h}_i$):

$$\mathbf{o}_i, \mathbf{h}_i = \mathrm{LSTM}(\mathbf{x}_{1:i-1}) \quad (1)$$

where $\mathbf{x}_i$ is the concatenation of the embedding of the word x_i and the embedding corresponding to the POS tag of x_i.

Decoder

The decoder receives the input from the encoder, and predicts the next edit label z_t for each timestep t. Internally, it contains two recurrent networks that represent the edit operations and the output words produced so far:

$$\begin{aligned} \mathbf{o}_t^{\text{edit}} &= \mathrm{LSTM}(\mathbf{z}_{1:t-1} \mid \mathbf{h}_{|x|}) \\ \mathbf{o}_t^{y} &= \mathrm{LSTM}(\mathbf{y}_{1:t-1} \mid \mathbf{h}_{|x|}) \end{aligned}$$

where $\mathbf{z}$ are the embeddings of the edit operations and $\mathbf{y}$ are the output embeddings[1]. The decoder also uses an attention mechanism between the current edit operation and the input words. Let O and O^{edit} be the matrices whose rows are the output vectors for the encoder and edit recurrent networks, respectively[2]. The attention mechanism is defined as follows:

$$\begin{aligned} E &= \mathrm{softmax}(K O^{\text{edit}} O^T) \\ \mathbf{c}_t &= E_{t:} O \end{aligned}$$

where K is a parameter learned by the model. The decoder predicts the next edit label z_t using a sequence of linear layers and activation functions. The input of the linear layer $\mathbf{i}_t$ is a concatenation

[1] The embedding matrix is shared between the encoder and decoder.

[2] That is, $O_{i:} = \mathbf{o}_i$ and $O^{\text{edit}}_{i:} = \mathbf{o}^{\text{edit}}_i$. Through the paper we use the notation $M_{i:}$ to represent the ith row of matrix M.

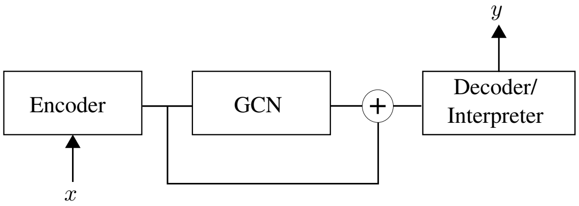

Figure 1: Architecture of the syntactic augmented EditNTS model.

of the output representation of the input word currently edited $\mathbf{o}_{k_t}$, the output representation of the previously generated edit labels $\mathbf{o}_t^{\text{edit}}$, the representation of the previous generated words $\mathbf{o}_{j_t}^{y}$ and the attention vector $\mathbf{c}_t$:

$$\begin{aligned} \mathbf{i}_t &= [\mathbf{o}_{k_t}; \mathbf{o}_t^{\text{edit}}; \mathbf{o}_{j_t}^{y}; \mathbf{c}_t] \\ z_t &= \mathrm{softmax}(V'(\tanh(V\mathbf{i}_t))) \end{aligned}$$

again, V and V' are parameters learned by the model.

Interpreter

The interpreter applies the predicted edit operation z_t on current word x_{k_t} and produces a new word y_{j_t}.

3.2 Graph Convolutional Network Module

To leverage syntactic information into EditNTS, we first compute the dependency tree of the complex sentence x, which is represented as an undirected graph[3]. The nodes of the graph are the words in x, and the edges represent syntactic relations among them. The graph is represented as an adjacency matrix A, where $A_{i,j} = 1$ if an edge between nodes i and j exist.

We then apply a series of graph convolutional operations over the syntactic graph using a Graph Convolutional Network module. The module contains a series of L linear layers, each one applying a convolution operation over the graph:

$$h^{l+1} = \mathrm{ReLU}(\tilde{D}^{\frac{1}{2}}\tilde{A}\tilde{D}^{\frac{1}{2}}h^{l}W^{(l)})$$

where ReLU is the linear rectifier activation function, $\tilde{A} = A + I_N$ is the adjacency graph with added self connections, $\tilde{D}_{ii} = \sum_{ij}\tilde{A}_{ij}$ is the degree matrix and $W^{(l)}$ is a layer specific parameter to be learned. h^0 is the input of the graph, a matrix that assigns an embedding to each vertex in the graph. At each level, the convolution operation aggregates the embeddings of neighbor nodes to produce new embeddings that implicitly encode the structure of the underlying graph.

Dataset	Lang	Train	Dev	Test
Wikilarge	EN	$296,402$	$2,000$	359
Newsela	EN	$94,208$	$1,129$	$1,076$
Simplext	ES	574	143	693
Newsela-es	ES	$50,301$	$2,794$	$2,795$
Italian	IT	$29,260$	$1,475$	$1,475$

Table 2: Sizes (number of sentences) of the datasets used in the experiments

Dataset		Sent. Length		Sent. Depth	
		Avg.	Std	Avg.	Std
Wikilarge	complex	26.78	13.13	6.47	2.60
	simple	19.96	11.81	5.39	2.35
Newsela	complex	26.79	10.81	7.07	2.49
	simple	16.79	6.01	5.59	1.76
Simplext	complex	40.25	17.45	7.24	2.24
	simple	15.94	4.13	4.74	1.01
Newsela-es	complex	30.78	13.99	6.40	1.94
	simple	20.33	9.01	5.29	1.54
Italian	complex	11.37	6.51	3.66	1.22
	simple	10.70	6.56	3.63	1.21

Table 3: Corpora statistics (train). Sentence length measures the number of tokens and the depth shows the depth of the dependency tree.

3.3 Augmenting EditNTS with syntactic information

In our final system the GCN initial input h^0 is initialized with the encoder outputs $\mathbf{o}_i$ of each word in the sentence, and the corresponding adjacency graph derived from the dependency tree. The output h^L of the GCN module is then combined with the original $\mathbf{o}_i$ vectors as a residual connection, which is then passed to the EditNTS decoder (see Figure 1).

4 Experiments

In this section we describe the experiments performed within this work. We start by describing the datasets used for training and testing the system, followed by the experimental setting, which includes a description of the metrics used to evaluate the models.

4.1 Datasets and parameters

We experiment our approach on ATS datasets from three languages: English, Spanish and Italian. The datasets used for each language are the following:

[3] Preliminary experiments showed no gain by representing the dependency tree as a directed graph.

English We used two datasets for English, Wikilarge/TurkCorpus and Newsela. **WikiLarge** is one of the most used to train ATS systems and was built by automatically aligning sentences belonging to the same article in English Wikipedia and Simple English Wikipedia. We used the split provided by Zhang and Lapata (2017a) for training and development, with a total of 296,402 and 2,000 sentences, respectively. Following usual practice, we tested the models trained on Wikilarge using the TurkCorpus dataset (Xu et al., 2016), which contains eight manually generated reference simplifications for 359 sentences. **Newsela** consists of 1130 news articles that were rewritten four times at different complexity levels. We used the train/development/test splits from (Xu et al., 2015), containing 94,208/1129/1076 sentences respectively.

Spanish We used two datasets for Spanish, **Simplext** and **Newsela-es**. The **Simplext** corpus contains 200 news texts from different domains, that were manually simplified (Saggion et al., 2015). We use the splits provided by Martin et al. (2020b) with 574/143/693 sentences for training, development and test. **Newsela-es** is similar to its English counterpart, we used the splits from Palmero Aprosio et al. (2019) and comprises 50,301/2,794/2,795 sentences for train/dev/test.

Italian For Italian we use the documents provided by Palmero Aprosio et al. (2019), a corpus containing $32,210$ complex-to-simple pairs sentences that were obtained by merging three available data sets: the SIMPITIKI corpus (Tonelli et al., 2016), the corpora *Terence* and *Teacher* (Brunato et al., 2015), and a subset of the PaCCSS-it corpus (Brunato et al., 2016).

Table 2 shows the size of each dataset, and in Table 3 we present the average and standard deviation of the sentence length and depth for each corpus, for both the complex and simple sentences. The sentence length measures the number of tokens of each sentence and the depth shows the maximum depth of the dependency trees.

4.2 Experimental settings

We tokenized and syntactically analyzed the documents with spacy[4], using the large models for each particular language. All words were lowercased. In the case of Newsela-EN, we follow (Xu et al., 2015) and replace all named entities with a placeholder that represents the entity type.

Wikilarge/TurkCorpus	**SARI**	**BLEU**
Identity	25.98	97.34
Dress-LS (Zhang and Lapata, 2017a)	37.27	**80.12**
DMASS-DCSS (Zhao et al., 2018)	40.45	
ACCESS (Martin et al., 2020a)	**41.87**	–
EditNTS (Dong et al., 2019)	38.22	
EditNTS†	36.75	72.99
SDISS (Zhe Lin, 2021)	38.66	77.36
Edit+synt	36.97	75.35

(a) Wikilarge/TurkCorpus

Newsela-en	**SARI**	**BLEU**
Identity	13.17	23.68
Dress-LS (Zhang and Lapata, 2017a)	26.63	**24.30**
DMASS-DCSS (Zhao et al., 2018)	27.28	
EditNTS (Dong et al., 2019)	31.41	
EditNTS†	33.62	22.43
SDISS (Zhe Lin, 2021)	32.30	18.81
Edit+synt	**38.08**	20.58

(b) Newsela-en

Newsela-es	**SARI**	**BLEU**
Identity	17.05	**33.95**
NTS (Palmero Aprosio et al., 2019)	30.08	
Edit+synt	**35.03**	20.84

(c) Newsela-es

Simplext	**SARI**	**BLEU**
Identity	5.65	7.39
MBART+ACCESS*(Martin et al., 2020b)	28.56	6.90
Edit+synt	**39.48**	**7.51**

(d) Simplext

Italian	**SARI**	**BLEU**
Identity	26.01	**63.84**
NTS (Palmero Aprosio et al., 2019)	50.00	
Edit+synt	**52.25**	53.54

(e) PaCCSS-it,SIMPITIKI,Terence-Teacher

Table 4: Comparison with state of the art systems. The figures are obtained from the respective papers except those marked with †, which means that we ran and evaluated the systems ourselves. $*$ means that the system is unsupervised.

During training, a teacher forcing strategy is followed half of the times. When teacher forcing is followed, the decoder is provided with the gold edit labels and target token; when not, the decoder at each time step is fed with the output produced in the previous edit label and target token. Default hyperparameters from EditNTS are used, and no hyperparameter tuning was performed: a batch size of 64, a hidden dimension of 200 and a learning rate of 10^{-3}. We used Adam optimizer and a decay factor of 10^{-6}. The models are trained during 50 epochs, and the model that obtained the best SARI score in the corresponding development split is selected and tested.

[4] https://spacy.io/

Version	Sentence
Orig. (EN)	He died on May 29, 1518 in Madrid, Spain and was buried in the church of San Benito d'Alcantara.
Ref.	He died on May 29, 1518 in Madrid, Spain. It was buried in the church of San Benito d'Alcantara.
Edit+synt	he died on may 29, 1518 in madrid, spain. he was buried in the church of san benito d'alcantara.
Edit	he died on may 29, 1518 in madrid, spain. buried in the church of san benito d'alcantara.
Orig. (ES)	*Los especialistas advierten que el asma se agrava en otoño.* Specialists warn that asthma worsens in autumn.
Ref.	*Los especialistas advierten que el asma es peor en otoño.* Specialists warn that asthma is worse in autumn.
Edit+synt	*los especialistas dicen de que el asma se reduce en otoño.* Specialists warn that asthma is reduced in autumn.
Orig. (IT)	*Ma la cosa più interessante non è questa.* But this is not the most interesting thing.
Ref.	*Ma non è questa la cosa più grave.* But this is not the most difficult thing.
Edit+synt	*ma la cosa non può essere questa.* But the thing cannot be this.
Edit	*ma la cosa più importante non è.* But the most important thing is not.

Table 5: Example of a sentence from Turkcorpus, Newsela-ES and the Italian corpus.

Regarding evaluation, the following evaluation metrics are used:

- **SARI** is a common evaluation metric for ATS systems (Xu et al., 2016) that measures the number of ngrams that have been added/removed/kept by the simplification system.

- **BLUE**. Following usual practice, we also include the BLUE score (Papineni et al., 2002) between the complex and simple sentences. Although BLUE has been criticized as a measure to evaluate simplification systems (Sulem et al., 2018; Dong et al., 2019; Martin et al., 2020b), we use it here for completeness.

We compute the evaluation metrics using the EASSE package (Alva-Manchego et al., 2019). We do not report the readability score Flesch–Kincaid Grade Level (FKLG) (Kincaid et al., 1975) because it is a language dependent metric which is only valid for English.

4.3 Baselines

We use the Identity baseline that simply copies the complex sentence. Apart from this, we compare our syntax-aware system against the state-of-the-art on each language. For English, we consider the deep reinforcement based neural system Dress-LS (Zhang and Lapata, 2017a), the transformer based model DMASS-DCSS (Zhao et al., 2018) and BART+ACCESS (Martin et al., 2020b), which also includes special tags to perform controllable text generation. There are fewer systems to compare against in the Spanish and Italian datasets, as, like many other areas in NLP, ATS systems have been developed mostly for English. For Newsela-es and the Italian dataset, we compare ourseves Neural TS (Palmero Aprosio et al., 2019), an MT system based on an attention encoder-decoder model. Finally, for Simplext we include the unsupervised system in Martin et al. (2020b). In the datasets for English we also report the results of EditNTS (Dong et al., 2019) with no syntax.

5 Results

Table 4 shows the results of our syntax aware system (dubbed Edit+synt in the tables), and compares them with the best performing systems on the different datasets. We see that the syntax aware system obtains very good results overall, and improves state of the art SARI results in four out of five datasets. This is a remarkable result that stresses the importance of syntax in text simplification. The table shows that datasets with smaller training data are most benefited from our approach, and suggests that the combination of edit operations and syntactic information is able to generalize in low training data regimes.

It is worth noting that the results obtained by us when running EditNTS without syntax are different to those reported in (Dong et al., 2019), and that the gap is specially large in the Wikilarge/Turkcorpus dataset[5]. We attribute this difference to the fact that the reported results in the original paper are obtained using the model that performed best in the test split, whereas we performed model selection according to the development dataset (c.f Section 4.2). There is also a slight difference in the Newsela dataset, which we attribute to the use of different evaluation scripts.

[5] The latter results are marked with † in the tables.

	No syntax		Syntax		Δ_{SARI}	Δ_{BLEU}
	SARI	**BLEU**	**SARI**	**BLEU**		
Wikilarge/TurkCorpus	36.75	72.99	36.97	75.35	0.22	2.36
Newsela	33.62	22.43	38.08	20.58	4.46	-1.85
Simplext	36.52	7.31	39.48	7.51	2.96	0.2
Newsela-es	34.50	22.78	35.03	20.84	0.53	-1.94
Italian	51.95	53.43	52.25	53.54	0.3	0.11

Table 6: Main results. *No syntax* stands for the original system, whereas *Syntax* uses syntactic information. The Δ measures the difference between both systems.

6 Analysis

In this section we perform an analysis of the results. We start by analyzing examples of the outputs, followed by an ablation study. Finally, we analyze the effect of the sentence complexity in the simplification process.

6.1 Output analysis

In Table 5 we present three sentences, one for each language that we explain next. In the English sentence (from Wikilarge), we see that both systems have been able to carry out a sentence splitting, but in the second sentence of the edit system the subject (he) is missing, which makes the ungrammatical (note that in the reference sentence, the pronoun is not correct). In the Italian sentence (from PaCCSS-it,SIMPITIKI,Terence-Teacher), the edit system has deleted the subject and it is ungrammatical (it could be grammatical with a different word ordering). The sentence created by Edit+synt has changed the modality of the verb (*è* 'it is'-> *può essere* 'it can be') and deleted the attribute *più interessante* 'more interesting', and this affects the meaning of the sentence. All in all, our analysis suggests that there is still a large room for improvement in non-English simplification. For example, the Spanish example in Table 5 is not correct at grammatical level: there is a *dequeísmo*, which is the misuse of the preposition *de* in front of the conjunction *que* when the preposition is not required like in this case by the verb *decir*. This is not grammatically correct but it can be understood without problems. Moreover, the *dequeísmo* is a common mistake by many speakers, and it would be interesting to check the corpus to find out if there are non standard grammatical variations or misuses. However, the main problem of the sentence is related to lexical simplification. The verb *se agrava* (it worsens) has been replaced with *se reduce* (it reduces). This is a wrong simplification, but if the sentence was not comprehensible at grammar level, the meaning preservation and simplicity cannot be correctly evaluated.

6.2 Ablation study

Table 6 shows the results of the system using syntactic dependencies or not. We see that, in general, syntactic dependencies are helpful and lead to an improvement in SARI on all datasets. The gain in SARI is particularly large in the Newsela and Simplext datasets, which are the datasets with highest average sentence depths. These results suggest that syntactic information is particularly helpful when simplifying complex sentences. This trend does not hold if we compare the SARI gain with the average sentence length. We analyze this correlation further in the next section. Regarding BLEU, the table shows mixed results, with gains in all datasets except in Newsela and Newsela-es

6.3 The effect of the sentence complexity

The results in the ablation study indicate a correlation between the sentence complexity and the performance gain obtained when using syntactic information, and now we analyze this correlation further. Figure 2 shows an analysis of length and depth per sentence that helps understanding this relationship. The x axes in the figures correspond to the sentence depth (left) and length (right), and the y axes show the average SARI gain, that is, the average of the differences between the *Syntax* and *No syntax* scores for all sentences with one particular depth or length[6]. While the results vary among datasets, the left graph shows a general tendency where the gain of using syntax is greater on the deepest sentences. That is, sentences that have complex dependency trees are better simplified, according to SARI, when using syntactic information.

[6] The graphs have been smoothed using the Exponential Moving Average (EMA) technique with a smoothness factor of 0.75 to flatten the peaks. While the smoothing process removes information from the graph, the loss is outweighed by the improved visibility.

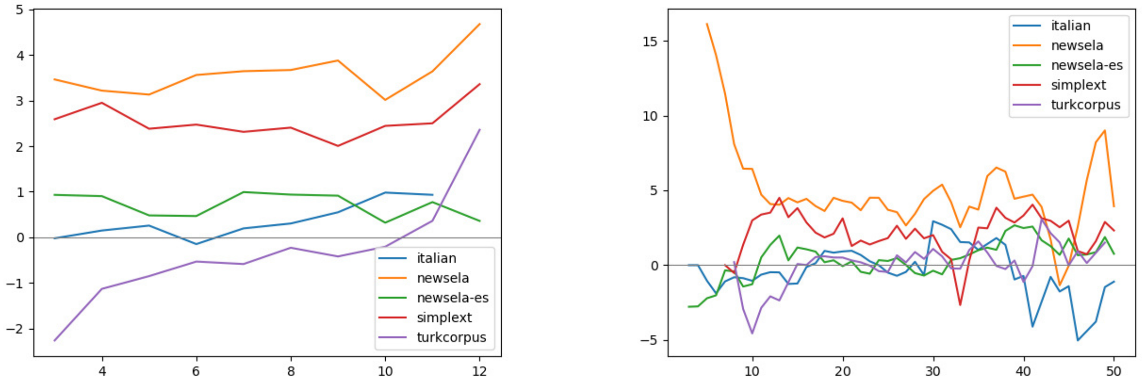

Figure 2: Average SARI gain for different sentence depths (left) and length (right). Best viewed in color.

This gain is particularly high on the Newsela, Wikilarge/TurkCorpus and Simplext datasets.

The right graph in Figure 2 shows no overall correlation with the sentence length and the SARI gain. This result suggests that the length of a sentence per-se is not a valid indicator of the syntactic complexity of the sentences.

7 Conclusion and Future Work

In this paper we show that syntactic information is a valid source of information for edit-based text simplification systems. We have presented a system that extends a well known edit-based system with explicit syntactic information derived from dependency trees, by virtue of graph convolutional networks. The results show that the dependency information is useful, obtaining state of the art results on four out of five datasets in different languages. Further analysis show that the syntactic information is always beneficial (sometimes by a large margin), and that the improvement is often correlated with the depth of the dependency tree.

In the future we want to analyze the inclusion of dependency syntax information into transformer based seq2seq systems. In particular, we want to analyze whether explicitly modeling syntactic information is still a valid approach when the transformer based ATS system is initialized with large language models such as BART (Lewis et al., 2020).

Acknowledgments

This work was supported by the Basque Government (excellence research group IT1343-19 and DeepText project KK-2020/00088), project BigKnowledge (*Ayudas Fundación BBVA a equipos de investigación científica 2018*) and the Spanish MINECO (project DeepReading RTI2018-096846-B-C21).

References

Fernando Alva-Manchego, Joachim Bingel, Gustavo Paetzold, Carolina Scarton, and Lucia Specia. 2017. Learning how to simplify from explicit labeling of complex-simplified text pairs. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 295–305, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. 2019. Easse: Easier automatic sentence simplification evaluation. In *EMNLP-IJCNLP 2019-Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (demo session)*, pages 49–54.

Fernando Alva-Manchego, Carolina Scarton, and Lucia Specia. 2020. Data-driven sentence simplification: Survey and benchmark. *Computational Linguistics*, 46(1):135–187.

María Jesús Aranzabe, Arantza Díaz de Ilarraza, and Itziar Gonzalez-Dios. 2013. Transforming complex sentences using dependency trees for automatic text simplification in basque. *Procesamiento del lenguaje natural*, 50:61–68.

Beata Beigman Klebanov, Kevin Knight, and Daniel Marcu. 2004. Text simplification for information-seeking applications. In *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*, pages 735–747, Berlin, Heidelberg. Springer Berlin Heidelberg.

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, and Giulia Venturi. 2016. Paccss-it: A parallel corpus of complex-simple sentences for automatic text simplification. In *Proceedings* *of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 351–361.

Dominique Brunato, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2015. Design and annotation of the first Italian corpus for text simplification. In *Proceedings of The 9th Linguistic Annotation Workshop*, pages 31–41, Denver, Colorado, USA. Association for Computational Linguistics.

R. Chandrasekar, Christine Doran, and B. Srinivas. 1996a. Motivations and methods for text simplification. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Raman Chandrasekar, Christine Doran, and Srinivas Bangalore. 1996b. Motivations and methods for text simplification. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Yue Dong, Zichao Li, Mehdi Rezagholizadeh, and Jackie Chi Kit Cheung. 2019. EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3393–3402, Florence, Italy. Association for Computational Linguistics.

Richard Evans and Constantin Orasan. 2018. Identifying signs of syntactic complexity for rule-based sentence simplification. *Natural Language Engineering*.

Richard Evans and Constantin Orasan. 2019. Sentence simplification for semantic role labelling and information extraction. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 285–294, Varna, Bulgaria. INCOMA Ltd.

Laurie Gerber and Eduard Hovy. 1998. Improving translation quality by manipulating sentence length. In *Machine Translation and the Information Soup*, pages 448–460, Berlin, Heidelberg. Springer Berlin Heidelberg.

Eva Hasler, Adrià de Gispert, Felix Stahlberg, Aurelien Waite, and Bill Byrne. 2017. Source sentence simplification for statistical machine translation. *Computer Speech & Language*, 45:221–235.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Technical Training Command Millington TN Research Branch.

Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the 5th International Conference on Learning Representations*, ICLR '17.

Sigrid Klerke and Anders Søgaard. 2013. Simple, readable sub-sentences. In *51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop*, pages 142–149.

Dhruv Kumar, Lili Mou, Lukasz Golab, and Olga Vechtomova. 2020. Iterative edit-based unsupervised sentence simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7918–7928, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Jonathan Mallinson, Aliaksei Severyn, Eric Malmi, and Guillermo Garrido. 2020. FELIX: Flexible text editing through tagging and insertion. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1244–1255, Online. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1506–1515, Copenhagen, Denmark. Association for Computational Linguistics.

Louis Martin, Éric de la Clergerie, Benoît Sagot, and Antoine Bordes. 2020a. Controllable sentence simplification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4689–4698, Marseille, France. European Language Resources Association.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2020b. Multilingual unsupervised sentence simplification. *arXiv preprint arXiv:2005.00352*.

Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P Dinu. 2017. Exploring neural text simplification models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 85–91.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text simplification by tagging.

Alessio Palmero Aprosio, Sara Tonelli, Marco Turchi, Matteo Negri, and Mattia A. Di Gangi. 2019. Neural text simplification in low-resource conditions using weak supervision. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 37–44, Minneapolis, Minnesota. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Horacio Saggion. 2017. Automatic Text Simplification. *Synthesis Lectures on Human Language Technologies*, 10(1):1–137.

Horacio Saggion, Sanja Štajner, Stefan Bott, Simon Mille, Luz Rello, and Biljana Drndarevic. 2015. Making it simplext: Implementation and evaluation of a text simplification system for spanish. *ACM Transactions on Accessible Computing (TACCESS)*, 6(4):1–36.

Matthew Shardlow. 2014. A Survey of Automated Text Simplification. *International Journal of Advanced Computer Science and Applications*, 4(1):58–70.

Advaith Siddharthan. 2011. Text simplification using typed dependencies: A comparision of the robustness of different generation strategies. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 2–11, Nancy, France. Association for Computational Linguistics.

Advaith Siddharthan. 2014. A Survey of Research on Text Simplification. *ITL-International Journal of Applied Linguistics*, 165(2):259–298.

Punardeep Sikka, Manmeet Singh, Allen Pink, and Vijay Mago. 2020. A Survey on Text Simplification. *arXiv preprint arXiv:2008.08612*.

Sara Botelho Silveira and António Branco. 2012. Enhancing multi-document summaries with sentence simplification. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer ....

Sanja Štajner and Maja Popovic. 2016. Can text simplification help machine translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018. Bleu is not suitable for the evaluation of text simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 738–744.

Sai Surya, Abhijit Mishra, Anirban Laha, Parag Jain, and Karthik Sankaranarayanan. 2019. Unsupervised neural text simplification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2058–2068.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Sara Tonelli, Alessio Palmero Aprosio, and Francesca Saltori. 2016. Simpitiki: a simplification corpus for italian. *Proc. of CLiC-it*.

David Vickrey and Daphne Koller. 2008. Sentence simplification for semantic role labeling. In *Proceedings of ACL-08: HLT*, pages 344–352, Columbus, Ohio. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 409–420.

Kristian Woodsend and Mirella Lapata. 2017. Text rewriting improves semantic role labeling (extended abstract). In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 5095–5099.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Xingxing Zhang and Mirella Lapata. 2017a. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

Xingxing Zhang and Mirella Lapata. 2017b. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium. Association for Computational Linguistics.

Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. 2018. Integrating transformer and paraphrase rules for sentence simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3164–3173, Brussels, Belgium. Association for Computational Linguistics.

Xiaojun Wan Zhe Lin. 2021. Neural sentence simplification with semantic dependency information. *AAAI Workshop on Deep Learning on Graphs: Methods and Applications*.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361.