1-1-2019

Towards Misleading Connection Mining

Md Main Uddin Rony

TOWARDS MISLEADING CONNECTION MINING

A Thesis
presented in partial fulfillment of requirements
for the degree of Master of Science
in the Department of Computer and Information Science
The University of Mississippi

by

Md Main Uddin Rony

August 2019

ABSTRACT

This study introduces a new Natural Language Generation (NLG) task – Unit Claim Identification. The task aims to extract every piece of verifiable information from a headline. The Unit Claim identification has applications in other domains; such as fact-checking where the identification of each verifiable information from a check-worthy statement can lead to an effective fact-check. Moreover, the extracting of the unit claims from headlines can identify a misleading news article, by mapping evidence from contents. For addressing the unit claim identification problem, we outlined a set of guidelines for data annotation, arranged in-house training for the annotators and obtained a small dataset. We explored two potential approaches - 1) Rule-based approach and 2) Deep learning-based approach and compared their performances. Although the performance of the deep learning-based approach was not very effective due to small number of training instances, the rule-based approach showed a promising result in terms of precision (65.85%).

DEDICATION

I would like to dedicate this work to my beloved "Nanavai" (Grandfather) who passed away last year. He had always been my inspiration since my childhood. I wish he could see me graduate.

ACKNOWLEDGEMENTS

Foremost, I would like to express my gratitude to ALMIGHTY ALLAH for His graces and blessings without which this work wouldn't be successful. After that, I would like to thank my thesis committee chair Dr. Naeemul Hassan and members, Dr. Dawn E. Wilkins, Dr. Yixin Chen, and Dr. Kristen Alley Swain, for their endless support and cordial guidance. Although they supervised me throughout my work, in no way they are responsible for any mistake of this work. I appreciate your direction and assistance. Last but not the least, my sincere thanks goes to my parents, my elder brother and my beloved wife for their continuous support throughout my graduation work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 *Unit Claim* Identification

Unit Claim identification is the task of identifying each of the verifiable claims of a news headline. A claim is a proposition or an idea that is verifiable, in other words, that is either true or false (Palau and Moens (2009)). Generally, a news headline contains multiple claims. Consider the following headline –

"Super PAC backing Jeb Bush unlikely to hit $100 million by end of June."

This headline is claiming that *Jeb Bush* is being backed by *Super PAC* and that he is unlikely to hit 100 million by the end of June. We define each of these claims as a *Unit Claim*. Specifically, we define each of the verifiable claims of a headline as a *Unit Claim* (*UC*). The goal of this task is to identify such *UC*s from a given headline. So, given the above headline, our goal is to find the following *UC*s –

UC 1: *Super PAC is backing Jeb Bush.*

UC 2: *Jeb Bush is unlikely to hit* $100 *million by the end of June.*

Note that, a *UC* can be written in various ways. For example, we can write the **UC 1** as *Jeb Bush is backed by Super PAC*. However, a *UC* shouldn't omit important and relevant contexts. For example, we should not write **UC 2** as *Jeb Bush is unlikely to hit* 100 *million*. Because, it omits the important context *end of June* of the unit claim. Below is another example of *UC* identification from a headline.

Example 1.1.1.

Headline*: "USA Swimming Bans Brock Turner Forever"*

UC 1: *USA Swimming Bans Brock Turner.*

UC 2: *The ban is forever.*

1.2 Application of *Unit Claim* Identification

Unit Claim identification task can be useful in many applications. For example, in the fact-checking task, the truthfulness of claims made by public figures such as politicians, pundits, etc are assessed to avoid spreading false information (Vlachos and Riedel (2014)). As a single statement may contain several claims, identifying the *UC*s of the statement may help in checking each claim precisely and hence produce better correctness. The statement of the former US president Barack Obama showed in (Vlachos and Riedel (2014)) contains several verifiable information, each of them is identified (underlined) here as a *UC* (Example 1.2.1).

Example 1.2.1.

"For the first time in over a decade, business leaders around the world have declared that China is no longer the worlds No. 1 place to invest; America is – President Barack Obama"

So, automated fact-checking systems can consider each of the *Unit Claim* for verification and determine whether the entire statement is TRUE, FALSE, MOSTLY TRUE or HALF TRUE.

Unit Claim identification can also be helpful in recognizing "False Connection" which is one of the seven types of mis- and disinformation (Wardle and Derakhshan (2017)). A false connection can be defined as a scenario *"When headlines, visuals or captions do not support the content"*. Example 1.2.2 shows an instance of False connection. Just after reading the headline, an ordinary reader may think that the Starbucks will be permanently shutting down 8,000 of their stores. But in the content we find that it was planning to close for several hours. So, here the headline is misrepresenting the content and the initial impression created from the headline went wrong after going through the full content.

Example 1.2.2.

Headline: Starbucks will close 8,000 US stores May 29 for racial-bias training [1]

Content: Starbucks plans to close more than 8,000 U.S. stores for several hours next month to conduct racial-bias training for nearly 175,000 workers. This comes after two black men were arrested in one of its stores in Philadelphia.

To detect such misleading connections automatically, we can identify each claim presented in the headline and then computationally check its evidence from the content, then the incongruency between the headline and content can be identified. So, *Unit Claim* identification can be a serviceable step for this process also.

1.3 Overview of proposed approach

As we discussed earlier *Unit Claim* identification has application in multiple domains, to our best knowledge, no previous work addressed this issue before and also there is no suitable dataset for this task. So, in this study, we address the *Unit Claim* Identification task by building a novel dataset and exploring two possible methods. In the first method, we develop some rules based on some Natural Language Processing (NLP) annotations to identify unit claims of a headline. The second approach mainly explores LSTM (Long Short-Term Memory) based encoder-decoder architecture as a possible solution by considering *Unit Claim* identification as a sequence-to-sequence generation task.

Before that, we explored some sample headlines to develop an annotation scheme. This annotation scheme is used for in-house training of the annotators. The annotators then work individually on identifying unit claims which results in a dataset of 1,052 annotated headlines.

So, in this study, our contributions are as follow:

- We built a small but novel dataset for *Unit Claim* identification task.

[1] https://www.usatoday.com/videos/news/nation/2018/04/18/starbucks-close-8000-us-stores-may-29-racial-bias-training/33943675/

- We explored two possible solutions for *Unit Claim* identification.
- We also analyzed the proposed solutions' performances and identified their strength and weakness.

The rest of the thesis is organized as follows: Chapter 2 summarizes the works from the areas which are closely related to *Unit Claim* identification; Chapter 3 expounds the steps of Data Collection process; Chapter 4 presents the detailed description of our two explored solutions; Chapter 5 shows the experimental results on our dataset in details; Chapter 6 expands some discussions on the explored methods and discussed the limitations of our work; Chapter 7 concludes this work.

CHAPTER 2

RELATED WORK

2.1 Claim Identification

Claim identification is an important step of the automated fact-checking process and many researchers addressed the problem before (Vlachos and Riedel (2014); Hassan et al. (2015); Patwari et al. (2017)). But all of them considered a statement as a single claim whether it contains single or multiple verifiable information. For example, the statement from Barack Obama shown in Example 1.2.1 was considered as a claim, where we can see it contains more than one verifiable statement (Underlined portions).

There are some publicly available datasets for fact-checking task where claims are labeled into different truthful categories. Thorne et al. (Thorne et al. (2018)) released a dataset, FEVER, for fact extraction and verification which consists of 185,445 claims generated by altering sentences extracted from Wikipedia. In this work, annotators were asked to generate a set of claims containing a single piece of verifiable information. So this dataset is not ideal for fact extraction task where a single sentence can contain multiple verifiable information e.g. news headlines. Some previous works focused on political debates to collect political statements and identified the set of statements with 'check-worthy' claims (Hassan et al. (2015); Patwari et al. (2017)). Their work and the datasets were also designed by focusing on a single statement as a single claim regardless of the presence of multiple verifiable unit claims.

2.2 Textual Entailment

Textual Entailment is a useful method in a wide range of natural language processing tasks and can be defined as methods that recognize, generate, or extract pairs $< T, H >$ of

natural language expressions, such that a human who trusts T (*Premise*) would infer that H (*Hypothesis*) is most likely also true (Dagan et al. (2005)). Example 2.2.1 shows an instance of textual entailment where the hypothesis is entailed from the premise.

Example 2.2.1.

Premise: If you help the needy, God will reward you.

Hypothesis: Giving money to the poor has good consequences.

There is a close resemblance between Unit Claim identification and Textual Entailment identification tasks. If the headline is considered as a premise, then each of the unit claims extracted from the headline can be considered as a hypothesis and will be textually entailed from the headline. For instance, the unit claims "USA Swimming Bans Brock Turner" and "The ban is forever" shown in example 1.1.1 entail the headline "USA Swimming Bans Brock Turner Forever".

But there is a subtle difference between textual entailment and unit claim identification. The hypothesis containing information inferred from the premise's information yields an entailment to the premise but in unit claim identification we only consider the information that can be directly perceived from the headline. Moreover, all the previous works on textual entailment focused on determining whether a textual pair forms a textual entailment rather than generating all the possible set of hypothesis where the hypothesis is entailed from the premise (Androutsopoulos and Malakasiotis (2010)). Although the task is different from our purpose, we can still use it for checking the quality of the unit claims generated from a headline, as the headline is supposed to entail the strong unit claims more.

2.3 Sentence Simplification

Sentence simplification is the task of reducing the linguistic complexity of a text, while still retaining its original information and meaning (Zhang and Lapata (2017)). Some sentence simplification tasks focused on replacing complex words with simpler substitutes

(Biran et al. (2011); De Belder and Moens (2010)), whether some other works directed towards syntactic simplification. The goal of syntactic simplification is to identify grammatical complexities in a text and rewriting these into simpler structures (Shardlow (2014)). Split-and-Rephrase is a type of syntactic simplification task where the aim is to split a complex sentence into a meaning preserving sequence of shorter sentences (Narayan et al. (2017)). For example, the complex sentence "John Clancy is a labour politician who leads Birmingham, where architect John Madin, who designed 103 Colmore Row, was born" would be splitted into shorter sentences ("Labour politician, John Clancy is the leader of Birmingham", "John Madin was born in this city", "He was the architect of 103 Colmore Row") by the split-and-rephrase method.

Although at first glance, the unit claim identification task seems to be similar to Split-and-Rephrase task, we argue that there is a remarkable difference between them. Sometimes a simple sentence can contain multiple unit claims which can't be extracted by mere Split-and-Rephrase process. For example, the headline "USA Swimming Bans Brock Turner Forever" is a simple sentence which results into two unit claims ("USA Swimming Bans Brock Turner" and "The ban is forever") but the headline is unlikely to be split into shorter sentences by the split-and-rephrase process.

CHAPTER 3

DATA COLLECTION

3.1 Dataset selection for annotation

The goal of this study is to identify the unit claims of the given news headline. In the future, we want to expand this work to identify incongruency between the headline and news content. So, to annotate we need a dataset containing news articles. Although there are many news article datasets (Horne et al. (2018); Yoon et al. (2018)), we choose the **Clickbait Challenge** dataset for two main reasons. First, clickbait challenge dataset contains news articles with clickbait decision (clickbait or non-clickbait). As clickbait articles are more likely to be misleading in nature, this dataset will be useful in our future work where we will try to develop an automated system to identify misleading news articles. Second, this dataset also contains news article related metadata (e.g. news content, media, etc.) which can be useful for our future development also. Clickbait challenge organization committee[1] provided three datasets for competition, two of them are labeled and one is unlabeled. The smaller labeled dataset contains 2,495 articles (762 clickbait, 1,697 non-clickbait), whether the larger one has 19,538 news articles (4,761 clickbait and 14,777 non-clickbait). For annotation purpose, we proceeded with the smaller dataset.

3.2 Data Annotation Scheme Design

We randomly picked 100 samples (50 clickbait and 50 non-clickbait) from clickbait challenge dataset and explored them for creating the annotation scheme. The goal of the annotation scheme is to provide some guidelines to the annotators so that they can extract

[1] https://www.clickbait-challenge.org/

Unit Claims (UCs) from news headlines. We provided 12 rules with proper explanation and examples so that the annotators could get enough knowledge to identify unit claims. For a better understanding of the annotation task, we manifest the guidelines here.

Rule 1: Any information that is subject to verification will be considered to form a **UC**. Consider the following example:

Example 3.2.1.

Headline*: Unemployment rates up in 90 percent of U.S. cities*

UC 1*: Unemployment rates go up.*

UC 2*: The rate increased in 90 percent of the U.S. cities.*

The unemployment rate scenario which is described in Example 3.2.1 is a verifiable information because we need to check whether it is actually going up or not (**UC 1**).

Rule 2: Does the headline contain any numerical information? If it does, then the numerical information can be formed into a **UC**.

Example 3.2.1 contains a piece of numerical information (*90 percent*) which results in **UC 2**.

Rule 3: Does the headline contain any adjective or phrase that is modifying or attributing an entity or action? If it does, the modifying word/phrase will form a **UC**. Let's look at the following examples:

Example 3.2.2.

Headline*: Retiring 60-year-old teacher completely slays 'Uptown Funk' dance with her students*

UC 1*: The teacher is retiring.*

***UC 2**: The teacher is 60 years old.*

***UC 3**: The teacher slew 'Uptown Funk' dance.*

***UC 4**: She slew it completely.*

***UC 5**: She danced with her students.*

Example 3.2.3.

***Headline**: Only one in three unhappy NHS patients actually complain, says new survey*

***UC 1**: There is a new survey.*

***UC 2**: There are unhappy NHS patients.*

***UC 3**: Only one in three unhappy NHS patients actually complain.*

***UC 4**: New survey is the source.*

In Example 3.2.2, the adjective *retiring* and the phrase *60-year-old* are attributing the entity *Teacher*. So, we consider them as unit claims (**UC 1, UC 2**). Example 3.2.3 contains 2 adjectives, *unhappy* and *new*, which form **UC 1** and **UC 2** of the headline.

Rule 4: Does the headline contain an adverb that is subject to verify? If yes, then the adverb can be represented as a **UC**.

In Example 1.1.1 (*"USA Swimming Bans Brock Turner Forever"*), the adverb *forever* emphasizes the action *ban*. The reader might be interested in verifying whether the ban is actually forever or for the time being. That's why we consider this as a **UC** (**UC 2**). The same rule is applied for **UC 4** (*She slew it completely*) of Example 3.2.2.

Rule 5: Does the headline contain any exaggerating, vulgar or sensational word? These types of words are subject to verification, hence can be considered for **UC**. Check the following example:

Example 3.2.4.

Headline*: Jared Kushner fails security clearance; Trump's response is so outrageous*

UC 1*: Jared Kushner takes security clearance.*

UC 2*: He fails in the security clearance.*

UC 3*: Trump responds to this failure.*

UC 4*: Trumps response is outrageous.*

The writer used an exaggerating word (*outrageous*) to express her opinion towards the event (*Trump's response*) in Example 3.2.4. We need evidence to be convinced that the response is actually *outrageous*, so it has been formed as a **UC** (**UC 4**).

Rule 6: Is there any noun phrase in the headline which we need to verify? If it does, we will consider it as a **UC**.

In Example 3.2.4, the noun phrase *Trump's response* can be represented as a **UC** (**UC 3**) as the readers might need verification if Trump really responded to it.

Rule 7: What is/are the main verb/verbs of the headline? Turn each of them into **UC**.

The main verb of the headline presented in Example 3.2.4 is *fails*, which is considered for **UC 2**.

Rule 8: Does the headline contain any modal verb (e.g. may, might, etc.) which expresses a speakers attitude and the strength of that attitude? These type of words also show the degree of certainty of an event which we will consider for **UC**. Consider the following example:

Example 3.2.5.

Headline*: Caerphilly farmer may get full payout after 24 year wait*

***UC 1**: The farmer is from Caerphilly.*

***UC 2**: The farmer may get a payment.*

***UC 3**: The payment will be in full.*

***UC 4**: He will get the payment after waiting for 24 years.*

In Example 3.2.5, the modal verb *may* expresses the certainty level of the event (*getting the full payment*) which is a subject to verification and considered here as a **UC** (**UC 2**).

Rule 9: Does the headline mention any source? (e.g. Trump says, reported by CNN, etc.). The mention of the source can be converted into a **UC**.

UC 4 of Example 3.2.3 is a representative of this rule.

Rule 10: Does the headline contain an event-cause pair? If the headline contains any event-cause pair, then the event and cause will be considered individually for **UC**. Look at the following example:

Example 3.2.6.

***Headline**: Azeri government behind foreign media ban, say European Games officials*

***UC 1**: There is a foreign media ban.*

***UC 2**: Azeri government behind the ban.*

***UC 3**: European Games officials are the sources.*

In Example 3.2.6, a *foreign media ban* is an event which is caused by the *Azeri Government*. So, **UC 1** and **UC 2** are extracted following the above rule.

Rule 11: Does the headline contain an event and there are multiple actors involved in it? When multiple actors are involved in an event, we will create an individual claim for each of the involving actors. Here is an example:

Example 3.2.7.

Headline*: How theme parks like Disney World left the middle class behind?*

UC 1*: Disney World left the middle class behind.*

UC 2*: There are other theme parks left the middle class behind.*

There are two actors (*Disney world* and *Other theme parks*) involved in an event (*leaving middle class behind*) described in Example 3.2.7. So the **UC 1** and **UC 2** have been constructed separating the actors.

Rule 12: Is there any action in the sentence that is performed in association with some other entities? Or is the action performed over other entities? If so, the entities which are associated with the event and the entities on whom the action has been performed will be considered to form a **UC**.

In Example 3.2.2, the teacher danced (*action*) with her students (*associated entity*). So the association is subject to verify which leads to building up a **UC** (**UC 5**).

3.3 Data Annotation

Unit Claim identification is a complex task and annotators need to have a deep understanding to complete the task. That's why instead of using crowdsourced annotation service (e.g. Amazon Mechanical Turk) we arranged inhouse training for the annotators. There were 8 annotators in total. Five of them were female and three of them were male. Among them, six annotators were graduate students during the time of training, and two have just completed their undergraduate program. Only one annotator aged more than 30 and others age fell between 22-30.

Each annotator was provided with 100 headlines and their performance was measured when they finished annotating 10 headlines. The performance was measured based on the precision and recall of the identification task. For the unit claim identification task, the precision and recall are defined as the follows:

Annotators	Total UC	# of identified UC	# of correct UC	Precision (%)	Recall (%)
Annotator 1	42	36	31	86.11	73.81
Annotator 2	41	37	33	89.19	80.49
Annotator 3	39	33	28	84.85	71.79
Annotator 4	44	35	31	88.57	70.45
Annotator 5	38	45	33	73.33	86.84
Annotator 6	41	33	29	87.88	70.73
Annotator 7	37	31	26	83.87	70.27
Annotator 8	35	33	25	75.76	71.43

Table 3.1. Performance Evaluation of the Annotators

$$Precision = \frac{\text{Number of correctly identified unit claims}}{\text{Number of identified unit claims}}$$

$$Recall = \frac{\text{Number of correctly identified unit claims}}{\text{Actual Number of unit claims}}$$

To maintain the standard of the annotation work, we set the threshold of precision and recall to 70%. After the first round of training, only two annotators were able to achieve satisfactory performance. Then we arranged another round of training session for the annotators who didn't achieve the required precision-recall. After the second round of evaluation, all but two passed the barrier. We had to arrange the third round of training session for these two annotators, and after that, they successfully achieved the required performance. The overall training process and performance evaluation showed the complexity of the task and justified our decision of inhouse training session. Table 3.1 shows the performance details of the annotators.

3.4 Data Exploration

All the qualified annotators were given 100 headlines individually to annotate. One of the authors took the responsibility of annotating 300 headlines. Annotators were instructed to skip a headline if they find it difficult to annotate either for language complexity or lack of information. Table 3.2 shows the statistical descriptions of the dataset. In total there are 1,052 annotated headlines in the dataset. We got total 3,571 unit claims, so on average,

Dataset Statistics	
Total Headlines	1052
Total Unit Claims (UC)	3,571
Avg. no of UC per headline	3.39
Median no of UC per headline	3
Max no of UC in a headline	10
Avg. length of UC (tokens)	6.03
Median length of UC (tokens)	5
Clickbait (CB) headlines	527
Non-Clickbait (NCB) headlines	525
Avg. no of UC per CB headline	3.41
Avg. no of UC per NCB headline	3.37

Table 3.2. Unit Claim Dataset Statistics

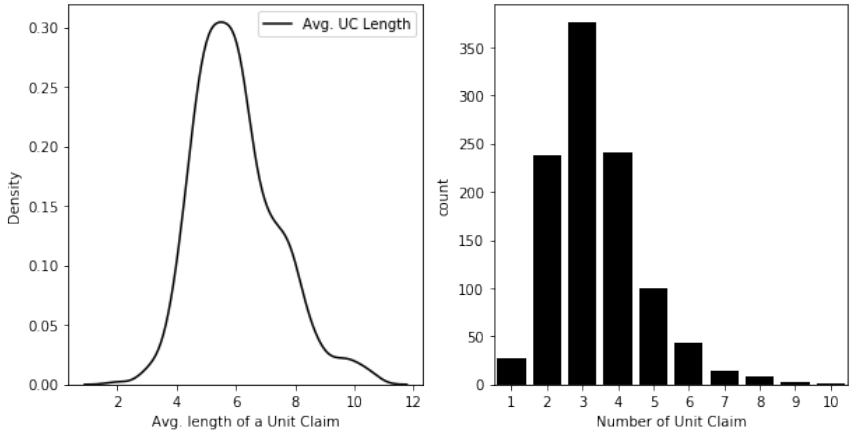

Figure 3.1. Density plot for average length of UC (Left) and UC count per headline (Right)

each headline contains 3.39 unit claims. We also computed the length of a unit claims in terms of the number of tokens it contains. On average, a unit claim has 6.03 tokens. Figure 3.1 shows the density plot the average unit claim length for a headline and the count plot for the number of unit claims per headline. The peak of the density plot indicates that the average length (in terms of the token) of a unit claim is around 6. On the other hand, from the count plot, we find that most of the headlines have close to 3 (median) unit claims.

We also inspected the distribution of unit claim count over clickbait and non-clickbait headlines. In our dataset, 527 headlines are clickbait and the average number of unit claims per clickbait headline (3.41) is slightly higher than the average unit claim count for the

non-clickbait headline (3.37). Although our hypothesis was clickbait headlines contain more verifiable information which would result in more unit claims than non-clickbait headlines do, the difference between the average number of unit claims of clickbait and non-clickbait headlines is too low to support the hypothesis.

CHAPTER 4

METHODOLOGY

4.1 Rule Based Method

To extract unit claims from the headlines we developed a set of rules based on some NLP annotations. In this section, we outline the annotation details and rules construction process.

4.1.1 NLP Annotations

4.1.1.1 Part-of-Speech Tagging

This annotation returns each token of the headline with their POS tag (Manning et al. (2014)). POS tags are helpful to identify the relations between words and capture the word sense. In the example shown in Figure 4.1[1], the adjective "perfect" is modifying the noun "wedding". This type of modification should be considered results in a unit claim "The wedding was perfect". Moreover, the adverb "nicely" followed by the verb "photographed" is also subject to verification hence will construct a unit claim "The dude photographed nicely".

4.1.1.2 Named Entity Recognition

Named Entity Recognition identifies named entities (location, numeric entry, person and company names, etc.) in text. This annotation helps us to extract relevant information

[1]The figure is generated from: http://corenlp.run/

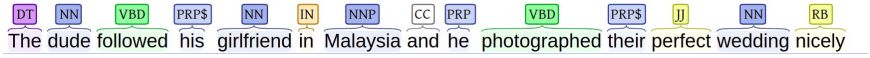

Figure 4.1. POS tagging

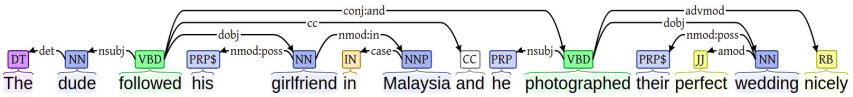

Figure 4.2. Dependency Parsing

which is important to identify the content which may be subject to verification.

4.1.1.3 Dependency Parsing

Dependency parsing (Kübler et al. (2009)) identifies syntactic structure of a sentence to describe the relationships between the words. These relationships were used as directional conditions to construct the unit claim identification rules. Figure 4.2[2] shows the dependency parsing of a sample headline. We can easily identify the subject-verb-object pattern from the parsing result (dude[nsubj] - followed[vbd] - girlfriend[dobj]) and construct the unit claim "The dude followed his girlfriend".

4.1.1.4 Coreference Resolution

Coreference resolution identifies the expressions that refer to the same entity in a text (Soon et al. (2001)). We used coreference resolution to resolve pronominal and nominal reference to extract unit claims from the headlines. For example, from the dependency parsing shown in Figure 4.2, we can see another subject-verb-object pattern (he[nsubj] - photographed[vbd] - wedding[dobj]) where "he" is the subject. From coreference resolution (shown in Figure 4.3) we can resolve this pronoun reference ("he" refers to "dude") and construct the unit claim "The dude photographed their wedding".

4.1.2 Rules Construction

Based on the annotation results, we developed a set of algorithms to extract unit claims. There are 9 basic algorithms which capture various predefined patterns using the annotation result. Algorithm 1 shows a sample algorithm for extracting a unit claim from the presence of an adjective in a headline.

[2]The figure is generated from: http://corenlp.run/

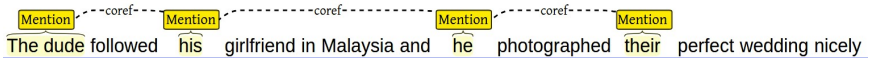

Figure 4.3. Coreference Resolution

Algorithm 1: Constructing Unit Claim for Adjective

input : Token List, $Tokens$; POS Tag List, POS; Syntactic dependencies, $Dependencies$

output: A list of Unit Claim of the form, "Noun" → "Auxiliary Verb" → "Adjective"

$unit\_claims = []$;

for $dependency \in Dependencies$ **do**

    // *amod* is a relation that connects a noun to an adjective

    if *dependency is amod* **then**

        // In *amod* relation, first index of *dependency* is for a Noun

        $noun \leftarrow Tokens[dependency[first]]$;

        /* In *amod* relation, second index of *dependency* is for an Adjective */

        $adj \leftarrow Tokens[dependency[second]]$;

        /* $identify_auxiliary_verb()$ is another function that defines the auxiliary verb based on different grammatical conditions */

        $aux\_verb \leftarrow identify\_auxiliary\_verb(POS, Tokens, noun\_index, adj\_index)$;

        $unit_claim \leftarrow join(noun, aux_verb, adj)$;

    end

end

return *unit_claims*

4.2 Deep Learning Based Method: Sequence-to-Sequence Approach

Sequence-to-Sequence model is a deep learning-based approach which is also known as an encoder-decoder model. This type of model has been used successfully in many text generation tasks such as machine translation (Sutskever et al. (2011); Bahdanau et al. (2014)),

document summarization (Rush et al. (2015)), etc. In this architecture, the encoder network takes the entire input sequence to encode to a fixed-length internal representation and the decoder network uses that internal representation to predict the output sequence until the sequence ending token is found. Many works (Rush et al. (2015); Sutskever et al. (2011); Bahdanau et al. (2014)) use a recurrent neural network (RNN) for enoding the input sequence and then use another RNN to decode the internal representation to target sequence.

Standard RNN architecture (Cho et al. (2014)) performs well on mapping sequences to sequences when the alignments between the inputs the outputs are perceived beforehand. But whenever the input and output sequences have different lengths with complicated and non-monotonic relationships, it's difficult for RNN to handle the problem. As the headlines as input sequences and unit claim the output sequences can vary in length, it would be difficult to map their alignments apriori, hence RNN based encoder-decoder might not be an ideal choice. LSTM (Long Short-Term Memory) (Hochreiter and Schmidhuber (1997)) model's success over learning problems with temporal dependencies inspires Sutskever et al. to use it for solving this type of problem (Sutskever et al. (2014)). Actually, LSTM is used to measure the conditional probability of the output sequences given the fixed dimensional representation of the input sequences with a different length than the output sequence. This fixed dimensional representation of the input sequence is obtained from the last hidden state of the LSTM and then used to compute the probability of the output sequence with a standard LSTM-LM formulation whose initial hidden state is set to the representation of the input state (Sutskever et al. (2014)). Mathematically,

$$p(y_1, ..., y_{T'}|x_1, ..., x_T) = \prod_{t=1}^{T'} p(y_t|v, y_1, ..., y_{t-1})$$

where $x_1, ..., x_T$ is the input sequence,$y_1, ..., y_{T'}$ is the output sequence , v is the fixed dimensional representation of the input sequence. To make the model able to define a distribution over different length sequences, an end-of-sentence symbol is required (e.g. "<EOS>" in Figure 4.4). Figure 4.4 is used by Sutskever et al. (2014) to outline the design

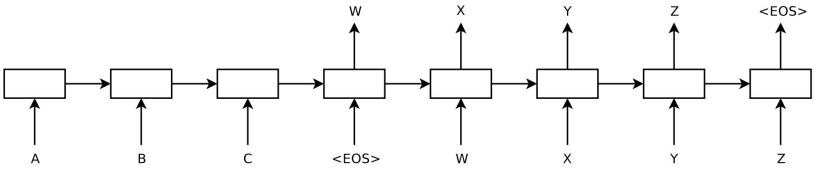

Figure 4.4. LSTM based encoder-decoder model for sequence-to-sequence generation task

of the model where the input sequence “A”, “B”, “C”, “<EOS>” is fed into the LSTM to compute the probability of output sequence “W”, “X”, “Y”, “Z”, “<EOS>”.

As our target is to produce character-level sequence-sequence generation, we fed the input character sequence into an LSTM layer which works as an encoder. The encoder layer processes the character sequence and returns the internal state which works as the “context” of the decoder in the next step. The decoder is another LSTM layer which predicts the next characters of the target sequence, given previous characters of the target sequence.

CHAPTER 5

EXPERIMENT

5.1 Experimental Setup

5.1.0.1 Rule Based Method

Before applying the set of predefined rules, the headlines were annotated using Stanford CoreNLP tools [1]. We set up the CoreNLP server in our local machine and used a python package named "stanford-corenlp" [2] to annotate the headlines. As we discussed earlier, we used POS tagging, dependency parsing, Named Entity Recognition, and Coreference resolution tools for annotation and applied some predefined rules on the annotated results to extract the unit claims from headlines. We didn't perform any kind of data cleaning such as lemmatization, stemming, or removing punctuation because lemmatization or stemming may hurt the performance of the annotations, and punctuation bears special significance in identifying the relations between words.

5.1.0.2 Deep Learning Based Method

We implemented a character-level sequence-to-sequence model, Seq2Seq which processes the input character by character and also generates output character by character. In our problem, the input is a headline (a single sentence) and output is a set of unit claims (a set of sentences). So, to keep the resemblance with the traditional sequence-sequence learning task, we converted the set of unit claims into a sequence by concatenating them one after another by a special symbol (<UC>). The symbol is used to determine the boundaries

[1] https://stanfordnlp.github.io/CoreNLP/index.html
[2] https://pypi.org/project/stanford-corenlp/

between the claims. We implemented the model following the steps described in the Keras Blog [3].

The headline and the unit claim sequences are converted into one-hot vectorizations to use as encoder input and the decoder input respectively. There is another hot-vector representation to use as decoder output which is the same as decoder input but their offset differs by one timestep. Then we trained the Seq2Seq model to predict the decoder output sequence given the encoder input and decoder input. Our model was built on LSTM layers with 256 hidden states. We performed mini-batch training with a batch size of 64 sentences. The model was trained on 1,000 samples for 100 epochs and for monitoring the loss of the training a hold-out was set to 20% of the samples. We used RMSprop as the optimizer, softmax as activation function, and categorical cross-entropy as a loss function.

5.2 Experimental Results

5.2.1 Evaluation Metric

The Bilingual Evaluation Understudy (BLEU) (Papineni et al. (2002)) is used to evaluate the quality of the generated texts in many natural language generation tasks such as machine translation (Sutskever et al. (2014); Davoodi et al. (2018); Cho et al. (2014), sentence simplification (Zhang and Lapata (2017); Narayan et al. (2017)), and so on. We also used BLEU score to assess the degree to which generated unit claims differed from the gold standard references (generated by the annotators). Table 5.2 shows the score of N-gram BLEU scores over different values of N. We can see the decrease of BLEU score with an increase of N and the reason is quite obvious. Matching only single word produces a better matching score than matching multiple words in a specific order. As there is no prior work on unit claim identification, we couldn't compare the performance of our model directly. But we compared the performance with other NLG tasks (Table 5.3). Although the other NLG tasks outperform the rule-based approach, they used comparatively larger datasets to train

[3] https://blog.keras.io/a-ten-minute-introduction-to-sequence-to-sequence-learning-in-keras.html

Performance Evaluation Statistics	
Testing Samples	100
# of actual Unit Claims	366
# of identified Unit Claims	246
# of correctly identified Unit Claims	162
Avg. Unit Claim length (tokens)	4.90
Precision	65.85%
Recall	44.26%

Table 5.1. Performance evaluation of rule-based approach

N-gram BLEU (N)	Score
1	0.22
2	0.16
3	0.12
4	0.09

Table 5.2. N-gram BLEU score of Rule-based approach

their models (The sizes of the training corpus for Sutskever et al. (2014) and Narayan et al. (2017) are 12M and .9M approximately).

Although unit claim identification is also a natural language generation task, BLEU is not an optimal metric for this because: First, the wording and structure of the unit claims can be different. So, BLEU which is used to evaluate the quality of the generated texts might not be useful in our task. Second, the performance of the unit claim identification mainly depends on two factors: how many units claims the model can identify and how many of them are correctly identified. So mere evaluation of the quality of the text can't capture the effectiveness of the model. That's why we also evaluated the performance of the models based on precision and recall which we defined in section 3.3.

5.2.2 Performance Analysis of Rule-based Approach

We applied the models on 100 sample headlines. Table 5.1 shows the performance of the rule based method. The model extracted 246 unit claims (2.46 unit claims per headline) where the actual number of unit claims were 366 (3.66 unit claims per headline). So, clearly the model generates fewer unit claims than the ideal scenario. The average length of the

NLG Task	BLEU Score
Machine Translation (Sutskever et al. (2014))	0.348
Split-Rephrase (Narayan et al. (2017))	0.78
Rule-based UC identification	0.22

Table 5.3. Performance comparison of Rule-based approach with other NLG tasks

Rules	# of Unit Claim (%)	# of Correct Unit Claim (%)
Rule 1	18 (7.21%)	13 (72.22%)
Rule 2	81 (32.93%)	55 (67.90%)
Rule 3	19 (7.72%)	8 (42.11%)
Rule 4	52 (21.14%)	31 (59.62%)
Rule 5	13 (5.28%)	13 (100%)
Rule 6	19 (7.72%)	10 (52.63%)
Rule 7	20 (8.13%)	14 (70%)
Rule 8	11 (4.47%)	9 (81.82%)
Rule 9	13 (5.28%)	9 (69.23%)
Total	246	162

Table 5.4. Performance breakdown of the rules

unit claims (in terms of token count) generated by the model is 4.90 which is lower than the average length of UC of the training samples (6.03). As our rule-based method generates text based on some simple patterns (e.g. subject-verb-object), it uses less word than an average human does although both the texts may express the same meaning. For example, *"The dishes are brilliant"* is a human-generated unit claim where the corresponding rule-generated unit claim is *"Dishes are brilliant"*, having less number of word. Among 246 identified unit claims, 162 were correct. So, the model can generate on average 1.62 unit claims per headline which is around 44% of the actual unit claims. Although the precision looks satisfactory (65.85%), but the recall is much lower (44.26%). The result shows that we need to devise more rules to identify the larger number of unit claims.

We further investigated the performance of each rule. Table 5.4 shows the breakdown of the evaluation. Also, Table 5.5 and Table 5.6 list some correctly and incorrectly generated unit claims by the rules we developed. From Table 5.4, we can see that most of the unit claims are identified by Rule 2 and Rule 4. Rule 2 mainly looks for adjectives in a headline

and converts them into unit claims, and its identification rate is quite high because reporters often use powerful adjectives for creating eye-catching headlines. In Rule 4, we scanned for the *"Subject-Verb-Object"* pattern which is quite common in news headlines as reporters generally summarize the action of an event through headlines. This justifies the high number of unit claims identified by Rule 4. The other rules have also contributed some UCs, but the number is pretty low compared to Rule 2 and Rule 4, which explains the scarcity of common patterns exist in news headlines.

In terms of accuracy, Rule 5 outperforms other rules. This is pretty obvious because this rule is constructed based on the presence of "WH words" (what, why, how, etc.) in the headline. This rule just maps the "WH words" to generate some predefined texts (e.g. *"The article describes"*, *"The article explains"*, etc.) to construct a unit claim. We can see some examples of this scenario in Table 5.5 (Rule 5). Rule 2 and Rule 4 also show some satisfactory accuracies. Usually, adjectives and action expressing patterns (*Subject-Verb-Object*) are always subject to verification which results in higher accuracies. Examples listed for Rule 2 and Rule 4 in Table 5.5 corroborate this claim. Rule 1, Rule 8, and Rule 9 have also high accuracies but the number of unit claims identified by them is not high enough to provide strong support.

News headlines often have complex syntactical structures which NLP tools find sometimes difficult to parse and annotate correctly. As the rules are developed based on the annotation results of the NLP tools, the annotation performance has an effect on our unit claim identification task. For example, the unit claim generated by Rule 2 in Table 5.6 is inaccurate because POS tagger incorrectly identifies *"Play"* as an adjective modifying the noun *"Ball"*. Moreover, the rule built on a specific syntactic relation fitting in a particular scenario can produce inaccurate context in other scenarios. For example, in order to extract a claim, Rule 6 scans for a particular pattern where a noun has a 'nsubj' relation with another noun and at the same time it has a 'cop' relation with a verb. This rule successfully produces a unit claim for the headline *"eSports is a massive industry ... and growing"*

Rules	Headline	Correctly Generated UC
Rule 1	1. 5 things Australians need to know from Apple's 2015 WWDC keynote 2. Philae, Europe's Comet Lander, Wakes Up After Seven Months	1. The WWDC Keynote is from Apple 2. The Comet Lander is from Europe
Rule 2	1. Why Ariana Grande's Feminist Twitter Post Was a Brilliant Career Move 2. Gay Hair Salon Owner Installs Anti-bigotry Sign After Homophobic Incident	1. Twitter Post is feminist 2. Salon Hair Owner is gay
Rule 3	1. Corset Training, a celebrity weight loss trend, largely busted 2. NAACP Leader Rachel Dolezal Allegedly Faked Being A Black Woman For Years	1. Corset Training largely busted 2. Rachel Leader Naacp Dolezal allegedly faked
Rule 4	1. Gay Hair Salon Owner Installs Anti-bigotry Sign After Homophobic Incident 2. Man unearths dads never-before-seen footage of JFK	1. Salon Hair Owner installs Sign 2. Man unearths Footage of JFK
Rule 5	1. Why Arsenal need Zlatan Ibrahimovic 2. Study finds how your birth month affects your health	1. This article explains why Arsenal need Zlatan Ibrahimovic 2. This article shows how your birth month affects your health
Rule 6	1. Why Ariana Grande's Feminist Twitter Post Was a Brilliant Career Move 2. eSports is a massive industry ... and growing	1. Twitter Post was a Career Move 2. ESports is an industry
Rule 7	1. 5 things Australians need to know from Apple's 2015 WWDC keynote 2. Two Women Get Into Wild Brawl In Walmart Shampoo Aisle, Child Joins In	1. There are 5 things 2. There are two women
Rule 8	1. Snowden files 'read by Russia and China': five questions for UK government 2. Fifty Conservative MPs to challenge David Cameron over 'rigged' EU referendum rules	1. Snowden Files read by Russia 2. MPs challenge over Referendum Rules
Rule 9	1. Snowden files 'read by Russia and China': five questions for UK government 2. Antwerp now has 'text lanes' for pedestrians who are glued to their mobile phones	1. Questions are for UK Government 2. Text Lanes are for Pedestrians

Table 5.5. Samples of correctly generated unit claims by the rules

(Table 5.5) where the noun *"industry"* is connected to another noun *"eSports"* by 'nsubj' relation and the verb *"is"* connected to *"industry"* by 'cop' relation. But the same rule extracts a meaningless unit claim (Table 5.6) for the headline *"The 'Obama is a Muslim' conspiracy theory gets a Shiite twist from a former Iraqi lawmaker"* although the syntactic structures are the same for both cases (*"conspiracy theory"* as a noun connected to another noun *"Obama"* by 'nsubj' relation and the verb *"is"* is connected to *"conspiracy theory"* by 'cop' relation).

5.2.3 Performance Analysis of Se2Seq Model

Due to the low number of training instances, our deep learning-based model didn't perform well. Each time it generates only two unit claims, but they are meaningless. So, we couldn't measure the performance for this model. Table 5.7 shows some examples generated by the sequence-sequence learning model and rule based models. One conspicuous thing is that each sequence generated by Seq2Seq model starts with "the source is". It happens because there are some headlines which mention the source of the information and we model the unit claim for this source mention following a particular pattern: "The source is *source_name*". For example, the headline *"Only one in three unhappy NHS patients actually complain, says new survey"* contains a source mention (*"says new survey"*) and and this produces a unit claim (*"The source is new survey"*). Rule 9 described in Section 3.2 gives

Rules	Headline	Incorrectly Generated UC
Rule 1	This women's rugby player broke her nose	The Player is from This Women
Rule 2	Play Ball! Independent Baseball Does Major League Business	Ball is play
Rule 3	Snack Bars Push the Price Envelope and Find Consumers Dont Push Back	Consumers don back
Rule 4	100 things that happened this season (and you might have forgotten)	That happened This Season
Rule 6	The 'Obama is a Muslim' conspiracy theory gets a Shiite twist from a former Iraqi lawmaker	The Obama is a A Conspiracy Theory
Rule 7	Jim Brown honored by LeBron James acknowledgement before Game 3	There is 3 game
Rule 8	Rolf Harris writes song mocking his victims as money-grabbing 'wenches', report claims	Song mocking ' Wenches
Rule 9	Whats Working Honor Roll: The Case For Teaching Students Emotional Skills, Not Just Academic Ones	The Case is for Students Teaching Skills

Table 5.6. Samples of incorrectly generated unit claims by the rules

Headlines	Actual Unit Claims	Unit Claims from seq2seq model	Unit Claims from Rule based model
Brilliant Italian dishes you haven't tried before	1. This article shows some dishes we haven't tried before 2. The dishes are Italian 3. The dishes are brilliant	1. the source is a succe in sead. 2. the source is about the sead in wat	1. Dishes are brilliant 2. Dishes are italian
Mystery man donates wedding dress with touching note about late wife	1. The man is mysterious 2. The man donates wedding dress 3. The man donates wedding dress with note 4. The note is touchy 5. The note is about his wife 6. His wife is late	1. the source is about to gea sou comen 2. the source is about the sead in wat	1. The note is touching 2. Wife is late 3. Mystery Man donates Wedding Dress 4. Mystery Man donates Wedding Dress with Note 5. Mystery Man donates Wedding Dress about Wife
Little kid breaks opponent's ankles with nasty crossover	1. A kid breaks someone's ankles 2. The kid is little 3. The kid breaks his opponent's ankles 4. The kid breaks his opponent's ankles with crossover 5. The crossover was nasty	1. the source is about to gea s age to gea 2. the source is about the sead in wat	1. The Ankles are from Breaks Kid Opponent 2. Kid is little 3. Crossover is nasty
Duggar Family Scandal: officials called 911 after visiting home	1. The article is about a Duggar Family scandal 2. Officials called 911 3. Officials called 911 after visiting home	1. the source is a succe in sead 2. the source is about the sead in wat	1. Officials called 911
Samira Wileys opinion on 21 random things	1. The article shows opinion of Samira Wiley 2. Samira Wiley gave opinion on some random things 3. There are 21 random things 4. The things are random	1. the source is a succe in sead2. 2. the source is about the sead in wat	1. The Opinion is from Samira Wiley 2. Things are random 3. There are 21 things

Table 5.7. Some samples of unit claims generated by rule-based and Seq2Seq models

more details of this scenario. Looks like Seq2Seq model only learns this pattern from our training instances. This finding also infers that we need more training samples associated with each rule so that the model can learn the other patterns successfully.

CHAPTER 6

DISCUSSION

We have developed two computational models for extracting unit claims from headlines. Our first model is based on some predefined rules which work on annotation results produced by some traditional NLP tools. In the second approach, we designed a character level seq2seq generator where the encoder and decoder are mainly LSTM layers. Although LSTM based encoder-decoder architecture has been very successful for sequence generation task (e.g. Machine Translation), it didn't perform well with unit claim identification task. The obvious reason for this poor performance is a very small number of training samples. On the other hand, the rule-based approach has yielded a satisfactory precision but the recall of the process is not up to the mark. This low recall demands more variation of rules to cover more unit claims. But it's always difficult to come up with a complete set of rules that covers all the variations of unit claims. For example, the headline *"Airport Expansion: Heathrow vs Gatwick"* would result in 3 unit claims (*"Heathrow Airport is expanding"*, *"Gatwick Airport is expanding"*, and *"This article compares the expansion between the Heathrow Airport and Gatwick Airport"*). This type of headline is really difficult to handle by rule-based method and this difficulty was our primary motivation to explore deep learning-based approach. Before exploring the models, we also designed the data annotation guidelines for the annotators. The performance of the annotators has shown that unit claim identification is a time consuming and complex task which requires much expertise.

CHAPTER 7

CONCLUSION

In this study, we outlined the unit claim identification task, designed an annotation scheme for data collection and explored two potential methods for extracting unit claims automatically. The annotation schemes were used to train the annotators and with their help, we collected a small dataset of 1, 052 annotated news headlines. Among the attempted methods, the rule-based approach showed comparatively better performance than the deep learning-based seq2seq generation approach. The problem became difficult due to a low number of training samples. Unit claim identification has potential applications in many domains such as fact-checking, false connection identification, etc. As there is a shortage of data for this task, so data collection along with the exploration of new possible solutions can be future research directions.

BIBLIOGRAPHY

BIBLIOGRAPHY

Androutsopoulos, I., and P. Malakasiotis (2010), A survey of paraphrasing and textual entailment methods, *Journal of Artificial Intelligence Research*, 38, 135–187.

Bahdanau, D., K. Cho, and Y. Bengio (2014), Neural machine translation by jointly learning to align and translate, *arXiv preprint arXiv:1409.0473*.

Biran, O., S. Brody, and N. Elhadad (2011), Putting it simply: a context-aware approach to lexical simplification, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pp. 496–501, Association for Computational Linguistics.

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

Dagan, I., O. Glickman, and B. Magnini (2005), The pascal recognising textual entailment challenge, in *Machine Learning Challenges Workshop*, pp. 177–190, Springer.

Davoodi, E., C. Smiley, D. Song, and F. Schilder (2018), The e2e nlg challenge: Training a sequence-to-sequence approach for meaning representation to natural language sentences, in *in prep. for INLG conference*.

De Belder, J., and M.-F. Moens (2010), Text simplification for children, in *Proceedings of the SIGIR workshop on accessible search systems*, pp. 19–26, ACM Press New York.

Hassan, N., C. Li, and M. Tremayne (2015), Detecting check-worthy factual claims in presidential debates, in *Proceedings of the 24th acm international on conference on information and knowledge management*, pp. 1835–1838, ACM.

Hochreiter, S., and J. Schmidhuber (1997), Long short-term memory, *Neural computation*, 9(8), 1735–1780.

Horne, B. D., S. Khedr, and S. Adali (2018), Sampling the news producers: A large news and feature data set for the study of the complex media landscape, in *Twelfth International AAAI Conference on Web and Social Media*.

Kübler, S., R. McDonald, and J. Nivre (2009), Dependency parsing, *Synthesis Lectures on Human Language Technologies*, 1(1), 1–127.

Manning, C., M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky (2014), The stanford corenlp natural language processing toolkit, in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pp. 55–60.

Narayan, S., C. Gardent, S. B. Cohen, and A. Shimorina (2017), Split and rephrase, *arXiv preprint arXiv:1707.06971*.

Palau, R. M., and M.-F. Moens (2009), Argumentation mining: the detection, classification and structure of arguments in text, in *Proceedings of the 12th international conference on artificial intelligence and law*, pp. 98–107, ACM.

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

Patwari, A., D. Goldwasser, and S. Bagchi (2017), Tathya: A multi-classifier system for detecting check-worthy statements in political debates, in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 2259–2262, ACM.

Rush, A. M., S. Chopra, and J. Weston (2015), A neural attention model for abstractive sentence summarization, *arXiv preprint arXiv:1509.00685*.

Shardlow, M. (2014), A survey of automated text simplification, *International Journal of Advanced Computer Science and Applications*, 4(1), 58–70.

Soon, W. M., H. T. Ng, and D. C. Y. Lim (2001), A machine learning approach to coreference resolution of noun phrases, *Computational linguistics*, 27(4), 521–544.

Sutskever, I., J. Martens, and G. E. Hinton (2011), Generating text with recurrent neural networks, in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 1017–1024.

Sutskever, I., O. Vinyals, and Q. V. Le (2014), Sequence to sequence learning with neural networks, in *Advances in neural information processing systems*, pp. 3104–3112.

Thorne, J., A. Vlachos, C. Christodoulopoulos, and A. Mittal (2018), Fever: a large-scale dataset for fact extraction and verification, *arXiv preprint arXiv:1803.05355*.

Vlachos, A., and S. Riedel (2014), Fact checking: Task definition and dataset construction, in *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pp. 18–22.

Wardle, C., and H. Derakhshan (2017), Information disorder: Toward an interdisciplinary framework for research and policy making, *Council of Europe report, DGI (2017)*, 9.

Yoon, S., K. Park, J. Shin, H. Lim, S. Won, M. Cha, and K. Jung (2018), Detecting incongruity between news headline and body text via a deep hierarchical encoder, *arXiv preprint arXiv:1811.07066*.

Zhang, X., and M. Lapata (2017), Sentence simplification with deep reinforcement learning, *arXiv preprint arXiv:1703.10931*.

VITA

Education

B.S. Computer Science and Engineering 2014

Bangladesh University of Engineering and Technology, Bangladesh

Employment

Research Assistant 2017 - Current

University of Mississippi, US

Software Engineer 2015 - 2016

Infolytx, Inc., Bangladesh

Junior Software Engineer 2014 - 2015

Nascenia IT, Bangladesh,

Publications

Dhoju, S., Main Uddin Rony, M., Ashad Kabir, M., & Hassan, N. (2019, May). Differences in Health News from Reliable and Unreliable Media. In Companion Proceedings of The 2019 World Wide Web Conference (pp. 981-987). ACM.

Rony, M. M. U., Hassan, N., & Yousuf, M. (2018, April). BaitBuster: A Clickbait Identification Framework. In Thirty-Second AAAI Conference on Artificial Intelligence (pp. 8216-8217).

Rony, M. M. U., Hassan, N., & Yousuf, M. (2017, July). Diving deep into clickbaits: Who use them to what extents in which topics with what effects?. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017 (pp. 232-239). ACM.