Improving Neural Net Machine Translation Systems with Linguistic Information

by
Yuan-Lu Chen

A Dissertation Submitted to the Faculty of the
DEPARTMENT OF LINGUISTICS
In Partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy
In the Graduate College
The University of Arizona

2018

Get the official approval page
from the Graduate College
before your final defense.

Statement by Author

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

Signed: _____

Dedication

For Eva, the best linguist and the most powerful neual machine.

Acknowledgments

acknowledgment! Many people help me and keep me company in my this journey. Thank you.

Contents

CONTENTS—*Continued*

CONTENTS—*Continued*

List of Figures

List of Tables

Abstract

Interlinear Glossed Text is widely used in linguistic studies. The following is an example of Scottish Gaelic Interlinear Glossed Text.

(i) Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

In a simple form of Interlinear Glossed Text, the first line is a sentence of the language of interest, the second line is a word-by-word translation, annotated with relevant grammatical information, and the third line is an English translation.

The Innovation of the current work is to incorporate the gloss information of Interlinear Glossed Text data into neural net machine translation systems.

Critically, if the Gaelic data and the gloss data are combined in a specific way as the training data, I term which as Parallel-Partial treatment, the performance of the systems is improved significantly. The Parallel-Partial treatment lets the machine learn four sets of mappings: 1.) from source sentence to target sentence, 2) from gloss lines to target sentences, 3) from gloss lines to source sentences, and 4) from source language words to gloss items.

Moreover, the boosting effect of the Parallel-Partial treatment is consistent across different languages and across neural net machine translation systems with different hyper-parameter settings.

How theoretical linguistics may work hand in hand with natural language processing, and how neural net machine learning may exploit linguistics are important questions. Pater (2017). The current work also exemplifies how theoretical linguistics may work hand in hand with natural language processing successfully, in addition to practically building better machine translation systems.

Improving Neural Net Machine Translation Systems with Linguistic Information

Yuan-Lu Chen, Ph.D.
The University of Arizona, 2018

Director: Mike Hammond

Interlinear Glossed Text is widely used in linguistic studies. The following is an example of Scottish Gaelic Interlinear Glossed Text.

(i) Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

In a simple form of Interlinear Glossed Text, the first line is a sentence of the language of interest, the second line is a word-by-word translation, annotated with relevant grammatical information, and the third line is an English translation.

The Innovation of the current work is to incorporate the gloss information of Interlinear Glossed Text data into neural net machine translation systems.

Critically, if the Gaelic data and the gloss data are combined in a specific way as the training data, I term which as Parallel-Partial treatment, the performance of the systems is improved significantly. The Parallel-Partial treatment lets the machine learn four sets of mappings: 1.) from source sentence to target sentence, 2) from gloss lines to target sentences, 3) from gloss lines to source sentences, and 4) from source language words to gloss items.

Moreover, the boosting effect of the Parallel-Partial treatment is consistent across different languages and across neural net machine translation systems with different hyper-parameter settings.

How theoretical linguistics may work hand in hand with natural language processing, and how neural net machine learning may exploit linguistics are important questions. Pater (2017). The current work also exemplifies how theoretical linguistics

may work hand in hand with natural language processing successfully, in addition to practically building better machine translation systems.

Chapter 1

Introduction

The fundamental argument that I am trying to make in the dissertation is as follows:

(1) Without linguistics, we only have ordinary natural language processing systems. With linguistics, we have extraordinary natural language processing systems.

Specifically, the curious question is whether or not interlinear glossed texts, an everyday tool for linguistic studies, can help machine translation. The following is an example of interlinear glossed text:

(2) Indonesian (Sneddon et al., 2012, p. 237)

Mereka di Jakarta sekarang. (*sentence of interest*)
they in Jakarta now (*gloss line: word-by-word gloss translation*)

'The are in Jakarta now.' (*English translation*)

The first line in an interlinear glossed text is the sentence of interest in its written form, the second line is a word by word morpheme by morpheme translation, and the third line is the corresponding English translation.

The glossing data has very interesting properties. First, it contains linguistics information. The glosses are not raw natural data, but are already processed by linguists based on the linguistic theory they adopt. Second, it is a type of big data, because interlinear glossed texts are so widely used in linguistics. Both natural language processing and linguistics are studying human languages. Gloss is the right 'lingua franca' for the two fields.

Also thanks to the advent of neural net sequence to sequence machine learning, which is a generic algorithm that can learn almost any sequence to sequence mapping,

the chance to successful incorporate the gloss line into machine translation is also better than in the past.

With high expectations on the gloss line information, I conducted a series of machine translation experiments. It is found that the gloss information is a very effective booster for neural net machine translation systems in all conditions.

The rest of the dissertation is organized as follows: chapter 2 discusses the nature of gloss lines, and argues that they are proper representations of meanings; chapter 3 provides a general overview of machine learning and machine translation; chapter 4, chapter 5 and chapter 6 are a series of neural net machine translation experiments; chapter 7 outlines potential future researches and concludes the dissertation.

Chapter 2

What are glosses? Why are they golden representations of meanings?

2.1 Introduction: What are Glosses?

Interlinear Glossed Text is widely used in linguistic studies. The following is an example of Interlinear Glossed Text.

(3) Indonesian (Sneddon et al., 2012, p. 237)

Mereka di Jakarta sekarang. (*sentence of interest*)
they in Jakarta now (*gloss line: word-by-word gloss translation*)

'The are in Jakarta now.' (*English translation*)

A chunk of an interlinear glossed text has three lines. The first line is the sentence of interest. The second line is the gloss line, which is a word-by-word translation of the first line. And the third line a free English translation of the first line.

The conventional way to show the word-by-word translation from the first line to the gloss line is to use vertical alignment. In (3), '*Mereka*' is glossed as '*they*', '*di*' is glossed as '*in*', '*Jakarta*' is glossed as '*Jakarta*', and '*sekarang*' is glossed as '*now*'. These pairs are vertically aligned.

The gloss line also provides morphological information. Consider the following example:

(4) French

aux chevaux
to.ART.PL horse.PL

'to the horses'

The morphemes of a single word are linked by a '.'. The French word 'aux' is actually a combination of three separate morphemes[1]: '*to*', '*ART*', and '*PL*' and *'chevaux'* is decomposed into *'horse'* and *'PL'*.

Bickel et al. (2008) compile a set of widely used conventions of IGT called the Leipzig Glossing Rules. Note that they are just guidelines of the formats of Interlinear Glossed Texts, so that Interlinear Glossed Texts can be more standardized.

The underlying intuition of Interlinear Glossed Text is that it provides an access to look into the subparts of a sentence. We may imagine the situation without the gloss line; then all we have is just the sentence and the English translation of that sentence. This will make it really hard to discuss the internal structure of the sentence. On the other hand, with the presence of the gloss line, with which each word is glossed and annotated, we then have a meta-representation in hand to discuss the grammatical properties of the sentence of interest.

An important note of the gloss line is that it is NOT raw linguistic data, and it is already processed. A linguist has already committed to some theory or some analysis on the sentence of interest when he or she transcribes the sentence into a gloss line, even if he or she tries to be as neutral as possible. As such, the question of what the gloss of a word is not trivial at all. Actually, sometimes a whole linguistic paper or thesis is to discuss and argue what the right gloss for a word is.

(5) Mandarin Chinese

Zhangsan **hen** gao
Zhangsan HEN tall

'Zhangsan is tall.'

For example, Grano (2008), Chen (2010), and Liu (2010) discuss the nature of the Mandarin Chinese word '*hen*' in the above example and what the right gloss should

[1] A morpheme is a smallest unit of meaning. For example, '*boys*' has two morphemes in it: '*boy*' and '*-s*', where '*-s*' is a plural marker. Sometimes, the morpheme boundary is not visible. For example '*went*' is composed of '*go*' and '*-ed*'.

be '*hen*'. In cases like this, how one glosses a word is not trivial at all, but determining what the gloss of word is requires a set of evidence and arguments.

2.2 The Golden Properties of Glosses

A system of meaning representations is decomposed of three components: a) meanings, b) representations, and c) a mapping between meanings and representations. The most ideal meaning representation system should be built with one-meaning-to-one-representation mappings; in other words, a meaning is mapped to one and only one representation. Natural languages fail to do so, given that synonyms and ambiguous words/phrases are ubiquitous in natural languages. On the other hand, gloss provides a mapping that is close to this ideal one-to-one mapping. Thus gloss should be a better representation in term of representing meanings.

Theoretically, the claim that gloss representation is closer to the ideal one-to-one mapping than natural language representation can be tested empirically. Let's imagine a set of special golden meta-linguistic semantic representations, which has the following property: each concept is mapped to one and only one representation and each representation is mapped to one and only one concept. With this imaginary golden semantic representation system, we may now compare Gaelic words and glosses. First, it is expected that each golden representation token will map to more natural language words than gloss items do.

(6) a. $golden_i \rightarrow \{Gaelic\_word_1, Gaelic\_word_2, \ldots\}_{golden_i}$

b. $golden_i \rightarrow \{gloss_1, gloss_2, \ldots\}_{golden_i}$

c. $|\{Gaelic\_word_1, Gaelic\_word_2, \ldots\}_{gold_i}| \geq |\{gloss_1, gloss_2, \ldots\}_{gold_i}|$

(6a) and (6b) represent a singe golden token may map to multiple Gaelic words and glosses respectively. If we compare the size of them, it is expected that the set of Gaelic words is bigger than that of glosses, meaning that Gaelic words are more likely

to be homographs than glosses are. Section 2.2.1 will provide concrete examples to exemplify this property of glosses.

For the other direction, we may determine which one, Gaelic words or glosses, is more likely to be ambiguous.

(7) a. $Gaelic\_word_i \rightarrow \{golden_1, golden_2, \ldots\}_{Gaelic\_word_i}$

b. $gloss_i \rightarrow \{golden_1, golden_2, \ldots\}_{gloss_i}$

c. $|\{golden_1, golden_2, \ldots\}_{Gaelic\_word_i}| \geq |\{golden_1, golden_2, \ldots\}_{gloss_i}|$

(7a) and (7b) show the mappings from a Gaelic word to different concepts and the mappings from a gloss to different concepts respectively. (7c) is the expectation that Gaelic words are more likely to be ambiguous than glosses are. Section 2.2.2 will report concrete examples to show that glosses are less likely to be ambiguous.

To run statistical experiments to confirm the truth of (6c) and (7c) is the way to empirically support the claim that glosses are closer to the golden representations than Gaelic words. However, in reality, this is an impossible experiment to conduct, because there are no such golden representation[2]. In spite of the impossibility of conducting statistical experiments, we may still use some examples to show the intuition that glosses are better representations than natural languages are. The following sections describe how glosses cluster up words with different forms but with the same meaning, and how glosses represent words with the same form but with different meanings with different representations.

2.2.1 Glosses Cluster Different Words with the Same Meanings (Synonyms) Into a Single representation

Gloss collapses words with different forms with the same meanings into a single representation. In natural languages, the morphology of a word (i.e. the form of a word)

[2]It would solve the puzzle of semantics if one should be able to build the set of special golden meta-linguistic semantic representations, and the mappings between the golden representations to natural languages.

may be sensitive to the phonological environments and changing into different forms. Consider the following indefinite article in the English examples:

(8) John ate **an** apple.
John eat.past **INDF_ART** apple

(9) John ate **a** banana.
John eat.past **INDF_ART** banana

In the above example, *an* and *a* have the identical meaning[3]. In English, the same concept is realized as two representations, *a* or *an*, while in the gloss representation the one concept is neatly represented as *INDF_ART* (indefinite article).

Critically, synonyms like the English *a* and *an* commonly occur in many other natural languages if not in all languages. The definite article in the language of interest, Scottish Gaelic, is another example to show the noisiness of natural language representations. Consider the definite article in the following Gaelic examples.

(10) tha mi a' sireadh **an** leabhair bhig ghuirm
be-PRES-IND 1S PROG searching-VN **ART** book-G small-G blue-G
'I am looking for the small blue book' (Lamb, 2001, p. 29)

(11) **am** fear mòr
ART man big
'a big man' (Lamb, 2001, p. 31)

(12) thuit **a'** chlach air cas mo mhnà
fall-PAST **ART** stone on foot 1S-POSS wife-G
'the stone fell on my wife's foot' (Lamb, 2001, p. 30)

(13) doras **na** sgoile(adh)
door-N **ART** school-G
'the door of the school' (Lamb, 2001, p. 29)

[3]Semantically, *an* and *a* are existential quantifiers, which declare that a member of a set exists in the world. In formal semantics, *an* and *a* may be defined as follows: $\exists\lambda P[P(x)]$. In the current example, *apple* and *banana* will instantiate P in the formula, and the meanings will be 'an apple exists' and 'a banana exists'. Kratzer and Heim (1998) would be a nice introduction for interested readers to see how linguists, specifically semanticians, define, decompose, and compose meanings of languages formally.

(14) a chuir air dòigh **nan** àiridhean a-muigh a rubh' Eubhal agus an oidhche seo
to put-INF on order **ART** sheilings out-LOC to point Eaval and ART night this
'the girls big house' (Lamb, 2001, p. 100)

(15) fèis **nam** bàrd
festival **ART** poet.PL.GEN
'festival of the poets' (Lamb, 2001, p. 107)

The definite article in Scottish Gaelic may be realized as the following forms: *an*, *am*, *a'*, *na*, *nan* or *nam*. The alternation is determined by the case, gender and number of noun phrase that it modifies, and additionally the phonological property of the word following it also changes the form of the definite article (Lamb, 2001). All these different realizations refer to the same concept, the definite article. Again, the gloss notation nicely clusters them together as *ART*.

In Mandarin Chinese, similar patterns are found. Consider classifiers in the following examples:

(16) Yani mai-le **{pi/*tou}** ma , Lulu mai-le **{*pi/tou}** zhu.
Yani buy-PRF CL/CL horse , Lulu buy-PRF CL/CL pig
'Yani bought a horse and Lulu bought a pig.' (Zhang, 2013, p. 136)

In Zhang (2013), the classifier like *pi* and *tou* is a type of *indivual classifier* which co-occurs with countable nouns, like *ma*, 'horse', and *zhu*, 'pig', and this type of classifier is the head of *UNIT Phrase*. *Pi* and *tou* actually have the same semantics and the syntactic function; however, they are realized in different forms, specifically the form of which has to agree with the noun following it (i.e. *pi* goes with *ma*; *tou* goes with *zhu*). Here the gloss, *CL*, unifies the two forms of the same meaning.

Gloss collapses synonyms in natural languages. Learning the general distribution of the article and all its different forms is a challenge for the MT system, but the glossing information should make this easier.

2.2.2 Glosses Distinguish Homographs' Different Meanings

In natural languages, there are cases when a single form denotes distinct concepts. Words with this property are termed as homographs. Consider the word *for* in following English examples:

(17) a. I intended **for** Jenny to be present.

b. **For** Jenny, I intended to be present. (Adger, 2003, p.306-307)

For in (17a) and (17b) has the same form but different meanings. Specifically, *for* in (17a) is a complementizer with its part of speech being *C*, and it heads the non-finite clause *Jenn to be present*; on the other hand *for* (17b) is a preposition, which takes a Determiner Phrase, *Jenny*, as its benefactive argument.

The Scottish Gaelic word *a'* in the following examples also has different meanings.

(18) tha mi **a'** sireadh an leabhair bhig ghuirm.
be-PRES-IND 1S **PROG** searching-VN ART book-G small-G blue-G
'I am looking for the small blue book.' (Lamb, 2001, p. 29)

(19) thuit **a'** chlach air cas mo mhnà.
fall-PAST **ART** stone on foot 1S-POSS wife-G
'the stone fell on my wife's foot.' (Lamb, 2001, p. 30)

Critically *a'* in (18) is a progressive aspect marker while the same form in (19) is a definite article. Again, the semantic difference is preserved in the gloss representations but not in natural language words. As such, the usefulness of glosses becomes very apparent.

2.2.3 Glosses are Sensitive to Hierarchical Structures in Natural Language Sentences

Before I introduce how gloss information is linked to hierarchical structures, it is necessary to emphasize the importance of hierarchical structures in natural languages.

In this section, I will first review some linguistic arguments for why and how semantics and syntax of languages[4] are all about hierarchical structures instead of linear word orders. Then I will link gloss to hierarchical structures.

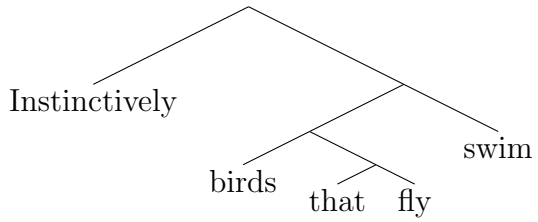
It is well-argued in linguistics that the syntax and semantics of natural languages are determined by hierarchical structures instead of linear orders of words, and essentially it is the sensitivity of hierarchical structures that distinguishes human natural languages from other animal communications (Berwick and Chomsky, 2015).

Semantics is determined by hierarchical structures instead of linear orders. Berwick and Chomsky (2015, p. 117) use the following simple example to demonstrate this property of natural languages:

(20) Instinctively birds that fly swim.

In the example above, *instinctively* is linearly closer to *fly* than *swim*; however, it unambiguously modifies *swim* instead of *fly*. The reason for this is the hierarchical structures (Berwick and Chomsky, 2015, p. 117):

(21)



In (21) it is shown that *fly* is more embedded than *swim*, and thus it is hierarchically further away from *instinctively*. So, *instinctively* can only modify *swim* instead of *fly*.

Syntax is also all about hierarchical structures. Consider the following sentence:

(22) a. Birds that can_1 fly can_2 swim.

b. *Can_1 birds that fly can_2 swim?

c. Can_2 birds that can_1 fly swim?

[4]When it turns to the sound aspect of languages, Phonetics is more about linear order, but Phonology is still sensitive to hierarchical structures just like syntax and semantics.

(22a) is a declarative sentence. To derive an interrogative sentence from it, the auxiliary needs to be moved; however, only can_2 can be moved but not can_1, even thought can_1 is linearly closer to the sentence initial position. Again, it is all because of the hierarchical structures. Can_2 is in the matrix clause while can_1 is in the embedded relative clause.

Glosses, on the other hand, are sensitive to the internal hierarchical structures or constituency of sentences. They provide more clues of the internal hierarchical structures or constituency of sentences than natural language words. Consider the following examples, modified from (17):

(23) For as *complementizer* (glossed as *complementizer*)

a. I intended **for** [Jenny] to be present.

b. I intended **for** [the girl] to be present.

c. I intended **for** [the little girl] to be present.

d. I intended **for** [the little girl who wants to eat some ice scream] to be present.

(24) For as *preposition* (glossed as *preposition*)

a. **For** [Jenny], I intended to be present.

b. **For** [the girl], I intended to be present.

c. **For** [the little girl], I intended to be present.

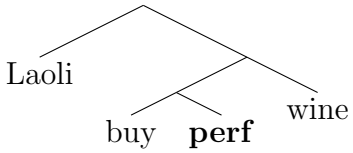
d. **For** [the little girl who wants to eat some ice scream], I intended to be present.

Linear length of the argument of *for* (i.e. the sequences in the square brackets) does not have any effect in determining what the gloss is, and instead it is the hierarchical structures that determine what the gloss is. Then the form of gloss hints to the internal structures of the sentence.

A even more dramatic example comes from Mandarin Chinese. A single sequences of words may have distinctive meanings because of different parses, and the difference of parses is marked by the differences of glosses. In the following examples, the sentence '*Lao3Li3 mai3 hao3 jiu3*'[5] may have two distinct meanings depending on the status of '*hao3*'.

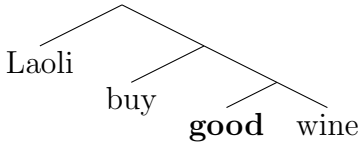
(25) a. Lao3Li3 mai3 hao3 jiu3
Laoli buy **Perf** wine
'Laoli bought a wine'

b.



(26) a. Lao3Li3 mai3 hao3 jiu3
Laoli buy **good** wine
'Laoli buys a good wine'

b.



In sentence (25), '*hao3*' goes with the verb '*mai3*'; as such '*hao3*' is interpreted as a Perfective marker and glossed as '*Perf*'; on the other hand, in (26), '*hao3*' goes with the noun '*jiu3*' and works as an adjective modifying '*jiu3*', and it is glossed as '*good*'.

With all the examples above, we have showed that gloss lines provide more clues of the the internal structures of the sentences are than natural language words do.

In the machine translation literature, it has been shown that the information of syntax and sentence parses can improve the quality of machine translation (Kiperwasser and Ballesteros, 2018) and increase the accuracy of extracting thematic role

[5]These specific examples are extensively discussed in Mandarin Chinese Tone Sandhi literature (e.g. Cheng (1973); Mei (1991); Shih (1997); Wang and Lin (2011)). Critically, the constituency plays a role in Mandarin Chinese Tone Sandhi.

relations in a sentence (Strubell et al., 2018). Given that glosses encode some sentence parse information, it is reasonable to hypothesize that glosses will improve machine translation.

2.3 Conclusion: What Is a Gloss Line and Why Do They Matter?

The gloss line is like a linguistic version of 'word embedding'. A natural language word is first converted to a gloss, which is readable for linguists. Also we may view a gloss line as an artificial sentence using the purified 'gloss words', a meaning representation with which one meaning is mapped to one and only one representation. Given the properties of gloss data, it can be a very useful data for machine translation. Moreover, gloss data is widely used in linguistics literature, so data is already out there and all we need to do is to clean the data. A loose end here is that, even if all the arguments should be sound, we still have no statistical evidence to show the usefulness of the gloss data. Chapter 4, 5, and 6 tie up this loose end, in which I will report machine translation experiments using gloss data.

Chapter 3

A General Introduction of Machine Learning and Machine Translation

Before I report neural net machine translation experiments in Chapter 4, 5 and 6, the current chapter serves as a general literature review and introduction of Machine Learning and Machine Translation. The current chapter is not meant to be a comprehensive summary of all the technical details of Machine Learning and Machine Translation, but instead it aims to provide high-level descriptions of the algorithms used in Machine Learning and Machine Translation.

3.1 What is Machine Learning?

This section provides a general, brief and non-technical introduction of Machine Learning.

3.1.1 What is a Function and What is a Model?

Machine learning is a specific algorithm that builds a function. Before we touch upon machine learning, we need to talk about what a function is. Here a function simply means a mapping from an object to another object; to put it in another way, a function takes an input and returns an output. A specifical property of that function is that an input is mapped to one and only one input. A general form of function is given below:

(27) $f : X \rightarrow Y$

or

$y = f(x)$

For example, the following is a simple function that takes a number as input and returns the double of that number:

(28) a. $f(x) = 2x$

b. $f : x \rightarrow 2x$

e.g.

i. $f : 1 \rightarrow 2$

ii. $f : 2 \rightarrow 4$

iii. $f : 3 \rightarrow 6$

iv. ...

Grammaticality judgment of a sentence can be viewed as a function, where the input is sentence and the output is either grammatical or ungrammatical.

(29) a. $English\ Grammar : sentence \rightarrow Grammaticality$

e.g.

i. $English\ Grammar : John\ loves\ Mary \rightarrow grammatical$

ii. $English\ Grammar : loves\ John\ Mary \rightarrow ungrammatical$

iii. ...

Translation can be viewed as function, too, where the input is a sentence in the source language and the output is a sentence in the target language that conveys the same meaning of the source sentence[1]. For example, a function that translates Scottish Gaelic into English looks like:

[1]Strictly speaking, translation for human translators is more like relation instead of function, because a source may be translated to multiple target sentences. In this manner, we may have one-to-many mappings. However, given that machine translation systems return one and only one output for one input, machine translation can still be viewed as a function.

(30) a. $Translation_{GaelicToEnglish} : Gaelic\ sentence \rightarrow English\ sentence$

e.g.

i. $Translation_{GaelicToEnglish}$:

thuit a′ chlach air cas mo mhn‘a

→

the stone fell on my wife′s foot

ii. $Translation_{GaelicToEnglish}$:

tha mi a′ sireadh an leabhair bhig ghuirm

→

I am looking for the small blue book

iii. . . .

Except for mathematical functions like (28), in most of the cases, we have no direct access to a function. In other words, the function is assumed to be unknown. To simulate the behaviors of an unknown function, we build a model with the hope that the model can do the same mapping as the target function. Schematically, the relation between a function and a model that simulates that function is depicted below:

(31) a. $f : X \rightarrow Y$ (unknown target function)

b. $g : X \rightarrow \hat{Y}$ (model)

where $Y \approx \hat{Y}$

(31a) is the unknown target function (conventionally it is represented by f in the literature), and (31b) is the model that we build with the hope that it does the same mapping as f (conventionally a model is represented by g in the literature).

In a nutshell, a model is a handmade mapping that simulates an unknown function.

3.1.2 Two Different Paradigms of Building a Model

Now the critical problem is how we build a models to approximate the target function. With the unknown target function being the same f, there can be two distinctive approaches of building the model: Human-Reasoning Approach and Machine-Learning Approach.

(32) Human-Reasoning Model: A set of rules manually written by a human that attempts to approximate f based on his or her own knowledge and experiences.

The Human-Reasoning approach relies on an individual's knowledge and is task-specific. Taking the English grammaticality judgment task as an example, this approach means that some expert in English grammar will come up a set of rules or some algorithms that can determine whether a sequence of strings is a grammatical English sentences. The rules and algorithms are based on this expert's knowledge and they are English specific, and are not applicable to other languages.

Another approach to build a model is using Machine Learning. Instead of building a model from our own reasoning, we may collect all the available examples and let a machine learn from the examples. This is the Machine Learning approach. Machine Learning can be defined as follows:

(33) Definitions of Machine Learning:

a. Machine learning is based on algorithms that can learn from data without relying on rules-based programming (Pyle and San Jose, 2015).

b. Machine learning algorithms can figure out how to perform important tasks by generalizing from examples (Domingos, 2012).

c. The field of Machine Learning seeks to answer the question 'How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?' (Mitchell, 2006).

For the Machine Learning approach, the most critical factors of how a model is built are: 1.) the generic learning algorithms, and 2.) the input data from which the generic learning algorithms learns to set up the parameters of the derived model.

Taking the English grammaticality judgment task as example again, the machine learning approach will first define some generic learning algorithm, and then feed the algorithm with labeled sentences, where the label is either 'grammatical English sentence' or 'ungrammatical English sentence'. Note that the learning algorithm is generic. This means that it is universally applicable. For example, instead of feeding the learning algorithm with sentences labeled with English grammaticality, we may feed the learning algorithm with sentences labeled with Scottish Gaelic grammaticality, and it will able to derive a model that can tell whether or not a sentence is a grammatical Scottish Gaelic sentence. Thus, the learning algorithms in machine learning is generic[2].

3.2 Statistical Machine Translation

Statistical Machine Translation uses statistical Machine Learning approaches to build machine translation systems (Brown et al., 1988, 1990, 1993; Koehn, 2009; Koehn et al., 2007). The goal is to build a model that simulates the unknown language mapping function. The training data is parallel corpus, which is a collection of pairs of a sentence in one language and the translation of the sentence in another language.

The fundamental translation equation of Statistical Machine Translation is shown as follows:

[2] In the genericness and universality of the learning algorithms, there is a reminiscence of Universal Grammar (Chomsky, 2007). Universal Grammar is a type of machine learning. Instead of teaching a set of rules, children are given a set of grammatical sentences and then they are able to learn grammar, and produce grammatical sentences.

Chomsky (2005) describes three factors in language design: 1) Universal Grammar, 2) Experience, and 3) other cognitive mechanisms/limitations. This is actually a precise description of how machine learning works. Universal Grammar is the generic learning algorithm. Experience is the training data. Cognitive mechanisms are all the other hardware specification of a machine (i.e. a human brain). So, when the Experience is language X, a child acquires language X.

$$Pr(T|S) = \frac{Pr(T)Pr(S|T)}{Pr(S)} \tag{3.2.1}$$

In this equation, we try to translate a sentence S in the source language to a sentence T in the target language. The left part of the equation represents the probability of T given S. The right side of the equation is just the result of the application of Bayes' theorem on the conditional probability. The advantage of the conversion is that now $Pr(T)$ (the probability of sentence T) can be incorporated into the model. This information can be retrieved by building a language model of the target language, which can estimate the probability of a sequence of strings. $Pr(S|T)$ is the translation model, which measures the probability of S given T. Given this equation, translating a given sentence S simply means to find a sentence $\hat{T}$ that maximize $Pr(T|S)$.

$$\hat{T} = \underset{T}{\operatorname{argmax}}\, Pr(T|S) = \underset{T}{\operatorname{argmax}}\, \frac{Pr(T)Pr(S|T)}{Pr(S)} = \underset{T}{\operatorname{argmax}}\, Pr(T)Pr(S|T) \tag{3.2.2}$$

Note that the denominator, $Pr(S)$, of $\frac{Pr(T)Pr(S|T)}{Pr(S)}$ in the equation can be ignored because it is a constant when we compare all the possible $\frac{Pr(T)Pr(S|T)}{Pr(S)}$. Even though we are using probability here, Statistical machine translation still has the function structure that maps one input to an output with the input being the source sentence and the output being the target sentence.

$$f_{Statistical\_machine\_translation}(S) = \underset{T}{\operatorname{argmax}}\, Pr(T)Pr(S|T) \tag{3.2.3}$$

Now a Statistical Machine Translation system is decomposed into two sub-models: the language model of the target and the translation model. To build the language model, the common practice is to train a N-gram model given a corpus. To build the translation model, we will need to use parallel corpus. To understand the nature of Statistical Machine Translation, we need to look into what information is extracted

from the parallel corpus. Critically, the probability of words in the source language aligned with words in the target language is the heart of the translation model of Statistical Machine Translation.

Given that Statistical Machine Translation builds the translation model by using the information of string alignments, Statistical Machine Translation may be viewed as a complicated algorithm of string manipulations and string alignments. Importantly, this means that Statistical Machine Translation does not touch upon meaning, which limits the development and performance of Statistical Machine Translation.

As such, Statistical Machine Translation is doomed to be outperformed by other deeper algorithms that touch upon the domains of meaning. Specifically, 2016 is the year when neural net machine translation started to outperform Statistical Machine Translation. In the Conference on Machine Translation 2016, a neural machine translation system outperformed almost every statistical machine translation systems. And in the following year, the Conference on Machine Translation 2017 was dominated by neural systems. Almost all the submitted papers and models were using neural net machine translation systems.

In the next section, I will introduce the basics of artificial neural nets and the overview of neural net machine translation.

3.3 Neural Net Machine Translation

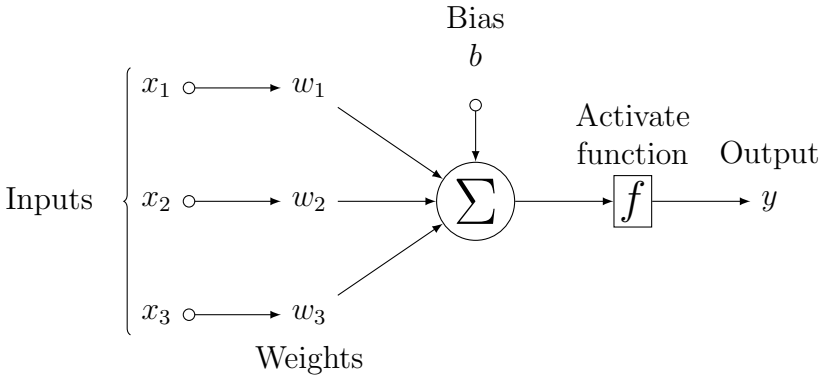
Neural Net Machine Translation is a type of machine learning. The training data (i.e. the example that machine learn from) is also parallel corpus, just like statistical machine translation. In what follows, I will first provide a overview of artificial neural networks, and then I will explain how artificial neural networks translate sentences in a non-technical way.

3.3.1 What Is an Artificial Neural Network?

An artificial neural network is a simple but powerful computation algorithm. It can be viewed as a mathematical function: given an number or a vector of numbers as input, it will return one and only numerical representation.

Historically, the basic structure of artificial neural networks was long completed more than 60 years ago, as Rosenblatt (1958) described the design and structure of a perceptron, which is the simplest artificial neural network. The following figure is a depiction of a perceptron:

FIGURE 3.1. An Example of a Perceptron



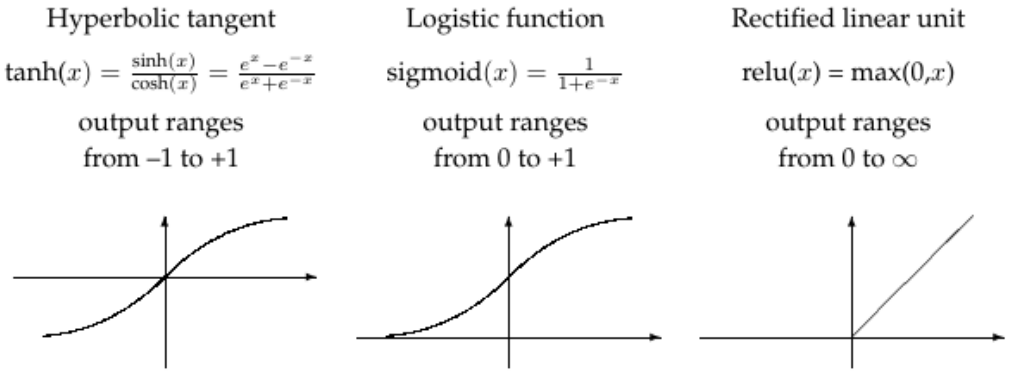
In a perceptron, we have input nodes, which carry numerical values. These input nodes are connected to another node via weighted connection arrows. The newly linked node will first have a raw numerical value. This raw value is simply the summation of the product of the numerical value of the input node and the weight of the linking arrow plus the bias value. In the example above, we will have:

$$\text{(34)} \quad \textit{Raw numerical value} = x_1w_1 + x_2w_2 + x_3w_3 + b$$

Then, an activation function will further normalize the raw numerical value by mapping it to another value.

$$\text{(35)} \quad y = f(\textit{Raw numerical value})$$

FIGURE 3.2. Activation functions



Here y is the final output, which is another numerical value.

The most commonly used activation functions in artificial neural nets are Hyperbolic tangent, Logistic, and ReLU.

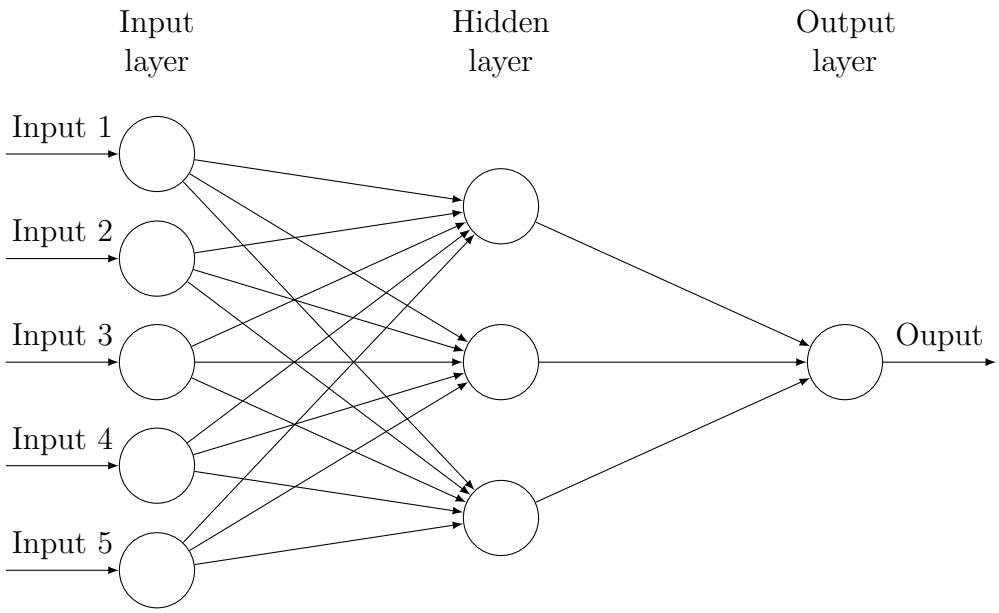
A perceptron is the basic building blocks of neural network computation. Note that the output of a perceptron can be the input of another perceptron. In this manner, the perceptron can be interconnected to one and other, yielding complex networks.

The following figure depicts a neural network with an additional layer between the input and output nodes:

This middle layer is called hidden layer. However, they are not really hidden; actually all the numerical values of each neuron in the hidden layer are all visible, but it is really hard to come up with a sensible interpretation on these numbers. When there are more than one hidden layers, the neural net is called deep neural net. However, the building blocks are still a simple perceptron. So, actually neural networks are simple and elegant computation operation[3]; however, they extremely powerful. They are Turing-Complete machines (Siegelmann and Sontag, 1991; Graves et al., 2014). Siegelmann (2003, 2012) even argues that neural networks are computations beyond

[3]It seems to me that linking the perceptron is just like the simple Merge operation in the Minimalist Program.

FIGURE 3.3. A neural network with a hidden layer



†For clarity, I don't show the bias nodes here.

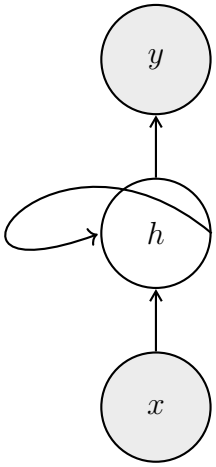
the Turing limit.

3.3.2 Recurrent Neural Network

Before we start to describe how a neural machine accomplishes the translation task, it is necessary to introduce a specific type of neural networks: Recurrent neural networks.

Recall that the output of a perceptron can be the input of another perceptron. A interesting manipulation is to let the perceptron connect back to itself. With this manipulation, we have the Recurrent Neural Networks. As shown in the following figures, the hidden node is connected back to itself at each time step. As such, at each time step, there are two input sources of the hidden node: 1) the information from the past hidden node, and 2) the information from a new input node.

FIGURE 3.4. A recurrent neural network with a hidden layer. The hidden neuron (h) is connected to itself.



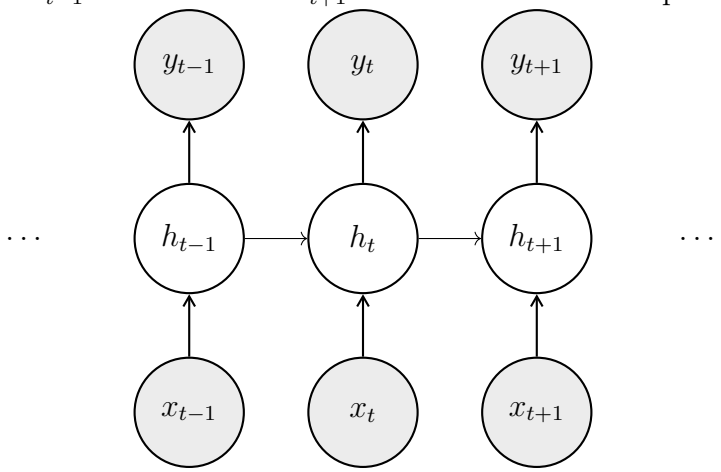
Is this manipulation useful? And if yes, why? It turns out Recurrent Neural Networks are really useful when we are interested in modeling a sequencing phenomenon and we believe that a fraction of the sequence will have an effect on the next coming fraction. The looping configuration opens the door to the history of the hidden neuron, and leads it to its future.

Recurrent neural networks are widely used. Critically, the main theme of the dissertation, Neural Net Machine Translation, is also implemented with it. The coming section provides an overview of how neural networks translate languages.

3.4 Introduction to Neural Net Machine Translation

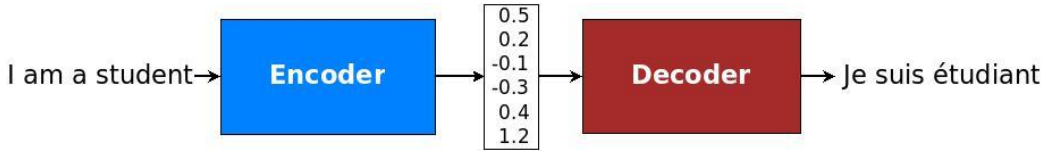
My 3-year old daughter, Trina, is growing up in Tucson, Arizona, The languages that she encounters are English, Spanish and Mandarin Chinese, and thus she picks up English and Mandarin, and some Spanish. She is able to count the quantity of objects in the three languages. When I told her **in Mandarin** that 'we can bring two candies for Alice' on the way to her preschool, and she was able to tell her best friend, Alice, in her preschool that she will give her one candy **in English** (not two because Trina has to save one candy for herself).

FIGURE 3.5. A recurrent neural network with a hidden layer (unfolded depiction). This figure depicts exactly the same structure in figure 3.4, the hidden neuron h_t is linked to its past h_{t-1} and its future h_{t+1}. Here t labels time step.



It is certain that a multilingual person can access the meaning of a sequence of strings in a language when he or she hears it, and then produces another sequence of strings in another different language with the same meaning. Given that a human brain is a neural network machine and artificial neural networks is similar to human neural networks, it fits our intuition that artificial neural networks can translate languages, of course. Actually, this is roughly how a neural net machine translation is implemented. Consider the following figure from Luong et al. (2017).

FIGURE 3.6. Encoder-decoder architecture - example of a general approach for NMT. An encoder converts a source sentence into a "meaning" vector which is passed through a decoder to produce a translation. (figure from Luong et al. (2017))



The figure outlines the heuristic of how neural networks computation may translate languages. When the neural net is fed with a source sentence, "I am a student", it will convert the sentence into a meaning representation or a thought vector. This is the

encoding process. Based on this meaning representation, the machine then derives another sequence of words in the target language. This is the decoding process. Now we may take a close look of how the encoding and decoding process works.

3.4.1 Word Embedding

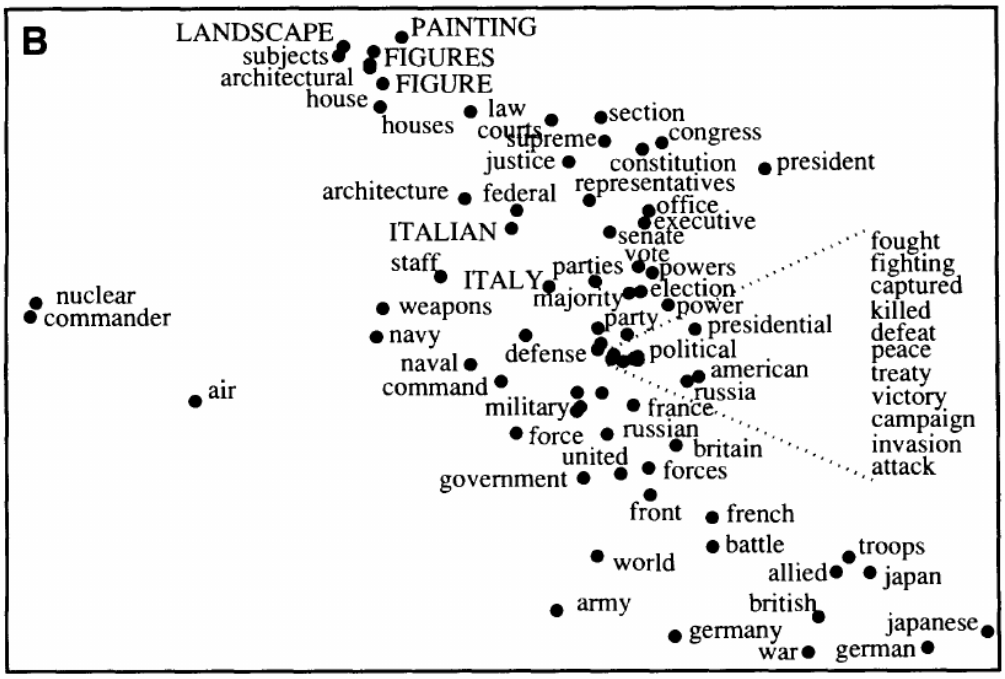
The neural machine can only process numeral representations, so the first step is to convert words into numerical representations. Specifically each word will be represented as a vector. The central idea is to map each word onto a location in "a meaning space", and each vector is the geographical coordinates of the word. This process is called word embedding. The ideal word embedding system would have the properties that words with similar meanings will be neighbors in the meaning space. The most commonly used technique of building word embeddings is proposed in Mikolov et al. (2013a,b). Critically, this word embedding implementation follows the distributional semantics philosophy: the meaning of a word is defined by its distribution in a corpus. So, to build a word embedding model all we need is a corpus. The training data for machine translation is a parallel corpus, which gives us two corpora: the source language corpus and target language corpus. So, the training data is sufficient for building word embedding models[4]. The following figure depicts how it is like when the word embeddings are mapped to a 2D space.

3.4.2 Encoding and Decoding

A simple preprocessing on the data is to add tags that marks the boundaries of the sentences. Each word in the source sentence is sent to recurrent network machine incrementally. Note the looping configuration of recurrent neural networks makes it possible for the machine to accumulate the information of all the words. When the

[4]It is also a common practice to build the word embedding separately using a bigger corpus when the parallel corpus is too small.

FIGURE 3.7. Word Embeddings (figure from Roweis and Saul (2000))



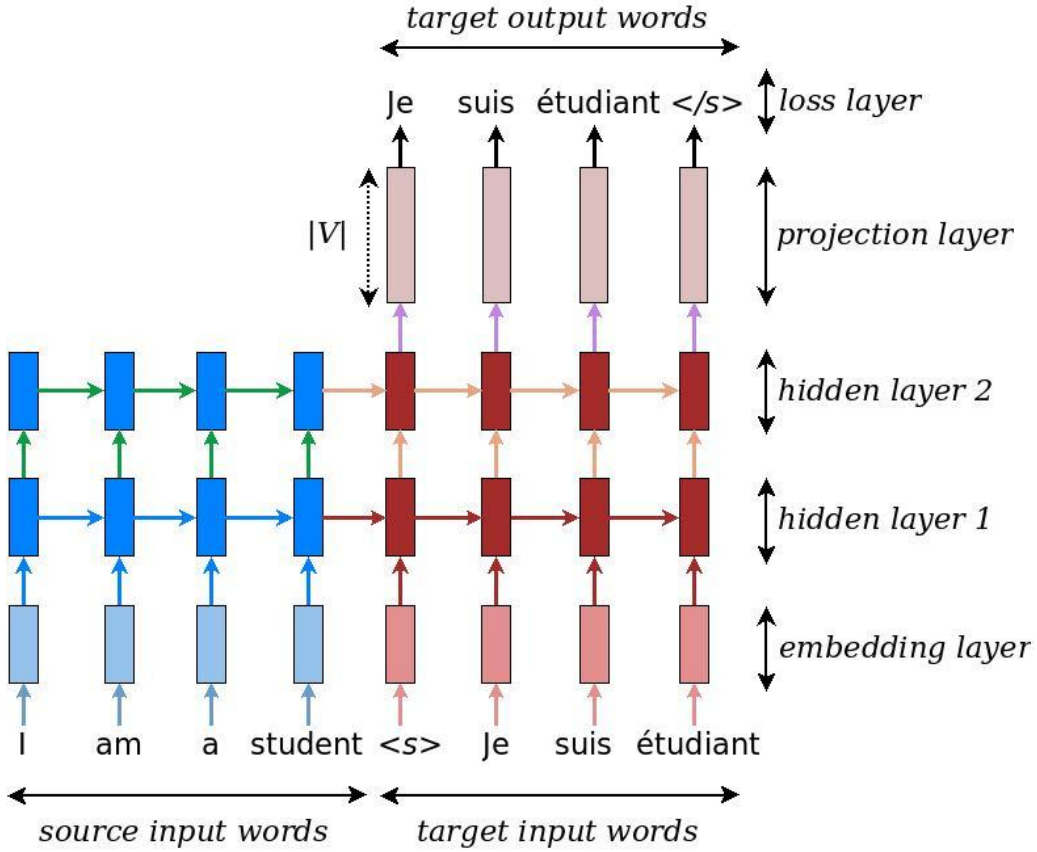
tag that marks the end of the source sentence is fed into the machine, the encoding process is completed, and the value of the hidden neurons at this time is the meaning representation of the source sentence.

Given the meaning representation, now the machine starts to decode. At each time step, it emits a word in the target language. The emitted word in the target language is copied and served as input in the next time step of decoding. So that the output sentence can have the context information of the previous word; this trick ensures the fluency of the output sentence.

Consider the following figure from Luong et al. (2017) for a concrete example of how the encoding-decoding process works.

In the current example, we are trying to translate a source sentence "I am a student" into a target sentence "Je suis étudiant". Here, "<s>" marks the end of the source sentences and the starting point of the decoding process. So, at the time step of "<s>", the vector in the hidden neurons is the meaning representation. In the

FIGURE 3.8. Neural machine translation (figure from Luong et al. (2017)).



decoding process, the first emitted word is "Je". Note that "Je" is sent back as the input in the next time step, when "suis" is emitted. Now we may pay attention to what the given information is when the machine is emitting "suis". Specifically, the given information is: 1) thought vectors plus "je" in the hidden layers, and 2) "je" from the input node. When the machine emitted the tag, "</s>", the decoding process is completed.

This section covers the basic structure of a neural machine translation system proposed in Cho et al. (2014a,b).

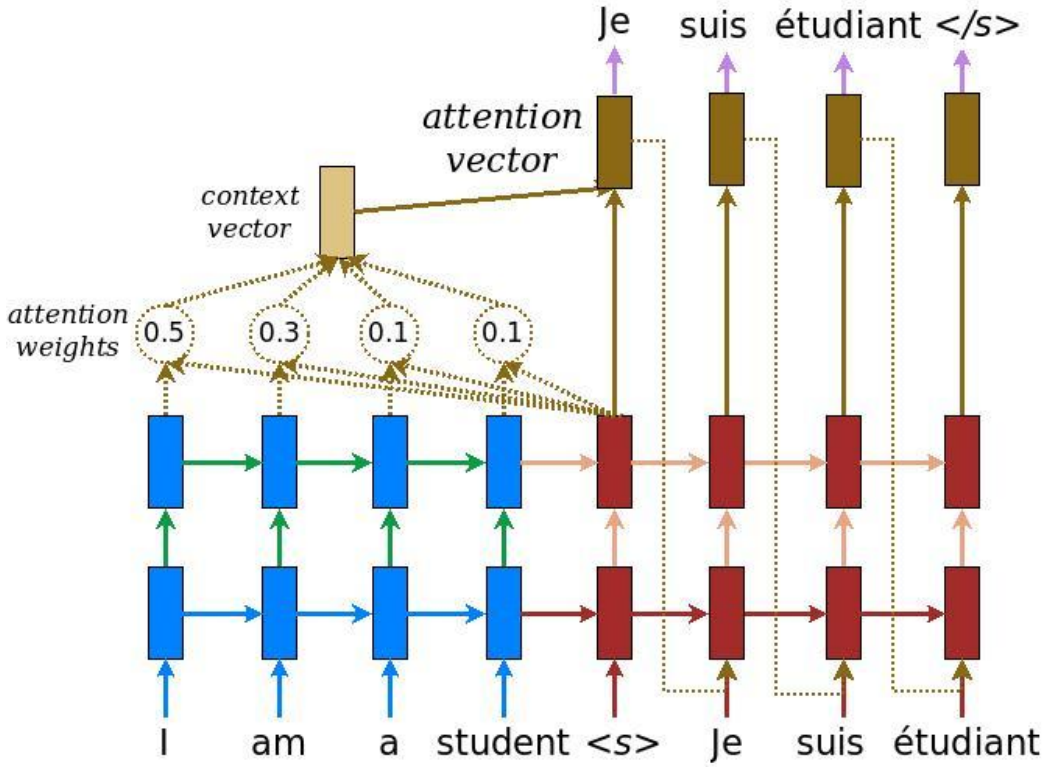
3.4.3 Adding Attention Mechanism

The meaning vector is the accumulated information of all the bits in the source sentence. The decoding process needs to sort out all the bits from the mixed information in the meaning vector. Actually, this is a lot to ask. The task is like to figure out what ingredients are from a pot of hot and sour soup by tasting the soup. To facilitate the sorting task of the decoding processing, Bahdanau et al. (2014) and Luong et al. (2014) propose methods to direct the decoder's attention to the relevant bits in the source sentence. Take the soup analogy again. With the attention mechanism, to guess what the ingredient is, the given information is now the taste of vinegar, in addition to the taste of the mixed-up soup in this time step. So that the chance of getting the right answer (i.e. vinegar) is much higher. And in the next time step, in addition to the taste of the mixed-up soup, one is also given the taste of pepper, and so on.

The following figure from Luong et al. (2017) nicely depicts how the attention mechanism works.

In the figure, the task is to translate a source sentence "I am a student" into a target sentence "Je suis étudiant". The encoding process is the same, but in the decoding process additional information is also provide. In the current example, when the decoder emits the word "je", it is given the meaning vector, and additionally the attention mechanism also gives the decoder 50% of the information of "I", 30% of the information of "am", 10% of the information of "a", and 10% of the information of "student". In this manner, the machine's attention is directed to "I" when the supposed emitted word is "Je". The attention mechanism is very similar to alignment; in the current example, "Je" is 50% aligned to "I", 30% aligned to "am", 10% aligned to "a", and 10% aligned to "student".

FIGURE 3.9. Attention mechanism - example of an attention-based NMT system as described in Luong et al. (2014) (figure from Luong et al. (2017))



3.4.4 Summary of Neural Net Machine Translation: interlingua plus string alignment

The state-of-the-art neural network machine translation is the combination of the encoding-decoding algorithm and the attention algorithm. The meaning vector in the encoding-decoding algorithm actually can be viewed as the holy grail in translation, the interlingua representation. The attention algorithm is a reminiscence of the string alignment approach used in statistical machine translation. So, neural net machine translation is actually a combination of interlingua translation approach and string alignment approach.

3.5 A Quick Historical Overview of Machine Translation and Conclusion

Machine Translation is to automate the process of translating one natural language to another using computers. The first peak of the developments of Machine Translation started in the 1940s but terminated in 1966. During this period of time, with the advent of the first computers, researchers held high expectations in Machine Translation and lots of resources are shifted into this area. However, in 1966 a report of the Automatic Language Processing Advisory Committee (Pierce and Carroll, 1966) terminated this 'machine translation rush' as it revealed that too many funds and resources were shifted to Machine Translation without yielding proportional scientific developments. In 1990s, with the advent of Statistical Machine Translation approach developed at IBM, the Machine Translation came back as one of popular scientific areas. Now with the advent of the technique of Artificial Neural Net Machine Learning and other natural language processing techniques, Machine Translation is one of the most dynamic research areas.

Chapter 4

Building Translation Systems using Interlinear Glossed Text: First Attempt

4.1 Introduction

The Innovation is to incorporate the gloss information of Interlinear Glossed Text data into machine translation.

In supervised machine learning models, two factors effect the performance of the trained systems (Kotsiantis et al., 2007): a.) the quality of the training data and b.) the choice of the features. The properties of the gloss data as described in chapter 2 make it a better training data than natural language data (Scottish Gaelic in the current case) for the following reasons. First, glosses are more purified than natural language words. The most ideal meaning representation system should be built with one-meaning-to-one-representation mappings; in other words, a meaning is mapped to one and only one representation. Natural languages fail to do so, given that synonyms and ambiguous words/phrases are ubiquitous in natural languages. Glosses provide this one-to-one mapping. Second, the gloss data provides hierarchical (non-linear) syntactic parsing information. To determine what the gloss of a word is, linguists have to look for hierarchical (non-linear) context information. See chapter 2 for the discussion on the golden properties of glosses.

Therefore, theoretically incorporation of the gloss data should improve the translation systems. Specifically, I propose the following hypothesis:

(36) **Gloss-help hypothesis: the translation systems trained with the gloss data incorporated should outperform the systems trained with only Gaelic and English sentences pairs (i.e. without gloss data).**

The hypothesis can have two versions, strong and weak:

a. Strong version: Gloss may replace the source natural language totally, and the system outperforms the system trained with source natural language to target language sentence pairs (i.e. the baseline systems).

b. Weak version: Gloss only increases the performance of the baseline systems, but cannot replace the source language.

The experiments in the current chapter will reveal that replacing Gaelic words with glosses doesn't boost up the performance of the translation systems. Thus, the strong version (replacing-Gaelic-with gloss) of the Gloss-help hypothesis is not empirically supported.

This chapter describes the experiments conducted to test the strong version of the Gloss-help hypothesis. The rest of the chapter is organized as follows: Section 4.2 describes related works in the literature, Section 4.3 describes the constant parameter settings across all the experiments and the corpus used in the experiments, Section 4.4 tests the hypothesis in (36a), Section 4.5 discusses the results and conclude this chapter.

4.2 Related Work

Attempts to improve machine translation systems by incorporating explicit linguistic information are reported in the literature. Syntax information is known to be effective in improving statistical machine translation (SMT). The efforts of using syntax information even derive a special type of SMT, termed as syntax-based SMT (Williams et al., 2016). The same trend is also found in neural net machine translation. For example, Sennrich and Haddow (2016) exploit the information of lemmas, part of speech tags, morphology of words, and dependency parses of sentences to improve MT systems. Nadejde et al. (2017) incorporate the Categorial grammar parse tags of the target sequences.

4.3 Technical Settings of the Machine Translation Experiments and Experimental Data of Scottish Gaelic Interlinear Glossed Text Corpus

4.3.1 Technical Settings

The experiments are conducted by using OpenNMT (Klein et al., 2017), which implements the state-of-the-art neural net machine translation algorithms (Cho et al., 2014a,b; Bahdanau et al., 2014). The following default hyper-parameter settings of OpenNMT[1] are used across all models so that the only independent variable is the type of the training data:

- Word vector size: 500

  In neural net machine translation, a word is represented as a vector. This hyper-parameter means that we are going to use vectors with 500 dimensions to represent words.

- Type of recurrent cell: Long Short Term Memory

  Long Short Term Memory recurrent neural net is a type of neural net that is suitable for sequence to sequence tasks.

- Number of recurrent layers of the encoder and decoder: 2

  This hyper-parameter specifies that we are going to have two recurrent layers of the encoder and decoder.

- Number of epochs: 13

  The training process of a neural net machine translation systems is done epoch by epoch. Each epoch is an iteration of training. Here 13 means that we are going to have 13 iterations of training and thus have 13 epochs.

[1] See their documentation for the complete default hyper-parameter settings: http://opennmt.net/OpenNMT-py/.

- Size of mini-batches: 64

  Training a neural net is to let the weights of the connections between the neurons fit the training samples. Theoretically, we may ask the neural net to adjust the weights according all the samples all together at one time. However, in practice, this is not memory efficient, and will cause errors in the process of optimizing the weight parameters. So, instead, the samples are split into smaller mini-batches, and the neural net just updates its weights to fit the samples in a mini-batch at one time. This hyper-parameter specifies the size of a mini-batch. Actually finding the right mini-batch size is not a trivial but an important question in Deep Learning. See Keskar et al. (2016) and Smith et al. (2017) for the experiments and discussions on the effects of the size of mini-batches.

The settings of the hyper-parameters do have effects on the performances of the trained models. A common practice to find the optimal settings of the hyper-parameters is to hold out a subset of the training dataset as the developing dataset, then test the models on the developing data to see what settings are optimal, then merge the developing dataset and training dataset as a new training set, and then train on this new training set using the found optimal hyper-parameters.

However, given that finding the optimal settings of the hyper-parameters is not relevant to our research and causes unnecessary complications, the process of optimizing the settings of the hyper-parameters is not implemented, and I simply adopt OpenNMT's default settings. The employed settings of the hyper-parameters should be viewed as arbitrarily chosen, and there are room to tune the models for better performance. Critically, these settings are viewed as constants, so that we can focus on the effects of different treatments on the source sequences in the translation experiments. We will leave the question of what hyper-parameters are optimal for our data for future research.

4.3.2 Experimental Data: a corpus of Scottish Gaelic Interlinear Glossed Texts

We use the same Scottish Gaelic Interlinear Glossed Text corpus (Chen et al., 2018) for all the experiments in chapter 4 and chapter 5. This corpus has 8,367 Gaelic sentences, and in term of words, it has 52,778 Gaelic words/glosses. The data of the corpus is from two different sources: linguistics fieldwork and data elicitation.

4.4 Gloss Representation Solely Does NOT Outperform Gaelic Sentences

This section tests the strong version of Gloss-help hypothesis in (36a). Given the assumption that gloss may be better than any natural language in terms of representing meanings, it is expected that for neural net machine translation systems it is easier to learn how to translate from the glosses of Scottish Gaelic to English than to learn how to translate from Scottish Gaelic to English. However, the results show that there is no significant difference between the two types of data (i.e. GLOSS → English and Gaelic → English).

4.4.1 Procedure of the Experiments

I use repeated random sub-sampling validation to compare the performances of the two types of models.

Totally we have 8,388 indexed 3-tuples of a Gaelic sentence, a gloss line and an English translation. Each line in the interlinear glossed text example below is an argument of a 3-tuple sample.

(37) Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

The 3-tuple representation of the above example is:

(38) <“Tha a athair nas sine na a mhàthair”, “be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother”, “His father is older than his mother”>

First, the samples (i.e. the 3-tuples) are randomly split into three datasets: training set (N=6,388), validation set (N=1,000), and test set (N=1,000)[2].

(39) Definitions of datasets:

Let:

a. $\text{Index}_{\text{Train}}$, $\text{Index}_{\text{Validation}}$, and $\text{Index}_{\text{Test}}$ be sets of random indexes from 0 to 8,387.

b. $\text{Index}_{\text{Train}} \cap \text{Index}_{\text{Validation}} \cap \text{Index}_{\text{Test}} = \emptyset$

c. $|\text{Index}_{\text{Train}}| = 6{,}388$; $|\text{Index}_{\text{Validation}}| = 1{,}000$; $|\text{Index}_{\text{Test}}| = 1{,}000$.

The step above just randomly splits the indexes of the 3-tuples into three distinct sets: $\text{Index}_{\text{Train}}$, $\text{Index}_{\text{Validation}}$, and $\text{Index}_{\text{Test}}$. Based on the indexes, we generate the sets of samples. For each index, the 3-tuple is split into two pairs: <gloss, English>, <Gaelic, English>, so that later we can compare the different effects of gloss lines and Gaelic sentences. For each pair, the first item is the source sequence, and the second item is the target sequence. The systems learn how to map the source sequence to the target sequence.

(40) Gloss to English

a. $\text{GLOSStoEN}_{\text{Train}} = \{< gloss_i, En_i > |\ i \in Index_{\text{Train}}\}$

b. $\text{GLOSStoEN}_{\text{Validation}} = \{< gloss_i, En_i > |\ i \in Index_{\text{Validation}}\}$

c. $\text{GLOSStoEN}_{\text{Test}} = \{< gloss_i, En_i > |\ i \in Index_{\text{Test}}\}$

[2]Here the random sampling process is achieved by using the `random.sample(population, k)` function in the standard library of python.

d. Example: <"be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother", "His father is older than his mother.">

(41) Gaelic to English

a. $\text{GDtoEN}_{\text{Train}} = \{< GD_i, En_i > |\ i \in Index_{\text{Train}}\}$

b. $\text{GDtoEN}_{\text{Validation}} = \{< GD_i, En_i > |\ i \in Index_{\text{Validation}}\}$

c. $\text{GDtoEN}_{\text{Test}} = \{< GD_i, En_i > |\ i \in Index_{\text{Test}}\}$

d. Example: <"Tha a athair nas sine na a mhàthair.", "His father is older than his mother.">

The models are trained with the training set and validation set (i.e. the model learns how to map the source sequence to the target sequence). Both training set and validation set are known information for the models[3]. Specifically, the neural net system learns how to map gloss lines to English sentences from samples in (40a) and (40b), and another neural net system learns how to map Gaelic sentences to English sentences from samples in (41a) and (41b).

(42) Models:

a. $\text{Model}_{\text{GLOSStoEN}}$ = Model trained with $\text{GLOSStoEN}_{\text{Train}}$ in (40a) and $\text{GLOSStoEN}_{\text{Validation}}$ in (40b)

b. $\text{Model}_{\text{GDtoEN}}$ = Model trained with $\text{GDtoEN}_{\text{Train}}$ in (41a) and $\text{GDtoEN}_{\text{Validation}}$ in (41b)

The two trained models (gloss-to-English and Gaelic-to-English) then take the right source sequences of the test sets (i.e. glossing lines and Gaelic sentences for $\text{Model}_{\text{GLOSStoEN}}$

[3]Technically speaking, the validation set is part of the training data in terms of machine learning. The presence of the validation set is a special requirement of neural net machine learning, which uses the validation set to evaluate the convergence of the training.

and $\text{Model}_{\text{GDoEN}}$ respectively) as inputs and then generate the predicted target sequences (i.e. English sentences).

(43) Predictions:

a. $\text{Predictions}_{\text{GLOSStoEN}}$ = A list of English sequences that $\text{Model}_{\text{GLOSStoEN}}$ maps to from the gloss sequences in (40c)

b. $\text{Predictions}_{\text{GDtoEN}}$ = A list of English sequences that $\text{Model}_{\text{GDtoEN}}$ maps to from the Gaelic sentences in (41c)

To evaluate the model, the predicted target sequences are checked against the target sequences of the test set (i.e. the gold standard/human-translated English sentences). Specifically, the BLEU (bilingual evaluation understudy)[4] score metric (Papineni et al., 2002) of each prediction is calculated using the **multi-bleu.perl**[5] script, a public implementation of Moses (Koehn et al., 2007).

The BLEU assumes that a sentence is a bag of n-grams (n is from 1 to 4). It measures how different the two bags of n-grams (the predicted sentence and the gold standard sentence) are. A bag of words means that the order is not important, and the difference is measured by modified precision. For concreteness, consider the following toy examples:

(44) a. Gold reference: 'one two three four five'

b. predicted sentence 1: 'one one two two two'

c. predicted sentence 2: 'two two two one one'

For simplicity, let's consider unigram precision first. With the bag of words assumption, (44b) and (44c) are identical in terms of unigram because they have the

[4]There are other automatic machine translation evaluation algorithms available, such as translation edit rate (Snover et al., 2006) and Damerau-Levenshtein edit distance (Damerau, 1964; Levenshtein, 1966). BLEU is chosen for the current experiments because it is the most widely used evaluation algorithm, and the correlation between the BLUE score evaluation and human judgment evaluation is also well-acknowledged.

[5]The script can be downloaded from: https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

same set[6] of unigrams:

(45) a. predicted sentence 1: 'one one two two two' =
{'one','one','two','two','two'} =
{'two','two', 'two', 'one', 'one'} =
predicted sentence 2: 'two two two one one'

The unigram bag of word format of the gold-standard of our example is:

(46) gold-standard unigram bag of words
a. {'one','two','three','four','five'}

Now to calculate of the proportional similarity between the predicted bag of words and the gold-standard bags of words, BLEU uses 'modified precision rate'. The technical meaning of 'precision' is whether the predicted items are actually present in the gold-standard. Now the size of the bag of unigrams of the candidate is 5, the denominator of the precision rate is 5, and the denominator is how many items in the candidate set are present in the gold-standard. In the current example, *one* and *two* are both present in the gold-standard, so the denominator of (44b) or (44c) is 5. Now we have a wrongly inflated rate, 5 out of 5, 100% matched, meaning (44b) or (44c) is 100% similar to the gold-standard. To counter the effect of this inflation, BLEU uses 'modified' precision rate. When the item in the gold-standard is matched, it is crossed-out, and invisible to the predicted bag of words[7]. With this modified precision measurement, the two *one*s only get one score and the three *two*s only get one score. Now the modified precision rate is 2 out of 5, instead of 5 out of 5.

In terms of bigrams, the same examples will be:

(47) a. Gold reference: {'one_two', 'two_three', 'three_four', 'four_five'}

[6]Here set does not mean the mathematical set but just unordered list.

[7]This is very similar to the feature checking mechanism in the Minimalist Program: one interpretable feature normally can only check out one uninterpretable feature.

b. predicted sentence 1: {'one_one', 'one_two', 'two_two', 'two_two'}

c. predicted sentence 2: {'two_two', 'two_two', 'two_one', 'one_one'}

The denominator of the precision rate is 4 because the length of the predicted bag of words is 4; predicted sentence 1 in (47b) get 1 score because 'one_two' is matched, yielding a rate of 1 out of 4, while for predicted sentence 2 in (47c) no bigram is matched, yielding a rate of 0 out of 4.

A loose end of the current measurement is that it will wrongly give a shorter predicted sentence a higher precision rate because the shorter the smaller the denominator is. To counter this, the final version of BLUE penalizes short predicted sentence by multiplying the ratio between the length of the predicted sequence and the length of the gold-standard sentence. For N from 1 to 4, each N-gram comparison yields a BLEU score; the multi-BLEU score is just the combination of the 4 BLEU scores (unigram to four-gram).

Put all together, a concise way of describing the calculation of BLUE is the following equation.

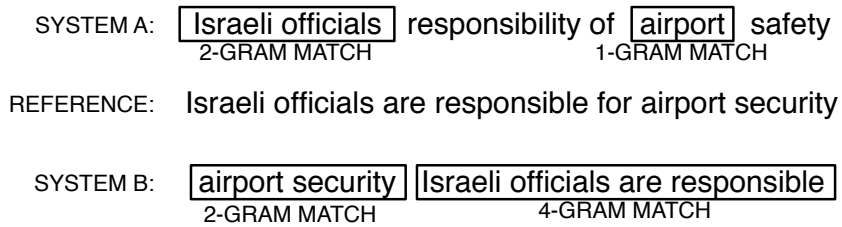
$$\text{BLEU} = \min\left(1, \frac{\textit{output-length}}{\textit{reference-length}}\right) \left(\prod_{i=1}^{4} \textit{precision}_i\right)^{\frac{1}{4}} \tag{4.4.1}$$

For a little bit more complicated example of calculating the multi-bleu score, consider the following example in figure 4.1 from Koehn (2009, p. 226-227).

In short, the BLEU score calculation is an automatic evaluation of how similar two copora are. In the current experiments we are comparing the predicted target sequences with the gold standard. The BLEU score of 100 means the two copora are identical, and the BLEU score of 0 means the two copora are completely distinct from each other.

(48) Gold-Standard = English sentences in (40c) = English sentences in (41c)

FIGURE 4.1. The BLEU score is based on n-gram matches with the reference translation (Koehn, 2009, p. 226-227)



Metric	System A	System B
precision (1gram)	3/6	6/6
precision (2gram)	1/5	4/5
precision (3gram)	0/4	2/4
precision (4gram)	0/3	1/3
brevity penalty	6/7	6/7
BLEU	0%	52%

Note that the gold-standard is the same because they are the same English sentences in the 3-tuples samples. Then the two sets of predicted English sentences are evaluated, yielding two BLEU scores.

(49) Scores:

a. $\text{Score}_{\text{GLOSStoEN}} = \text{BLEU}(\text{Gold-Standard}, \text{Predictions}_{\text{GLOSStoEN}})$

b. $\text{Score}_{\text{GDtoEN}} = \text{BLEU}(\text{Gold-Standard}, \text{Predictions}_{\text{GDtoEN}})$

This procedure of splitting the data into three sub-sets, training the models, and evaluating the models is executed ten times.

4.4.2 Result

After ten rounds of repeated random sub-sampling validation, ten pairs of scores of the two models are generated, as shown in the following table. The average

Round	Gaelic (Baseline)	GLOSS
0	17.29	18.39
1	16.42	18.00
2	15.29	16.02
3	15.97	20.22
4	17.79	19.02
5	16.73	15.53
6	17.11	18.00
7	16.37	20.08
8	15.93	15.82
9	16.99	15.93
Mean	16.59	17.70

TABLE 4.1. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{GLOSStoEn}}$

score of the $\text{Models}_{\text{GLOSStoEN}}$ is only slightly higher than the average score of the $\text{Models}_{\text{GDtoEN}}$. Also, after doing a paired T-test, the difference between the two types of models is not attested (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; $M_{\text{GLOSStoEN}}$=17.70, $SD_{\text{GLOSStoEN}}$=1.78; t(9)=1.97, p=0.080)

4.4.3 Summary

The ultimate practical goal of the dissertation is to use glossing data to develop better machine translation systems. Here *better* means to be better than a baseline system, which is the machine translation system trained with Gaelic-to-English translation samples. The models in (42b) are the baseline systems, and their scores are in the Gaelic column of table (4.1). These are the target scores that we aim to outperform. The experiment above is the first attempt to improve the scores by using the *gloss treatment*, in which the Gaelic sentences are replaced with gloss lines. However, the

result shows that this *gloss treatment* is not effective as the scores of the gloss models are not statistically higher than the baseline Gaelic-to-English models.

4.5 Discussion and Conclusion

It is assumed that the performances of the machine translation systems are correlated with the quality of the representation of meanings in the source sequences. Better representations of meanings yield better machine translation systems. Given the results in (4.4.2) that the gloss models are not better than the Gaelic models, it is concluded that glosses and natural languages are equally good in terms of representing meanings. The strong version of the Gloss-help hypothesis does not hold.

There are several remarks that need to be addressed for the current result.

First, the result falsifies the point of view about glosses in chapter 2 that the gloss line is a golden semantic representation hand-crafted by linguists. It turns that this artificial language, the gloss lines, is only marginal better than Gaelic, as the mean BLEU score of the gloss treatment is slightly higher than that of the baseline systems. This can be viewed as an evidence of language evolution. The written form of a natural language is actually already optimized for representing semantics to the same degree of gloss line representations.

Second, if we want to actually apply the gloss treatment to translate a Gaelic sentence to English, we encounter an immediate problem. The actual source sequence is a Gaelic sentence, while the required source sequence for the gloss treatment is a gloss line. For this treatment to be really viable, we will first need an automatic glosser that convert the Gaelic sentence to a gloss line with 100% accuracy. Given this, even if the gloss treatment should work, it is not practical unless we may convert Gaelic sentence to gloss line perfectly.

In the next chapter, I am going to combine Gaelic and Gloss sentences as the training data to test the weak version of the Gloss-help hypothesis.

Chapter 5
Combining Gaelic Words with Glosses

5.1 Introduction

In the previous chapter, we attempt to build a system by using the *gloss treatment* to outperform the baseline system. It turns out that solely using gloss line is not effective enough to improve the system. However, this result does not falsify the gloss-help hypothesis; instead, it indicates that combination of the gloss line data and the Gaelic sentence data is necessary. In other words, the questions now are:

(50) a. Does adding the gloss data into the Gaelic data will improve the translation system?

b. If yes, what are the right ways of blending these two types of meaning representations together?

This section reports various ways of combining the gloss line data and the Gaelic sentence data, and the experiments and their results using these different treatments. Critically, a specific way of combining Gloss data and Gaelic date (termed as '*Parallel-Partial*' treatment) boosts the performance significantly. The model trained with this specially arranged training data also significantly outperforms Google's Gaelic-to-English translation system.

In this section, I will first describe the most effective treatment, termed as '*Parallel-Partial*' treatment, and the results. Next, I will report the experiments done with other relevant logical treatments (i.e. other ways of combining glossing data and Gaelic data).

5.1.1 The Underlying Heuristics

At a high level, neural net sequence to sequence learning algorithm is learning how to map a high-dimension space to another high-dimension space. In the settings of machine translation, each dot in the high-dimension space is a meaning representation. Linking one dot to another dot is converting one representation of meaning to another, yielding the effect of translation. Given this heuristics, we may just feed the machine with all the available meaning mappings. Given the assumption that the gloss lines are linguistically guided representations of meaning, they are suitable training data for building machine translation systems. Specially, with the gloss data, we let the machine to learn the following mappings:

(51) Mappings Learned in the ParaPart treatment

a. Gaelic sentences → English sentences

b. Gloss lines → English sentences

c. Gloss lines → Gaelic sentences

d. Gaelic words → Gloss items

5.2 The 'Parallel-Partial' Treatment Outperforms Any Other Treatments and the Baseline Significantly

5.2.1 Related work

The Parallel-Partial treatment section may be viewed as a form of multi-task Sequence to Sequence Learning (Luong et al., 2015). Specifically, the parallel part of the treatment is very similar to the data manipulation used in building multi-language translation systems (Johnson et al., 2016).

5.2.2 Data Preprocessing Using the Parallel-Partial Treatment

The Parallel-Partial treatment uses the training and validation data of the baseline system and that of the gloss treatment system. The training and validation data of the baseline system are pairs of a Gaelic sentence and a English sentence (see (41a) and (41b)), and the data of the gloss treatment are pairs of a gloss line and a English sentence (see (40a) and (40b). These two groups of data are combined in a parallel manner in the current treatment. Now the sizes of training set and validation set are doubled. In the baseline system and the gloss treatment system, we have 6,388 samples in the training set and 1,000 samples in the validation set. The current treatment has 12,776 samples in the training set and 2,000 samples in the validation set. This is the *parallel* part of the treatment.

Additionally, I utilize the alignment property between the Gaelic word and the gloss to further build pairs of a Gaelic word and a gloss. These pairs are also included into the training set and validation set of the current treatment. This is the *partial* part of the treatment.

For concreteness, consider the following interlinear glossed text:

(52) Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

With the interlinear glossed text, the parallel treatment will generate three pairs of samples:

(53) a. Gaelic to English:
<"Tha a athair nas sine na a mhàthair", "His father is older than his mother.">

b. Gloss to English:
<"be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother", "His father is older than his mother">

c. Gloss to Gaelic:

<“be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother”, “Tha a athair nas sine na a mhàthair”>

The partial treatment then generates pairs of a Gaelic word and a gloss token:

(54) a. <“Tha”, “be.pres”>

b. <“a”, “3sm.poss”>

c. <“athair”, “father”>

d. <“nas”, “comp”>

e. <“sine”, “old.cmpr”>

f. <“na”, “comp”>

g. <“a”, “3sm.poss”>

h. <“mhàthair”, “mother”>

The samples in (53) and (54) are the training data for this Parallel-Partial treatment.

5.2.3 Results of the Parallel-Partial Treatment

Critically, the same technical settings and the same test sets in the previous experiments are used, and the same procedures are executed. The same split of the original IGTs is used, so as long as it is the same round, the training, validation and test are the same set of IGTs. The only difference is that now the training and validation IGT data are treated with the Parallel-Partial treatment. The results show that the Parallel-Partial treatment has a tremendous effect in improving the baseline system.

The first and the second columns are BLUE scores of the baseline systems and the systems with the Parallel-Partial treatment respectively. The latter is significantly better than the former (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; $M_{ParaPart}$=32.10, $SD_{ParaPart}$=1.33; $t(9)=48.95$, $p<0.01$). The comparison of the average BLUE scores

Round	Gaelic (Baseline)	ParaPart
0	17.29	32.64
1	16.42	32.28
2	15.29	29.94
3	15.97	31.18
4	17.79	32.83
5	16.73	31.11
6	17.11	32.19
7	16.37	33.52
8	15.93	30.93
9	16.99	34.35
Mean	16.59	32.10

TABLE 5.1. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{ParaParttoEn}}$

of the groups of systems shows that the Parallel-Partial treatment improves the performance of the baseline system by 93 percent.

Discussion With the ParaPart treatment, the baseline systems are improved for more than 93 percent. This result suggests the validity of our heuristics in section 5.1.1 that gloss lines can be viewed as an artifical language, and provide strong evidence for the gloss-help hypothesis in (36).

5.3 Other Possible Treatments

This section reports other possible ways of blending the Gaelic sentences and gloss lines[1]. However, all of these treatments are not as effective as the Parallel-Partial treatment. Again, the same procedure and the same test datasets are used across all the experiments.

[1]There must be other possible and logical ways to blend in gloss that are beyond my imagination. It seems me to that by simply attempting to incorporate gloss information we open many other doors to the possible ways of improving machine translation systems. This is another merit of combining theoretical linguists to natural language processing.

5.3.1 The Parallel Treatment

Method of the Parallel Treatment The Parallel treatment is using the parallel part of the Parallel-Partial treatment without exploiting the alignment properties of gloss lines. It is expected that this treatment will improve the baseline systems but will not be as effective as the Parallel-Partial treatment.

With this treatment, a chunk of interlinear glossed text is split into two pairs. For example, the chunk of interlinear glossed text in (55) becomes two samples in (56):

(55) Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

(56) a. Gaelic to English:
<"Tha a athair nas sine na a mhàthair", "His father is older than his mother.">

b. Gloss to English:
<"be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother", "His father is older than his mother">

Results of the Parallel Treatment The experiments gave us the expected results. The table in (5.2) compares the performances of this treatment and the baseline. Critically, the Parallel treatment is effective in improving the baseline systems (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; M_{Para}=29.56, SD_{Para}=1.46; $t(9)=34.42$, $p < 0.01$). However, the best treatment (i.e. the Parallel-Partial treatment) is still far better than this Parallel treatment (M_{Para}=29.56, SD_{Para}=1.46; $M_{ParaPart}$=32.10, $SD_{ParaPart}$=1.33; $t(9)=8.76$, $p < 0.01$). Critically, the comparison between the Parallel-Partial treatment and current Parallel-Only treatment shows the effectiveness of the word-gloss alignments. Our conjecture on the effectiveness is that with the pairs of a gloss item and a Gaelic word present in the training data, the burden of the attention algorithm (Bahdanau

Round	Gaelic (Baseline)	Para
0	17.29	30.40
1	16.42	30.07
2	15.29	26.05
3	15.97	29.67
4	17.79	28.86
5	16.73	29.46
6	17.11	30.57
7	16.37	30.45
8	15.93	28.83
9	16.99	31.28
Mean	16.59	29.56

TABLE 5.2. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{ParatoEn}}$

Round	Gaelic (Baseline)	Para	ParaPart
0	17.29	30.40	32.64
1	16.42	30.07	32.28
2	15.29	26.05	29.94
3	15.97	29.67	31.18
4	17.79	28.86	32.83
5	16.73	29.46	31.11
6	17.11	30.57	32.19
7	16.37	30.45	33.52
8	15.93	28.83	30.93
9	16.99	31.28	34.35
Mean	16.59	29.56	32.10

TABLE 5.3. BLEU scores of $\text{Model}_{\text{GDtoEN}}$, $\text{Model}_{\text{ParatoEN}}$ and $\text{Model}_{\text{ParaParttoEN}}$

et al., 2014) is largely alleviated. In other words, instead of asking the attention algorithm to estimate what to attend to, we explicitly teach the machine the alignment between the Gaelic word and the corresponding gloss.

5.3.2 Interleaving Gaelic Words and Gloss Items And Concating them

Method of the Interleaving Treatment Instead of putting the pairs of a Gaelic sentence and a English sentence and the pairs of a gloss line and a English sentence in a parallel manner, we may just literally blend a Gaelic sentence and a gloss line by interleaving

them[2]. Consider the following example:

(57) a. Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

b. <"Tha be.pres a 3sm.poss athair father nas comp sine old.cmpr na comp a 3sm.poss mhàthair mother", "His father is older than his mother">

Given the chuck of interlinear glossed text data in (57a), the Interleaving treatment generates the sample in (57b).

This way of blending gloss lines and Gaelic sentences may add useful information into the training data; however, the downside of this method is to increase the length of the source sequence. In neural net machine learning, the longer the sequences are, the harder it is to preserve all the information (i.e. it is harder for the attention mechanism to pay attention to the right tokens). So, this treatment may not be effective.

The results are given in the following table.

Round	Gaelic (Baseline)	interleavingGdGLOSS
0	17.29	13.67
1	16.42	12.49
2	15.29	11.01
3	15.97	12.33
4	17.79	12.56
5	16.73	12.13
6	17.11	11.55
7	16.37	12.78
8	15.93	12.43
9	16.99	11.65
Mean	16.59	12.26

TABLE 5.4. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{interleavingGdGLOSStoEn}}$

It turns out this treatment has a significant negative effect (M_{GDToEn}=16.59, SD_{GDToEn}=0.74;

[2]Nadejde et al. (2017) incorporate the Categorial grammar parse tags into natural sentences by interleaving the tags and the words.

$M_{interleavingGdGLOSS}$=12.26, $SD_{interleavingGdGLOSS}$=0.74,; t(9)=-17.06, p=0.000). This is not the right way of incorporating gloss line data.

Method of Concatenating Gaelic Words and Gloss Words A quick and close amendment of the Interleaving approach is to concatenate the aligned Gaelic word and gloss item as a single token. Given the same chunk of interlinear glossed text data, this treatment generates the following sample:

(58) a. Tha a athair nas sine na a mhàthair.
be.pres 3sm.poss father comp old.cmpr comp 3sm.poss mother
'His father is older than his mother.'

b. <"Tha_be.pres a_3sm.poss athair_father nas_comp sine_old.cmpr na_comp a_3sm.poss mhàthair_mother", "His father is older than his mother">

Concatenating words and glosses does solve the long sequence problem; however, it causes the sparse data problem. In this arrangement, the number of the types of tokens is increased; the number of tokens of each type is decreased. Thus, all the samples are put in a larger space. So, the treatment may not be effective either.

Results of Concating Gaelic Words and Gloss Words The performances of this treatment is given in the following table.
The result shows that this treatment hurts the baseline systems instead of improving them (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; $M_{ConcatGLOSSGaelic}$=15.44, $SD_{ConcatGLOSSGaelic}$=1.23,; t(9)=-3.64, p=0.010).

5.3.3 Hybrid: Gaelic or Gloss

Method of Hybrid The Hybrid treatment aims to reduce the potential lexical ambiguity. A Gaelic word may map to multiple glosses, and a glosses may map to multiple Gaelic words. Let's assume a toy example of a chunk of interlinear glossed text data (a one-word sentence):

Round	Gaelic (Baseline)	ConcatGLOSSGaelic
0	17.29	15.42
1	16.42	14.31
2	15.29	15.38
3	15.97	14.18
4	17.79	18.63
5	16.73	14.89
6	17.11	15.16
7	16.37	15.20
8	15.93	15.50
9	16.99	15.72
Mean	16.59	15.44

TABLE 5.5. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{ConcatGLOSSGaelictoEn}}$

(59) Gaelic_word
Gloss_item
English translation

Now we aim to build a single sample that is either <Gaelic_word, English translation > or <Gloss_item, English translation >. To decide, we need to know which one, the Gaelic word or the gloss item, is less ambiguous. The less ambiguous one is the winner. For example, if the Gaelic word is potentially mapped to 10 glosses and if the gloss item is potentially mapped 2 Gaelic words, then <Gloss_item, English translation> is chosen; on the other hand, if the ambiguity situation is reverted, then <Gaelic_word, English translation > is chosen. However, when the situation is tight (i.e. both the Gaelic word and gloss item are equally ambiguous), a default setting needs to be chosen. The choices of the default setting split this single treatment into two treatments: default as Gaelic or default as gloss.

The following is an example of the hybrid treatment:

(60) tha nathairichean a'chuir an t-eagal orm
be.pres snake.vn put det fear on.1s
'Snakes frighten me'

The length of the Gaelic sentence and the gloss line in the above IGT is 6. This means 6 ambiguity comparisons need to be made to decide which one, Gaelic word or gloss, should take the position. The final production of this hybrid treatment on the above IGT is shown as follows:

(61) <“be.pres nathairichean a’chuir det t-eagal on.1s” ,“Snakes frighten me” >

This treatment has the same potential downside as the concatenating treatment: sparsity. In this treatment, the size of the lexicon is the size of the lexicon of Gaelic word plus that of gloss, but what are really visible to the neural net is only about half size of the whole potential lexicon, because for each position it is either a Gaelic word or a gloss.

Round	Gaelic (Baseline)	HybridDefaultAsGaelic
0	17.29	9.44
1	16.42	9.07
2	15.29	7.69
3	15.97	9.12
4	17.79	9.08
5	16.73	10.45
6	17.11	8.62
7	16.37	10.00
8	15.93	10.52
9	16.99	8.46
Mean	16.59	9.24

TABLE 5.6. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{HybridDefaultAsGaelictoEn}}$

Result of Hybrid When the default setting is the Gaelic word, the performances are significantly worse than the baseline systems (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; $M_{\text{ReplacingGaelic}}$=9.24, $SD_{\text{ReplacingGaelic}}$=0.89,; t(9)=-21.03, $p < 0.01$), as shown in table (5.6).

When the default setting is the Gaelic word, the performances are sightly worse than than the baseline systems (M_{GDToEn}=16.59, SD_{GDToEn}=0.74; $M_{\text{ReplacingGaelic}}$=15.47, $SD_{\text{ReplacingGaelic}}$=1.03,; t(9)=-3.67, $p < 0.01$), as shown in table (5.7).

Round	Gaelic (Baseline)	HybridDefaultAsGLOSS
0	17.29	15.95
1	16.42	15.60
2	15.29	14.15
3	15.97	14.72
4	17.79	15.74
5	16.73	14.88
6	17.11	14.45
7	16.37	16.41
8	15.93	15.15
9	16.99	17.61
Mean	16.59	15.47

TABLE 5.7. BLEU scores of $\text{Model}_{\text{GDtoEN}}$ and $\text{Model}_{\text{HybridDefaultAsGLOSS}}$

The results show that this hybrid treatment indeed suffers from the sparsity problem.

5.4 Summary and Conclusion

Chapter 2 argues that the gloss representation is a golden representation of meanings, and thus theoretically with the gloss information incorporated, the performance of machine translation systems should improve. The experiments reported in this chapter reveal an effective way of combining gloss data and Gaelic sentences. It is found that the Parallel-Partial is highly effective, and the Gloss-help hypothesis is empirically supported by the results. The complete BLEU scores of various treatments are given in the following table.

The current experimental results confirm the hypothesis that gloss helps machine translation. However, there are some potential confounding factors. The next chapter discusses these confounding factors, and reports additional experiment results that rule out them.

Round	Baseline	GLOSS	google	ParaPart	Para	Interleaving	Concat	HybrGaelic	HybrGLOSS
0	17.29	18.39	22.09	32.64	30.40	13.67	15.42	9.44	15.95
1	16.42	18.00	25.38	32.28	30.07	12.49	14.31	9.07	15.60
2	15.29	16.02	23.72	29.94	26.05	11.01	15.38	7.69	14.15
3	15.97	20.22	23.21	31.18	29.67	12.33	14.18	9.12	14.72
4	17.79	19.02	22.31	32.83	28.86	12.56	18.63	9.08	15.74
5	16.73	15.53	23.41	31.11	29.46	12.13	14.89	10.45	14.88
6	17.11	18.00	24.53	32.19	30.57	11.55	15.16	8.62	14.45
7	16.37	20.08	22.78	33.52	30.45	12.78	15.20	10.00	16.41
8	15.93	15.82	25.67	30.93	28.83	12.43	15.50	10.52	15.15
9	16.99	15.93	23.42	34.35	31.28	11.65	15.72	8.46	17.61
Mean	16.59	17.70	23.65	32.10	29.56	12.26	15.44	9.24	15.47

TABLE 5.8. BLEU scores of the all treatments

Chapter 6
Tying Up Some Loose Ends

The results of the experiment in the previous chapter show that the Parallel-Partial treatment improves the baseline system (system that trained on Gaelic-to-English parallel data only) tremendously. As such, the hypothesis that gloss information helps machine translation is supported.

However, there are loose ends that need to be tied up. First, the effectiveness of the Parallel-Partial treatment is demonstrated when compared with other internal treatments. We would like to know how well it performs when compared to other well-established machine translation systems. Additionally, there are some potential confounding factors that may jeopardize the validity of the experiment results.

To tie up these loose ends, this chapter reports additional experiments and results.

6.1 Comparison with Google Translation

The Parallel-Partial treatment yields superb systems compared to the baseline systems, but is it still good when we compare it to other well-established machine translation systems? To answer this question, I compare our systems with Google translation. Specifically, I used a free Google translation API (Han, 2018) to translate the Gaelic sentences in our test set. Then I calculated the BLEU scores of Google's predicted outputs with the target sequences of our test set as the gold standard. In this manner, Google translation is just like an additional treatment in the comparison. The result is shown in the following table:

It turns out the systems trained with the Parallel-Partial treatment also vastly outperform Google translation.

Round	Gaelic (Baseline)	Gloss	ParaPartial	Google
0	18.39	17.29	32.64	22.09
1	18.00	16.42	32.28	25.38
2	16.02	15.29	29.94	23.72
3	20.22	15.97	31.18	23.21
4	19.02	17.79	32.83	22.31
5	15.53	16.73	31.11	23.41
6	18.00	17.11	32.19	24.53
7	20.08	16.37	33.52	22.78
8	15.82	15.93	30.93	25.67
9	15.93	16.99	34.35	23.42
Mean	17.70	16.59	32.10	23.65

TABLE 6.1. BLEU scores of $\text{Model}_{\text{ParaPart}}$ and Google Translation

6.2 Oversampling

A potential confounding factor of the effectiveness of the Parallel-Partial treatment is that this treatment has more English sentences than the baseline systems.

Recall what the training data for both treatments are:

(62) Mappings Learned in the Baseline treatment:

a. Gaelic sentences → English sentences

(63) Mappings Learned in the ParaPart treatment:

a. Gaelic sentences → English sentences

b. Gloss lines → English sentences

c. Gloss lines → Gaelic sentences

d. Gaelic words → Gloss items

Specifically, because the Parallel-Partial treatment learns the mappings in (63a) and (63b), the English sentences are oversampled. Oversampling of the target sequences has a positive effect, because the decoding process of the neural net machine translation system uses the language model information of the target language. Now given that in the ParaPart treatment the quantity of English sentences (i.e. the target

language) is doubled, the effectiveness of the ParaPart treatment may be just a result of this oversampling.

To exclude this confound, I oversampled the mappings in the baseline systems by repeating the mappings, so that we can have a fair comparison. Specifically, I doubled, tripled, and quadrupled the training data of the baseline treatment. The following table shows results of the oversampled baseline systems.

Round	ParaPart	Gaelic	Double Gaelic	Triple Gaelic	Quadruple Gaelic
0	32.64	17.29	29.05	28.52	29.74
1	32.28	16.42	28.61	27.04	25.53
2	29.94	15.29	23.78	28.19	23.60
3	31.18	15.97	27.50	28.77	28.61
4	32.83	17.79	25.51	28.43	28.61
5	31.11	16.73	27.88	26.28	25.31
6	32.19	17.11	25.72	27.73	26.50
7	33.52	16.37	27.12	27.84	28.84
8	30.93	15.93	25.20	28.40	29.19
9	34.35	16.99	26.39	28.44	28.73
Mean	32.10	16.59	26.68	27.96	27.47

TABLE 6.2. BLEU scores of $\text{Model}_{\text{ParaPart}}$ and Gaelic Treatment with Oversampling

It turns out that oversampling also improves the baseline systems; however, it is still not as effective as the Parallel-Partial treatment. Given these results, the oversampling confound is ruled out.

6.3 Other Hyper-Parameters

Another possible confounding factor is that maybe the set of the arbitrarily chosen hyper-parameters used in the experiments in chapter 5 just accidentally favors the

Parallel-Partial treatment and disfavors the baseline treatment. To exclude the potential confounding effect of the hyper-parameters, I sampled some hyper-parameters by changing the size of word embedding (into $\{100, 200, 300, 400, 500\}$) and the size of the mini-batches (into $\{16, 32, 64, 128\}$) using the same training, validation, and test data used in round 0. The advantage of the Parallel-Partial treatment is consistent across the different settings of the hyper-parameters. The results are shown in the following table:

Round	WordEmbeddingSize	MiniBatchSize	Gaelic	Gloss	ParaPart
0	100	16	28.85	31.07	33.08
0	100	32	22.02	25.28	32.41
0	100	64	17.29	12.01	31.58
0	100	128	17.29	0.00	26.02
0	200	16	29.38	30.45	35.10
0	200	32	25.47	26.78	33.55
0	200	64	15.24	18.29	32.00
0	200	128	0.00	0.00	27.41
0	300	16	29.82	31.62	34.14
0	300	32	27.21	29.18	33.68
0	300	64	14.75	18.65	32.41
0	300	128	6.35	2.76	28.72
0	400	16	29.96	32.03	34.18
0	400	32	25.87	27.64	34.27
0	400	64	16.19	19.81	32.52
0	400	128	6.21	8.14	28.29
0	500	16	31.26	30.81	34.70
0	500	32	26.30	28.42	35.16
0	500	64	17.29	18.39	32.64
0	500	128	8.03	8.57	30.24
mean			19.74	20.00	32.10

TABLE 6.3. BLEU scores of Round 0 using different Hyper-Parameters

In the table, we can see that the sizes of word embedding and mini-batches do have an effect on the performance of the trained system. However, the Parallel-Partial treatment still outperforms the baseline no matter what the hyper-parameters are.

6.4 Interlinear Glossed Text Data In Other Languages: the Universality

Given the above results, it is properly defended that the gloss information of Scottish Gaelic can improve Scottish Gaelic to English neural machine translation systems. An interesting leak of the gloss-help-hypothesis is that maybe it only works on Gaelic. Maybe the Gaelic language just has some special property that makes the gloss information relevant. Now the question is:

(64) Does the gloss-help-hypothesis hold universally across different languages?

This section reports the experiment result using the same settings as in chapter 5 except for the fact that the source language is not Gaelic.

The Online Database of Interlinear Text (Lewis and Xia, 2010; Xia et al., 2016) is a perfect database to run this experiment. Specifically, this database extracted the interlinear glossed texts in published linguistic papers that are open on the Internet. The interlinear glossed texts are stored in a consistent and machine readable format, called xigt (xml for interlinear glossed text). The Online Database of Interlinear Text contains interlinear glossed texts in 1,495 different languages; however, only 18 languages have more than 1,000 chunks of interlinear glossed text (i.e. the 3-tuples of a source sentence, a gloss line, and an English translation). In the current translation experiment, I use the Online Database of Interlinear Text in these 18 languages.

The result is shown in the following table:

Language Names	Language (Baseline)	GLOSS	ParaPart	Sample size
Mandarin	1.13	0.00	10.25	3719
German	0.00	3.13	18.08	4037
Greek	0.00	0.00	0.00	1188
Finnish	0.00	0.00	2.56	1968
French	0.96	0.00	9.49	3099
Hausa	0.00	0.00	4.41	1490
Hungarian	0.00	0.00	0.00	1197
Indonesian	0.00	0.00	2.34	1211
Icelandic	0.00	0.00	9.55	1719
Italian	0.00	0.00	0.00	1366
Korean	2.70	2.57	11.90	3630
Dutch	0.00	0.00	9.50	1454
Norwegian	0.00	0.00	3.16	1403
Polish	0.00	0.00	5.25	1713
Passamaquoddy	0.00	0.00	7.41	1166
Russian	0.00	0.00	4.64	2266
Spanish	0.00	0.00	4.42	1733
Turkish	0.00	0.00	2.62	1420

TABLE 6.4. BLEU scores of other languages

The paired t-test shows that the Parallel-Partial treatment outperforms the baseline systems ($M_{Baseline}$=0.32, $SD_{Baseline}$=1.33; $M_{ParaPart}$=5.87, $SD_{ParaPart}$=4.84,; t(17)=-5.56, $p < 0.01$) and the GLOSS treatment (M_{GLOSS}=0.27, SD_{GLOSS}=0.70; $M_{ParaPart}$=5.87, $SD_{ParaPart}$=4.84,; t(17)=-5.19, $p < 0.01$).

The result shows that incorporating gloss information works effectively across many different languages. Note that for Greek and Italian the ParaPart treatment has no effect. I do not have a well-supported explanation for now, and also it may be true that gloss information may help different languages to different degrees. In other words, for some languages gloss information is a huge booster while for other languages it is a small booster. The relation between the property of a language and how much gloss information helps that language is left for my future research.

6.5 Conclusion

This chapter describes additional experiments and results to confirm the effectiveness of gloss information by excluding other potential confounding factors. Given all these results, we may conclude that the gloss information is an effective booster for neural net machine translation systems across different hyper-parameters and languages.

Chapter 7

Conclusion and Future Research

7.1 Future Research

In my current work, I discovered that the gloss information boosts up machine translation systems. This is not the end of the story, but just a beginning, as it opens the door to many interesting questions:

(65) a. The Gloss representation is not actually standardized. There are many variants. How and what to gloss is based on the linguist's theoretical interests. For example, if one is studying gender agreements, the gender information must be properly glossed. However, if one is interested in the metrical structure, one may just gloss the word as its syllable weight[1]. Is there a way to determine what type of gloss representation is most suitable for a type of natural language processing task?

b. Are the machine translation BLEU scores correlated to the correctness of the glosses? For example, a linguistic theory argues that X should be glossed as p and q in different context, while another linguistic theory argues that X should be glossed as p in all context. Can we use machine translation experiments to validate these two theories? Can we say the theory with a higher BLEU score is more accurate than the other?

c. In chapter 6, I used the gloss from 18 different languages. It seems that for some language the gloss information is really helpful, and for others it is not that helpful. Can we come up with a typological generalization on the pattern of the BLEU scores based on the properties of the language?

[1] see Mahdavi Mazdeh (2018) for examples for this type of notation.

d. Gloss information helps machine translation. How about other linguistics information, like part of speech tags, and parse trees? Can we incorporate all the information? How so?

e. What is the optimal hyper-parameter setting? Can we make a sensible linguistic interpretation on the setting? In chapter 6, I tried various hyper-parameters to train the models. The optimal setting for the Gaelic treatment is Word Embedding Size being 500 and Mini-Batch size being 16; for Gloss treatment the optimal setting is 400 and 16; for the Parallel-partial treatment, it is 500 and 32. What is going on here?

In short, there are many other puzzling questions for my future research.

7.2 Conclusion

In the dissertation, I introduce a very effective way of incorporating the gloss data into neural net machine translation systems.

The interlinear glossed text representation already has its own merit in theoretical linguistic studies even without the discussion of the current dissertation. It is so basic and so widely used in linguistics studies. The most important discovery of the dissertation is that linguistics can be practically useful.

This fact suggests that gloss information is relevant to machine translation and other natural language processing applications.

How theoretical linguistics may work hand in hand with natural language processing, and how neural net machine learning may exploit linguistics are important questions in both fields (see Pater (2017) for a nice discussion on this topic). In addition to practically building better machine translation systems, the current work also exemplifies how theoretical linguistics may work hand in hand with natural language processing successfully.

The more fundamental potential influence of the current documentation is to show that the gloss line representation is an ideal meeting point for natural language processing and theoretical linguistics to understand and help each other.

The scientists in linguistics and natural language precessing related computer science are studying the black box of human languages. If opening the black box is a competition between the two camps (linguistics and natural language precessing), and the evaluation is how useful it is in real life, the natural language precessing camp is making good progress, while linguistics is like an underdog. If all the arguments reported in the dissertation should be sound, the current dissertation is a loud shouting voice from the linguistics camp that linguistics may have a practical and positive effect in machine translation and other natural language processing applications.

To build ordinary systems for natural language processing, theoretical linguistics is optional. However, to build extraordinary systems, theoretical linguistics is necessary.

Bibliography

Adger, David (2003), *Core syntax: A minimalist approach*, volume 33. Oxford University Press Oxford.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014), "Neural machine translation by jointly learning to align and translate." *arXiv preprint arXiv:1409.0473*.

Berwick, Robert C and Noam Chomsky (2015), *Why only us: Language and evolution*. MIT press.

Bickel, Balthasar, Bernard Comrie, and Martin Haspelmath (2008), "The leipzig glossing rules. conventions for interlinear morpheme by morpheme glosses." *Revised version of February*.

Brown, Peter, John Cocke, S Della Pietra, V Della Pietra, Frederick Jelinek, Robert Mercer, and Paul Roossin (1988), "A statistical approach to language translation." In *Proceedings of the 12th conference on Computational linguistics-Volume 1*, 71–76, Association for Computational Linguistics.

Brown, Peter F, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Fredrick Jelinek, John D Lafferty, Robert L Mercer, and Paul S Roossin (1990), "A statistical approach to machine translation." *Computational linguistics*, 16, 79–85.

Brown, Peter F, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer (1993), "The mathematics of statistical machine translation: Parameter estimation." *Computational linguistics*, 19, 263–311.

Chen, Yuan-Lu (2010), *Degree Modification and Time Anchoring in Mandarin*. Master's thesis.

Chen, Yuan-Lu, Andrew Carnie, Michael Hammond, and Colleen Patton (2018), "Developing an auto-glosser for scottish gaelic using a corpus of interlinear glossed text." The 10th Celtic Linguistics Conference.

Cheng, Chin-Chuan (1973), *A synchronic phonology of Mandarin Chinese*, volume 4. Walter de Gruyter.

Cho, Kyunghyun, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio (2014a), "On the properties of neural machine translation: Encoder-decoder approaches." *arXiv preprint arXiv:1409.1259*.

Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio (2014b), "Learning phrase representations using rnn encoder-decoder for statistical machine translation." *arXiv preprint arXiv:1406.1078*.

Chomsky, Noam (2005), "Three factors in language design." *Linguistic Inquiry*, 36, 1–22, URL https://doi.org/10.1162/0024389052993655.

Chomsky, Noam (2007), "Approaching ug from below." In *Interfaces + Recursion = Language?: Chomsky's Minimalism and the View from Syntax-Semantics* (U. Sauerland and H.M. Gartner, eds.), Studies in Generative Grammar [SGG], 1–29, De Gruyter.

Damerau, Fred J (1964), "A technique for computer detection and correction of spelling errors." *Communications of the ACM*, 7, 171–176.

Domingos, Pedro (2012), "A few useful things to know about machine learning." *Communications of the ACM*, 55, 78–87.

Grano, Thomas (2008), "Mandarin hen and the syntax of declarative clause typing." *Unpublished manuscript. Accessed online:< http://home. uchicago. edu/˜ tgrano/grano_hen. pdf>. First accessed*, 4.

Graves, Alex, Greg Wayne, and Ivo Danihelka (2014), "Neural turing machines." *arXiv preprint arXiv:1410.5401*.

Han, SuHun (2018), "Googletrans." https://github.com/ssut/py-googletrans.

Johnson, Melvin, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. (2016), "Google's multilingual neural machine translation system: enabling zero-shot translation." *arXiv preprint arXiv:1611.04558*.

Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang (2016), "On large-batch training for deep learning: Generalization gap and sharp minima." *CoRR*, abs/1609.04836, URL http://arxiv.org/abs/1609.04836.

Kiperwasser, E. and M. Ballesteros (2018), "Scheduled multi-task learning: From syntax to translation." *ArXiv e-prints*.

Klein, Guillaume, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush (2017), "Opennmt: Open-source toolkit for neural machine translation." *CoRR*, abs/1701.02810, URL http://arxiv.org/abs/1701.02810.

Koehn, Philipp (2009), *Statistical machine translation*. Cambridge University Press.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. (2007), "Moses: Open source toolkit for statistical machine translation." In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, 177–180, Association for Computational Linguistics.

Kotsiantis, Sotiris B, I Zaharakis, and P Pintelas (2007), "Supervised machine learning: A review of classification techniques." *Emerging artificial intelligence applications in computer engineering*, 160, 3–24.

Kratzer, Angelika and Irene Heim (1998), *Semantics in generative grammar*. Blackwell Oxford.

Lamb, William (2001), *Scottish Gaelic*, volume 401. Lincom Europa.

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

Lewis, William D. and Fei Xia (2010), "Developing odin: A multilingual repository of annotated language data for hundreds of the world's languages." *Literary and Linguistic Computing*, 25, 303–319, URL +http://dx.doi.org/10.1093/llc/fqq006.

Liu, Chen-Sheng Luther (2010), "The positive morpheme in chinese and the adjectival structure." *Lingua*, 120, 1010–1056.

Luong, Minh-Thang, Eugene Brevdo, and Rui Zhao (2017), "Neural machine translation (seq2seq) tutorial." *https://github.com/tensorflow/nmt*.

Luong, Minh-Thang, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser (2015), "Multi-task sequence to sequence learning." *arXiv preprint arXiv:1511.06114*.

Luong, Minh-Thang, Hieu Pham, and Christopher D Manning (2014), "Effective approaches to attention-based neural machine translation." *arXiv preprint arXiv:1410.5401*.

Mahdavi Mazdeh, Mohsen (2018), "The quantitative nature of meters in persian folk songs and pop song lyrics." In *First North American Conference on Iranian Linguistics*.

Mei, Tsu-Lin (1991), "Tone sandhi and morphological relics." *Journal of Chinese Linguistics Monograph Series*, 454–471.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean (2013a), "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781*.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean (2013b), "Distributed representations of words and phrases and their compositionality." In *Advances in neural information processing systems*, 3111–3119.

Mitchell, Tom Michael (2006), *The discipline of machine learning*, volume 9. Carnegie Mellon University, School of Computer Science, Machine Learning Department.

Nadejde, Maria, Siva Reddy, Rico Sennrich, Tomasz Dwojak, Marcin Junczys-Dowmunt, Philipp Koehn, and Alexandra Birch (2017), "Predicting target language ccg supertags improves neural machine translation." In *Proceedings of the Second Conference on Machine Translation*, 68–79.

Papineni, Kishore, 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 on association for computational linguistics*, 311–318, Association for Computational Linguistics.

Pater, Joe (2017), "Generative linguistics and neural networks at 60: foundation, friction, and fusion."

Pierce, John R and John B Carroll (1966), "Language and machines: Computers in translation and linguistics."

Pyle, Dorian and Christina San Jose (2015), "An executive's guide to machine learning." *Mckinsey Quarterly,(3)*, 44–53.

Rosenblatt, Frank (1958), "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review*, 65, 386.

Roweis, Sam T and Lawrence K Saul (2000), "Nonlinear dimensionality reduction by locally linear embedding." *science*, 290, 2323–2326.

Sennrich, Rico and Barry Haddow (2016), "Linguistic input features improve neural machine translation." *arXiv preprint arXiv:1606.02892*.

Shih, Chilin (1997), "Mandarin third tone sandhi and prosodic structure." *Linguistic Models*, 20, 81–124.

Siegelmann, Hava T (2003), "Neural and super-turing computing." *Minds and Machines*, 13, 103–114.

Siegelmann, Hava T (2012), *Neural networks and analog computation: beyond the Turing limit*. Springer Science & Business Media.

Siegelmann, Hava T and Eduardo D Sontag (1991), "Turing computability with neural nets." *Applied Mathematics Letters*, 4, 77–80.

Smith, Samuel L., Pieter-Jan Kindermans, and Quoc V. Le (2017), "Don't decay the learning rate, increase the batch size." *CoRR*, abs/1711.00489, URL http://arxiv.org/abs/1711.00489.

Sneddon, James Neil, K Alexander Adelaar, Dwi N Djenar, and Michael Ewing (2012), *Indonesian: A comprehensive grammar*. Routledge.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul (2006), "A study of translation edit rate with targeted human annotation." In *In Proceedings of Association for Machine Translation in the Americas*, 223–231.

Strubell, E., P. Verga, D. Andor, D. Weiss, and A. McCallum (2018), "Linguistically-informed self-attention for semantic role labeling." *ArXiv e-prints*.

Wang, Chiung-Yao and Yen-Hwei Lin (2011), "Variation in tone 3 sandhi: The case of prepositions and pronouns." In *Proceedings of the 23rd North American Conference on Chinese Linguistics (NACCL-23)*, 138–155.

Williams, Philip, Rico Sennrich, Matt Post, and Philipp Koehn (2016), "Syntax-based statistical machine translation." *Synthesis Lectures on Human Language Technologies*, 9, 1–208.

Xia, Fei, William D. Lewis, Michael Wayne Goodman, Glenn Slayden, Ryan Georgi, Joshua Crowgey, and Emily M. Bender (2016), "Enriching a massively multilingual database of interlinear glossed text." *Language Resources and Evaluation*, 50, 321–349, URL https://doi.org/10.1007/s10579-015-9325-4.

Zhang, Niina Ning (2013), *Classifier Structures in Mandarin Chinese*, volume 263. Walter de Gruyter.