Multimodal Semantic Understanding: Semantic Role Labeling with Vision and Language

by

Zijiao Yang

B.E., Dalian University of Technology 2016

B.E., Ritsumeikan University, 2016

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Master of Science

Department of Computer Science

2020

Committee Members:

James Martin

Martha Palmer

Chenhao Tan

Yang, Zijiao (M.S., Computer Science)

Multimodal Semantic Understanding: Semantic Role Labeling with Vision and Language

Thesis directed by Prof. James Martin

Semantic role labeling (SRL) is an important task in natural language understanding systems. Recently, researchers have moved from traditional syntactic feature based pipelines to end-to-end (merely token to tag, mostly) neural network based SRL system, and state-of-the-art performance has been reported from those systems. Nonetheless, the recent trend of inducing syntactic information back to neural models has led to a big success [Strubell et al., 2018]. On the other hand, language understanding should not be a single modal task, the most pervasive evidence is ourselves. Incorporating information from other modalities has drawn attention [Baltrusaitis et al., 2019]. This thesis introduces two different models trying to utilize image information to help SRL task. The two models use distinct ways to integrate between the vision and language modalities. Also, the models are trained on automatically generated data, and evaluated on ground truth data. The results and analysis are presented.

Dedication

To my mom, dad, my girlfriend Yuping, and all my friends.

Acknowledgements

I want to thank Professor. James Martin, for his help throughout my thesis process, as well as his insightful opinions on various other projects that could be related to this thesis. I would also like to thank my committee members. This thesis originates from a class project of Professor Martha Palmer's Computational Lexical Semantics class. And Professor Martha Palmer provided me valuable feedback on related topics and connects me with her student Abhidip Bhattacharyya. Also they helped with a lot problems regrading the datasets I used. I thank Professor Chenhao Tan for his valuable comments and feedback and some most important skills I learned for proceeding this thesis is within a class with him. Finally, I want to thank Abhidip Bhattacharyya for the valuable discussions we had and the tools he pointed me to.

Contents

Chapter

1 Introduction **1**

1.1 The Necessity of Knowledge Enrichment to NLP Systems 1

1.2 Task Selection 3

1.3 Goal Description and Question Proposition 4

1.4 Thesis Outline 4

2 Previous Work **5**

2.1 Semantic Role Labeling 5

2.2 Multimodal Learning 10

3 Data Description **14**

3.1 Datasets Description 14

3.2 Training and Validation Data Generation 15

3.3 Test Data Description 18

4 Model Description **20**

4.1 Text Model 21

4.1.1 Task Formulation 21

4.1.2 BiLSTM 22

4.1.3 Highway Connections 24

4.1.4 Decoding 26
4.2 Alignment Model 26
4.2.1 Image Module 26
4.2.2 Alignment Module 28
4.2.3 Loss 29
4.3 Attention Model 29
4.3.1 Attention Module 30
4.4 Training Setup 30

5 Results and Analysis **33**
5.1 Results 33
5.2 Analysis 34
5.2.1 Ablation 34
5.2.2 Confusion Matrix 35
5.2.3 Attention and Decoding 36

6 Conclusion and Future Work **49**
6.1 Conclusion 49
6.2 Future Work 50

Bibliography **52**

Tables

Table

2.1 Some of the common modifiers in current Propbank (not a complete list, only include the semantic role concerned in this thesis) [Bonial et al., 2012], each row is a modifier, its short definition, and one example. Attention on the emphasized texts. 8

5.1 This table gives the average precision, recall and F1 over five training instances with different random initialization for t-SRL, al-SRL, at-SRL. In side the brackets, the standard deviation across 5 training instances is reported 34

5.2 This table gives the precision, recall, and f1 for each semantic role label, along with overall score for t-SRL model results (This is not a complete table) 35

5.3 This table gives the precision, recall, and f1 for each semantic role label, along with overall score for al-SRL model results (This is not a complete table) 36

5.4 This table gives the precision, recall, and f1 for each semantic role label, along with overall score for at-SRL model results (This is not a complete table) 37

5.5 This table gives the result of ablation, involved models are the baseline model t-SRL, at-SRL, at-SRL without highway connection (no hi), and at-SRL without decoding (no decoding), the precision, recall and F1 for each model is reported. The number in brackets represent the F1 drop compared to at-SRL 38

5.6 This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for t-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table) 39

5.7 This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for al-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table) 39

5.8 This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for at-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table) 45

5.9 Performance drop without decoder on precision, recall, F-1, for t-SRL, al-SRL, and at-SRL 45

Figures

Figure

2.1 An example of span-based SRL 6

3.1 An example from MSCOCO dataset. Each image is paired with five captions 16

3.2 An example from Flickr30k dataset. The image data contains bounding boxes denoted by color-coded squares. (The bounding boxes shown here is from original Flickr30k dataset, which is human annotated, but the bounding box features used in this thesis are generated follow [Tan and Bansal, 2019]) The entities in the captions are also color-coded. Same color indicates correspondences between bounding boxes and entities mentioned in captions. 17

4.1 3 layer BiLSTM with highway connections, pink arrow denotes the highway connection; the red boxes are the gates that control the integration of highway knowledge and hidden states. 22

4.2 The alignment model al-SRL: it consists of three parts: image module(bottom left, blue box), BiLSTM (bottom right), and alignment module (where the alignment loss is computed). A sentence embedding is extracted from BiLSTM hidden states and resized image embedding to be fed into the alignment module for computing alignment loss. Two losses are integrated to form final loss. 27

4.3 The attention module: the initial layers of text part are identical to t-SRL4.1. Bounding box features are obtained by passing the concatenation of bounding box vector and coordinates to the image module, as described in 4.2.1. Then context vector is computed for each hidden state by attention module. Finally, the context vectors and hidden states are concatenated together to make final predication. 32

5.1 The confusion matrix for t-SRL, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 40

5.2 The confusion matrix for al-SRL, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 41

5.3 The confusion matrix for at-SRL. (note it is a simplified version of the complete confusion matrix.) Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 42

5.4 This heatmap represents the confusion difference between t-SRL and at-SRL without decoding. It is computed by subtracting the confusion matrix of at-SRL (no decoding) from t-SRL (no decoding). Blue color represents the confusion decrease for at-SRL compared with t-SRL; for example, $cell_{LOC,ARG2}$ represents at-SRL is less confused about LOC with ARG2 by 11. Furthermore, red color means the confusion increase for at-SRL compared with t-SRL. 43

5.5 This heatmap represents the confusion difference between t-SRL and at-SRL with decoding. It is computed by subtracting the confusion matrix of at-SRL from t-SRL. Blue color represents the confusion decrease for at-SRL compared with t-SRL; for example, $cell_{LOC,ARG2}$ represents at-SRL is less confused about LOC with ARG2 by 14. Furthermore, red color means the confusion increase for at-SRL compared with t-SRL. 44

5.6 The confusion matrix for t-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 46

5.7 The confusion matrix for al-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 47

5.8 The confusion matrix for at-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j 48

Chapter 1

Introduction

1.1 The Necessity of Knowledge Enrichment to NLP Systems

Language is an effective tool for communication, but communication is never a single-modal process. Furthermore, language should not be a standalone system; intuitively, at the initial step of language acquisition, multi-model information is involved. [Harnad, 1990] promotes the idea of making semantic interpretation of a formal symbol system intrinsic to the system, instead of just parasitic on the meanings in our heads, argues it is necessary for cognition. The problem was defined through an example: if we want to learn Chinese as a native language, and the only information resource we have is a Chinese dictionary, then to learn would be to pass endlessly from one meaningless symbol-string to another. Through this method, we will never understand the meaning of any symbol-string. Therefore, I conjecture that this is likely not the actual path on which we come through this far towards language learning. As for our current NLP systems, benchmark scores are refreshed every few weeks, with bigger models (like Bert) [Devlin et al., 2018] and more computation power, they seem to fulfill some of our needs for building better applications. Nevertheless, from the perspective of creating "Artificial Intelligence", the question remains: Do they truly understand human language, or do they start to understand human language beyond merely targeting the pre-designed goal from data set as well as model design?

[Forbes et al., 2019] examined to what extent natural language representations demonstrate physical commonsense reasoning. In their investigation, they tried to check if the representations could capture the relationship between affordances of objects, the action applicable to the object

(like a car can be driven), and properties of the object (like a car requires fuel). Moreover, they found the current neural models are not able to capture those interplays. They make the conjecture that this is unsurprising since the affordances of an object and the properties of an object are unlikely to show up together following the "distributional hypotheses" [Harris, 1954] (that words show up in similar context have similar meanings). The mutual mentions of these two types of information naturally lead to expression redundant. They concluded that, currently, representations learned no more than the associations that are written down explicitly.

So what are the possibilities for encoding more information into our language representation? One would be getting human involved directly, and let our model try to capture the multi-modal information provided by humans, and request whatever and whenever they need. The algorithm could have a better chance of learning more complex relations by pattern recognition on the "original complex data" from humans. However, this would be very slow and expensive in terms of both time and resources. Another approach is to construct a model of the world we humans observing and experiencing every day so that if the world has the "atoms" necessary, then the inductive bias we need from the model can evolve from its interaction with the world model. [Lake et al., 2018] has pointed out similar quests for if we are aiming at building machines that could learn and think like people. [Lake et al., 2018] suggested we need to build causal models of the world that supports understanding and explanation beyond merely pattern recognition, and we need to enhance the learned knowledge with physical and psychology theories. Although it is interesting as it sounds, this line of work remains as toy experiments like [Mordatch and Abbeel, 2017; Lazaridou et al., 2017].

Although the two approaches aforementioned are not easily accessible, the necessity of integrating more knowledge into natural language processing (NLP) beyond just explicitly written seems undoubtedly important at some point in the NLP research going forward. Furthermore, we need to aim at making semantic meaning to be intrinsic to the system [Harnad, 1990], meaning the system should have more knowledge of what is a "cat" in addition to the "cat" and "dog" co-occur in similar positions in a text. Without the "deeper" intrinsic knowledge of the language, a system

does not have the potential for tasks like casual reasoning and commonsense reasoning, as they require the system to understand and manipulate the meaning of the text.

1.2 Task Selection

Semantic role labeling (SRL) [Gildea and Jurafsky, 2002; Carreras and Màrquez, 2004, 2005; Palmer et al., 2010], is a task trying to determine from a given sentence: **Who** did **What** to **Whom**, **When**, and **Where**.

The selection of this task is due to the following reasons:

- Previous work [Shi et al., 2019] on predicting syntactic structure from pairs combined of image and text shows the effectiveness of image information on making inference of syntactic information. Moreover, it is believed that syntactic information is beneficial for SRL as described in [Punyakanok et al., 2008]. Furthermore, the former state of the art model on SRL [Strubell et al., 2018] testified the benefits of syntactic information. Following this chain, the natural hypothesis is that image information could be of help for semantic role labeling task by providing the information needed for inferring syntactic information. Therefore, this offers a reasonable signal that integrating image information might be helpful so that further investigation could be performed.

- The SRL task has been considered as an important task toward building better natural language understanding systems. And it is also called shallow semantic role parsing. This representation could serve as a semantic representation for feeding into various downstream tasks that consider semantic information. Such as Question Answering, Text Summarization, Machine Translation. Therefore, focus on this task provides a test-bed for investigation on whether knowledge of enrichment (image information) could bring about the enrichment of the semantic meaning representation acquired by the model. (And it should be "deeper" than the type of meaning represented by co-occurred in similar contexts)

1.3 Goal Description and Question Proposition

The goal of this thesis research is to try to enrich the representation by providing the model that aims at a natural language understanding task, the possibilities to ground onto its corresponding image. And analyze the performance of the proposed model, to testify if the enrichment proposed could benefit the model performance, in what ways it is beneficial? Further, I want to discuss the possibilities and necessity of the fusion between multiple modalities.

My questions proposed for this thesis project is mainly:

(1) Can images be helpful in improving accuracy in semantic role labeling?

(2) If images do help, how do they help? Could it provide us any insights on the necessity of multi-modal fusion? If it does not, why?

(3) What are good ways to fuse the information from differing modalities?

1.4 Thesis Outline

Chapter 2 reviews relevant prior work. Chapter 3 describes the data set and its related problems. Chapter 4 addresses the different model designs involved. Chapter 5 describes the experiments performed, presents the results along with the analysis. Furthermore, Chapter 6 gives the final discussion, conclusions, and future directions.

Chapter 2

Previous Work

In this chapter, I will start by reviewing the SRL related works: the definition of SRL, the forms of SRL, the major breakthrough of SRL, and current state of the art (SOTA) work of SRL. Then I will describe some recent research spotlights on multi-modal learning, what it is, the necessity of it, and the recent mainstreams of achieving multi-modal learning. Then I will end this chapter by giving a brief description of how the reviewed works are combined to form this thesis. Moreover, details will be elaborated in the next chapter.

2.1 Semantic Role Labeling

SRL extracts meaning representation from a sentence regarding *Who did what to whom.* The first automatic semantic role labeling system is developed in [Gildea and Jurafsky, 2002] based on FrameNet. Since then, the SRL has become a prevalent task in the NLP community. Semantic role labeling (SRL): The goal of the task is to develop a model to recognize the arguments of verbs in a sentence, and label them with semantic roles. The combination of a verb and its set of arguments form a *proposition*, and multiple propositions could exist for one sentence. (An example of SRL is shown in Fig 2.1) The arguments extracted are supposed to be non-overlapping, and the argument, as well as verbs, could be split into non-contiguous parts. For example, in the sentence "A baby is being coddled to [$_{\text{V}}$ calm] him [$_{\texttt{C-V}}$ down] as he finishes a crying tantrum", "calm down" as a verb phrase, is split into two parts non-contiguously. There are two argument annotation representations, span-based, which follows from the syntactic constituents of a sentence, therefore,

forms spans. Current span-based SRL research mainly focuses on the released data, and annotation format from CoNLL shared task of 2004, 2005, and 2012 [Carreras and Màrquez, 2004, 2005; Hovy et al., 2006]. The second annotation representation, Dependency-based SRL, requires the model to identify the syntactic heads of arguments rather than the whole span. It is proposed by CoNLL 2008 and 2009 [Surdeanu et al., 2008; Hajič et al., 2009]. Traditionally, the pipeline for SRL contains four subtasks: predicate detection, predicate sense disambiguation, argument identification, and argument classification. Predicate detection identifies which are the verb/verb phrase for the targeting sentence; predicate sense disambiguation finds the correct meaning of the verb in the context of the sentence, namely, the verb sense, (see lexical resources in [Baker et al., 1998; Schuler and Palmer, 2005; Palmer et al., 2005; Hovy et al., 2006]) for the detected verb. Then, argument identification needs to identify the correct argument spans or syntactic heads. Finally, argument classification gives the detected argument correct semantic role labels. The purpose of this task overlaps with natural language understanding (NLU). Therefore, often, SRL could be beneficial to some of the NLU tasks, like machine reading [Zhang et al., 2018; Wang et al., 2015], question answering [Shen and Lapata, 2007], machine translation [Bazrafshan and Gildea, 2013; Shi et al., 2016]. The semantic representation annotation I consider in this thesis is span-based SRL, and

Figure 2.1: An example of span-based SRL

it follows from The Proposition Bank (PropBank) [Palmer et al., 2005]. In PropBank, each verb has several senses, and for each sense, a specific role set is given. The role set is composed of generic labels like Arg0, Arg1, etc. The *frame files* are defined following the above description. For example, below is the simplified definition for lemma run sense "MAKE.01"

MAKE.01 create

Arg0	creator
Arg1	creation
Arg2	created-from, thing changed
Arg3	benefactive

Example: Loews Corp makes Kent cigarettes.

Arg0	Loews Corp
Arg1	Kent cigarettes

Note in the example given in 2.1. The explanations follow "Arg*" are not the formal definition, but easy-to-read descriptions. Typically agents are labeled as Arg0, Arg1 is for the arguments undergoes the change of state or is being affected by the action [Palmer et al., 2005], the patient. Arg2 stands for the instrument, benefactive, attribute. Arg3 and Arg4 usually represent the starting point and the ending point. These definitions are rather generic. They may vary for a specific role set of a verb sense. There are also modifiers, ARG-Ms that are universal across verbs; they represent the time, location, goal, and other modifications for the event described in the sentence. 2.1 is a list of examples of the available modifiers: [Palmer et al., 2005]

Traditionally, SRL systems rely heavily on syntactic information, and they are mostly feature-based. Namely, they start giving each input string a parse, then for each predicate, they collect a feature set for each node in the parse tree and train supervised classifiers to assign each node a semantic role following PropBank convention or FrameNet. The features extracted are a combination of the targeting predicate (since they define the role set), the phrase type of the constituent, the path in the parse tree from constituent to the predicate, part of speech tags, etc. [Gildea and Jurafsky, 2002; Jurafsky, 2000]. Nonetheless, the description above is only an oversimplification. The real system consists of multiple stages: feature selection and argument identification. Since

Table 2.1: Some of the common modifiers in current Propbank (not a complete list, only include the semantic role concerned in this thesis) [Bonial et al., 2012], each row is a modifier, its short definition, and one example. Attention on the emphasized texts.

Modifiers	Definition	Examples
Comitative (COM)	who an action was done with	I sang a song **with my sister**
Locative (LOC)	where	I read book **in Library**
Directional (DIR)	motion along some **path**	walk **along the road**
Goal (GOL)	goal of the action	The cild fed the cat **for her mother**
Manner (MNR)	how action is performed	words **well**
Temporal (TMP)	when	I read **in the morning**
Extent (EXT)	the amount of change	raised prices **by 15 percent**
Reciprocals (REC)	reflexives, reciprocals	**himself, itself**
Secondary Predication (PRD)	an adjunct of a predicate carrying some predicate structure	The boys pinched them **dead**
Purpose Clauses (PRP)	motivation of action	I review materials **in order to prepare for test**
Cause Clauses (CAU)	reason for action	I got cold **due to the changing weather**
Discourse (DIS)	markers connect two sentences	and, but
Modals (MOD)	modal verbs	will, may, can
Negation (NEG)	markers of negative sentence	not, never
Adverbials (ADV)	modifier of the event structure of the verb but do not fall in above labels	probably, possibly
Construction (CXN)	label arguments projected by a construction	comparative constructions: **similar to**

the algorithm implicitly assumes the prediction of each argument is independent, a final step is needed to resolve the inconsistency problem. The traditional SRL systems heavily rely on syntactic information, and people have argued that syntactic information is a prerequisite for SRL systems [Punyakanok et al., 2008]. However, recently, the trend of neural model-based SRL systems evoked a different approach by training SRL system in an end-to-end fashion, requires no explicit modeling of the syntactic information, and usually, the only inputs to the model are the raw tokens and the predicate in focus [Zhou and Xu, 2015; He et al., 2017; Tan et al., 2018]. The trend of not using syntactic information comes from various reasons, on the one hand, the pipeline might become more rather cumbersome, and cascade errors could be induced from each intermediate step, from

example, parsers, on the other hand, an effective parser might just not available for some languages. I will review some representative work in this line of work in the next paragraphs.

[He et al., 2017] used a deep highway BiLSTM structure, and explore various ways of constrained decoding follows from [Punyakanok et al., 2008]. It follows the end-to-end fashion that the input of the system is only sentence-predicate pairs. They are converted into word embeddings and binary embeddings to form the whole feature vector. Although they success at excelling previous SOTA results of CoNLL2005 and CoNLL2012 by a large margin, through their analysis, they argue syntactic information could still be of help as their model still suffer inconsistencies compared to non-neural models, and they found significant improvement could be induced by leveraging gold syntax in their oracle experiment. It is the baseline system I use, and more details will be covered in chapter 3.

[Tan et al., 2018], also end-to-end SRL system, utilize newly trending multi-head self-attention [Vaswani et al., 2017] on top of the traditional bidirectional LSTM outputs/Convolutional neural network/FeedForward neural network to address the long-range dependencies problem that is observed in RNN-based models targeting SRL tasks (as well as other NLP tasks though). Moreover, the work focused on span-based SRL and achieved the SOTA results at the time on CoNLL-2005 and CoNLL-2012.

While an amount of current span-based SRL systems trained with no explicit syntactic information. LISA (Linguistically-Informed Self-Attention) [Strubell et al., 2018] combines self-attention and multi-task learning, so that cumbersome pre-processing for preparing syntactic features are not needed, but instead the syntactic information is induced from jointly training SRL system with dependency parsing, part-of-speech tagging, predicate detection. It achieved SOTA results for CoNLL2005 and CoNLL2012 SRL at the time. Furthermore, it proved the gain of inducing syntactic information for neural model-based SRL systems.

I believe syntactic information certainly matters, but we should also try to avoid the cumbersome of traditional SRL pipelines and encounter a lack of useful external syntactic information, like a high-quality parser. From [Shi et al., 2019], I think it is reasonable to believe that image

information could induce syntactic information we need from a very intuitive perspective. Also, could object relations and other information encoded in image, bring us some new insights? Or at least some performance benefits? Next chapter, I will give a brief review of some multi-modal learning works.

2.2 Multimodal Learning

Why multimodal learning? First off, multimodal learning means we ought to build models than can process inputs from multiple modalities, find connections between them, then solve our problem of various kinds. The world around us is multimodal, and humans perceive, learn, reason about the world follow a multimodal fashion. Therefore, naturally, multimodal seems to be a milestone we want to get pass towards artificial intelligence as it has to understand the multimodal world surrounds us and be able to decode the relationship between them. For example, if person A is trying to show person B the way to the train station, A needs to understand what a train station is, what properties of the building could help identify a train station, namely its functionality and appearance. Also, A needs to understand the question, what does "show the way to place" actually mean, what action might be involved in solving this problem, and what other object and concept could be involved?. For B, B should have a common ground with A about everything might be involved in this task, like B has to understand the instructions given by A, the entities used by A when given instructions. Another level above, A should be able to guess B's intent for going to the train station by observing the person's vocal properties, facial expressions, etc. For example, if B is going somewhere in a hurry, A might suggest alternatives to help B arrive earlier rather than taking a train.

This task happens every day every moment all over the world. However, it is involved in that it requires the ability to retrace the information from different modalities, create connections between them, and on top of that, making inferences and propositions towards task solving. Throughout the solving process of the task, the dialog happens between A and B is continuous information transformation across multiple modalities. This type of task fills up people's daily

life. Therefore, the importance of multimodal learning is evident. [Baltrusaitis et al., 2019] gives a survey of multimodal learning and provides a taxonomy focus on five challenges: representation, which concerns how we should representing data from heterogeneous sources, especially when data from multiple modalities have very different properties. For example, language could be symbolic and discrete, whereas image could be numeric and continuous; Translation address when we want to map from one modality to another, typical tasks in this category include image captioning [Karpathy and Fei-Fei, 2017], where we want to "translate" image into a natural language description of the image, the caption. Alignment concerns how ingredients from different modalities are related together. For example, in the image-sentence retrieval task [Karpathy et al., 2014], parts of the image need to be aligned with parts of sentences for effective retrieval. Fusion concerns how to utilize the information that comes from different modalities for effective prediction. In [Kahou et al., 2016; Wöllmer et al., 2010], researchers try to utilize information of different modalities to predict emotion. Co-learning is the generalized term for language grounding. It concerns how to use the knowledge from one modality to help the training concerns other modalities. My thesis is directly related to the representation problem, alignment problem, fusion problem, the objective of this thesis is to achieve co-learning.

Therefore, I will review the multimodal learning works following their taxonomy with a focus on representation, alignment. And discuss the relationship of my thesis with these two problems along with fusion and co-learning, as well as the SRL work reviewed in last section 2.1.

Mainly, There are two types of multimodal representation. One is joint representation. To obtain the joint representation, the model takes all information from all modalities as input passes them through distinct neural networks separately. The resulted representations are passed through other layers of a neural network to form a joint prediction. [Silberer and Lapata, 2014] uses a stacked autoencoder to generate joint text-visual representation; they first build separated representations of text and visual. The concatenated hidden states of those two models are feed into a bimodal autoencoder to map the two modalities into a joint representation. The predictions are made upon the joint representation. The other way to obtain multimodal representation is the coordinated

representation, where separate representations are learned for distinct modalities, and they are coordinate together with some constraints, like similarity function. [Frome et al., 2013] tries to transfer the knowledge of semantic information learned in language model to a visual model that makes object prediction. They trained a visual model and a language model separately, then visual representation and text representation are coordinated together by a similarity function and a hinge rank loss.

Alignment between parts of modalities is a necessity for utilizing multimodal information effectively. Current, neural models address this issue often with attention module, which allows the model to attend to image bounding boxes or language segments as needed. For example, in [Xu et al., 2015], researchers use attention to attend over computed visual features during generating image caption with Recurrent Neural Network. Also, the alignment could be obtained using a similarity metric as in [Karpathy et al., 2014].

Currently, in multimodal learning concerns vision and language, some tasks are quite popular. Visual Question Answering [Antol et al., 2015] is a task, that given a pair of image and a natural language question regarding the image, the system needs to provide a natural language answer, a dataset is provided accompanying with the proposition of the task. MSCOCO dataset [Lin et al., 2014] and Flickr30k dataset [Plummer et al., 2016] provides image accompanying rich information including object labeling and segmentation, also natural language description of the scene in the image. A large amount of advancement in image captioning, visual-language retrieval, and text-to-image task [Xu et al., 2017] is encouraged by the presence of those datasets. NLVR [Suhr et al., 2019] is a visual reasoning task that tries to decide if a sentence is true given images, and the dataset is also provided. The development of these datasets widely stimulates the advance of vision and language research. For example, [Tan and Bansal, 2019] proposes a structure to learn the connections between vision and language. Their model consists of three encoders, object relationship encoder, and language encoder to capture the inner relationships within the modalities, and a cross-modality encoder to capture relationships between modalities. Then a pre-training consists of multiple tasks is performed to find a better initialization that could capture cross-modality connections better.

And indeed, the pre-training tasks they performed, brought about a substantial performance gain on multiple vision-language tasks, especially for NLVR.

My models consists of a baseline 4.1, alignment-based model 4.2, and attention-based model 4.3. Follow the taxonomy given in [Baltrusaitis et al., 2019]. My alignment model could be categorized as using coordinated representation or alignment with constraints. My attention model could be seen as using joint representation and doing alignment with the attention module. Both models fuse knowledge from different modalities to perform a language task, SRL. Moreover, my objective here is related to co-learning defined before, that is, to use information from other modality (image in this case) to help the training concerns text modality, namely, the training to predict semantic role labels for corresponding text spans. Also, as described before, another perspective of my objective follows from the current SRL research trend: to incorporate syntactic information into neural-based SRL models. Furthermore, my approach here is to incorporate syntactic information indirectly from image follows from the findings and insight of [Shi et al., 2019].

Chapter 3

Data Description

In this chapter, I will describe the data used in this thesis project. First, since the task of this project is rather novel, I did not have an existing data set for experimenting. Therefore, I needed to obtain training and test data. So in the following sections, I will first describe the problem I have encounter. Then I will elaborate on how my data is generated/obtained, illustrate my data formation, give some examples of my data.

3.1 Datasets Description

SRL tasks as described in previous work section 2, are often trained on data set generated with Propbank [Palmer et al., 2005] or FrameNet [Baker et al., 1998] conventions. And the data comes from Wall Street Journal [Carreras and Màrquez, 2005] or a collection of news, telephone speech, weblogs, talk shows, etc. [Bonial et al., 2012] They are single modal data set. Moreover, because of the nature of the contents of these datasets (news, talk shows, blogs), it is an unrealistic task for finding images that match the sentences from those types of articles. However, To investigate the effectiveness of inducing image information into SRL systems, we need to find a dataset with images and a description of the images. There are two data sets that fit our needs. The MSCOCO dataset [Lin et al., 2014], and the Flickr30k dataset [Plummer et al., 2016] fit our needs as they provide the image along with its caption.

The Microsoft Common Objects in Context (MSCOCO) dataset contains 91 common object categories, 82 of them have over 5000 labeled instances. Overall, MSCOCO has 328000 images and

2500000 labeled instances. And those images are taken from everyday scenes consisting of common objects in life. MSCOCO provides the category labeling of the objects, Instance spotting (find out all instances belonging to previously identified categories), and image segmentation (so that each object is segmented separately). Each image is annotated with five captions. The MSCOCO data set we used follow the splits from [Karpathy and Fei-Fei, 2017], which contains 82783 images for training, 1000 for development, 1000 for test. However, there are no semantic role labels exist. (See example in Fig 3.1) Flickr30k dataset, on the other hand, was created specifically for encouraging better image-to-sentence models. Flicker30k contains 31783 images focusing people and animals, five captions per image, and co-reference chains are provided links between entities both in the image and the captions of the same image. Then the bounding box features are provided. (See example in Fig 3.2 For the Flickr30k dataset, The captions of the images are currently being annotated in Professor Martha Palmer's group at the University of Colorado at Boulder. I was able to obtain the current annotated portion of the data set. The annotation follows from the format in [Bonial et al., 2012]. Overall, I have both image and text captions for two datasets, but I only have a small portion of the Flickr30k dataset annotated with semantic roles. I will describe my method for obtaining SRL annotated data for MSCOCO as train, validation set, obtaining the image feature and bounding box feature for MSCOCO dataset, and the Flickr30k dataset.

3.2 Training and Validation Data Generation

The SRL generation for training, validation data utilized the state of the art (SOTA) semantic role labeling model based on BERT [Devlin et al., 2018], as described in [Shi and Lin, 2019]. It achieved F1 86.5 on CoNLL 2012 and was the SOTA model when I created the dataset. I used the implementation from *allennlp*. I generated a JSON representation of semantic role labels for each caption in the MSCOCO dataset, 3.1. The generated SRL labels follow standard BIO tagging convention, so that "B-ARG*[1] " stands for the beginning of a semantic role span, "I-ARG*" represents the token is currently inside some semantic role span. Moreover, "O" stands for out

[1] * here stands for all possible SRLs

Figure 3.1: An example from MSCOCO dataset. Each image is paired with five captions

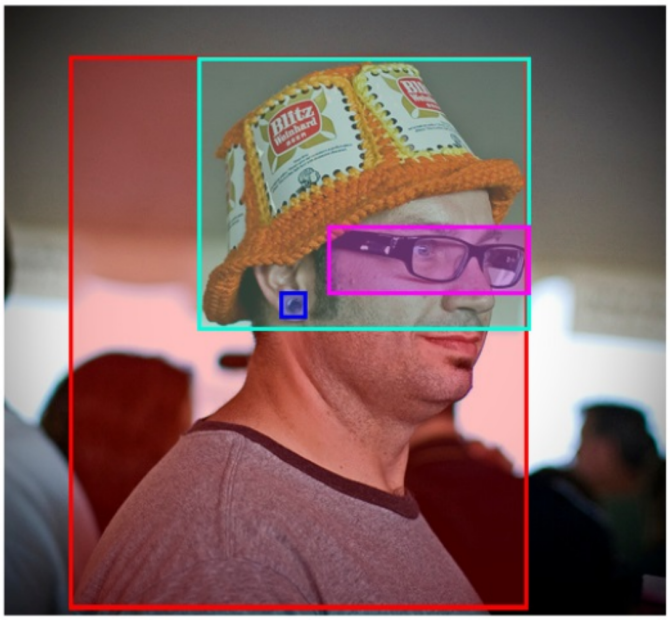
of span of any semantic roles. The generated JSON items are packed into a file, with each line contains a JSON object. Some of the captions do not contain any predicate; the labels of tokens in those sentences are set to O.

```
{'image': 'COCO_train2014_000000358253.jpg',
  'caption': 'There is a bedroom with fans going oft',
  'tag': 'O O O O O B-ARG0 B-V B-ARGM-ADV',
  'predicate': 6}
```

Listing 3.1: Example of generated json formatted data: each data instance contains a image id, the text of image caption, and the tag sequence in BIO-tagging format, and the index of predicate

For image feature generation, I obtained two sets of image features; the first is sets of vectors, which are overall representations of the whole image. The image feature is pre-computed using ResNet-101 [He et al., 2016]. The other sets of image features are fine-grained, that contains object bounding boxes in the images, and the coordinates for each bounding boxes; those bounding boxes

Figure 3.2: An example from Flickr30k dataset. The image data contains bounding boxes denoted by color-coded squares. (The bounding boxes shown here is from original Flickr30k dataset, which is human annotated, but the bounding box features used in this thesis are generated follow [Tan and Bansal, 2019]) The entities in the captions are also color-coded. Same color indicates correspondences between bounding boxes and entities mentioned in captions.



features are generated follows [Tan and Bansal, 2019][2] . The bounding boxes generated are similar to the Flickr30k example shown in Fig 3.2. As a result, the feature set of each image contains the object id, attribute id, their respective confidence, and the locations of the bounding boxes. Each image has 36 bounding boxes generated.

The original MSCOCO captions and whole image embeddings are matched by their order, so that even though the captions are five times number of image embeddings, I could get access to the corresponding image embedding by computing image index with $image\_index = \lfloor caption\_index/5 \rfloor$, $\lfloor \rfloor$ stands for floor division.

[2] I obtained them from Abhidip Bhattacharyya, CS Ph.D. student at the University of Colorado Boulder

The original MSCOCO Karpathy split [Karpathy and Fei-Fei, 2017] contains 82783 images for training, 1000 images for validation, 1000 images for test. Since I did not obtain the whole image embedding features for the test split of MSCOCO, I decided to create the train and validation dataset from the training split (82783) images. Due to some matching issues, the final numbers of train and validation instances are 417565 (corresponding to 66809 images) and 52195 (corresponding to 9057 images). (this is all available BIO-tagging formatted SRL data I generated automatically)

As a result, for my pure-text SRL model 4.1 (baseline), the train data and validation data are the BIO-tagging formatted SRL text data obtained above. The BIO-tagging formatted SRL text data and whole image embedding pre-computed from ResNet-101 forms the train data and validation data of my alignment based image SRL model 4.2. The BIO-tagging formatted SRL text data and bounding box features of images combine to form the training dataset and validation dataset for my attention based image SRL model 4.3.

3.3 Test Data Description

For test data, I obtained the Flickr30k dataset with human-annotated SRLs[3] here in CU Boulder. It is a double-blind annotation with adjudication by a third party for any conflicts. For evaluation purpose, I combined *onecaption.gold_conll, twocaption.gold_conll, fivecaption.gold_conll* from train file. The total instances number is 3177.

Follows from train data generation, I obtained two types of image features for the dataset: whole image embedding and bounding box features for objects in each image. I generated the whole image embedding with ResNet-101 *pytorch*. I obtained bounding box features of the Flickr30k dataset from Abhidip, too. Those images and the text file combinations follow the description in train data generation 3.2 to form the test data for my three models (baseline, alignment-based SRL, attention-based SRL).

Finally, since the train data generated automatically might be missing some labels as some

[3] Since the annotation follows CoNLL format, I converted it to BIO tagging format with a script (which is provided in the annotation folder I obtained).

labels are rarely showing up in the training set. To keep train data and test data consistent in terms of semantic roles that exist in both datasets, I removed the sentences from the test set that has a semantic role label within this set 3.2. After the process, the total number of available test data is 3127. 3.3 shows the final available labels in the train, val, and test dataset.

```
{'ARGM-PRR', 'C-V'}
```

Listing 3.2: The instances containing these labels are removed from flickr evaluation dataset

```
{'ARG1', 'ARG2', 'V', 'ARG0', 'ARGM-LOC', 'ARGM-TMP', 'ARGM-MNR', '
   ARGM-DIR', 'ARGM-ADV', 'ARGM-PRD', 'ARGM-PRP', 'R-ARG1', 'R-ARG0
   ', 'ARGM-COM', 'C-ARG1', 'ARG3', 'ARG4', 'ARGM-ADJ', 'ARGM-GOL',
    'ARGM-MOD', 'ARGM-EXT', 'ARGM-NEG', 'ARGM-PNC', 'R-ARG2', 'R-
   ARGM-LOC', 'ARGM-CAU', 'ARGM-DIS', 'C-ARG0', 'C-ARG2', 'ARGM-REC
   ', 'R-ARGM-MOD', 'ARGM-LVB', 'R-ARGM-TMP', 'ARGM-PRR', 'R-ARGM-
   MNR'}
```

Listing 3.3: The semantic role label vocabulary in train, val, and test data

Chapter 4

Model Description

In this chapter, I will describe my experiments, my model design, my hypotheses of the result of experiments. And present my results along with the analysis I have performed. My goal of experiments is to investigate the following questions:

- Could image information be helpful for SRL?
- How to capture the connection between two modalities (vision and language) to help SRL?
- If the image does provide help in SRL, in what way image information might be beneficial?

My experiments consist of three models: text SRL model (t-SRL), which serve as the baseline; alignment SRL model (al-SRL), which utilize a novel alignment function to align image embedding with corresponding captions; And finally, attention SRL model (at-SRL), which allows the SRL model access to the bounding box features of images during inference, and adjust its attention mechanism as required.

Because of the novelty of this thesis project, the datasets I used, as described in 3 are automatically generated or obtained from currently on-going annotation projects here at CU. Also, as described in 3, the input data for those three models are quite different. Namely, for t-SRL, the only input is the sentences and its SRLs in BIO tagging format; for al-SRL, the input is the sentence annotated with semantic role labels combined with the whole image embedding as described before; Then for at-SRL, the input is the sentence with semantic roles labels combined with the object bounding box features (the embedding for each bounding box, along with the coordinates for each

bounding box). Even though the input data for each model is quite different, the text input is all the same set of image captions (though the format might be a little different, since at-SRL and t-SRL were developed before Flickr dataset is obtained, therefore, the *Datareaders* are different). Moreover, arguably, al-SRL model and at-SRL model, although they provide the capability of inducing image information, the information quality is quite different. For al-SRL, the input image feature is the image embedding for the whole image, whereas, for the at-SRL model, the image feature incorporated is the 36 bounding box embeddings along with their coordinates.

4.1 Text Model

The text model I use here is an implementation of [He et al., 2017]. It is one of the first work that investigates the possibilities of end-to-end training for SRL. The implementation is included in *Allennlp*. It consists of several key model structures that I will elaborate in detail: Bidirectional Long short term memory network, highway connections, and decoding. Nevertheless, first, I will give a formal formulation of the task.

4.1.1 Task Formulation

The task is to predict a tag sequence t given a pair of sentence predicate pair (s, v). $t_i \in t$, and t is transformed into BIO format in advance, namely: For beginning word and inside words of a span of role r, B_r, and I_r are assigned respectively, words outside any role spans are assigned with O. In practice, each token is represented by a word embedding, and it is concatenated with a binary embedding which indicate if the current word is a verb. The BiLSTM, (t-SRL) model is shown in Figure 4.1 Specifically, the prediction of tag sequence $\mathbf{t}$ of a sentence $\mathbf{s}$ is obtained by:

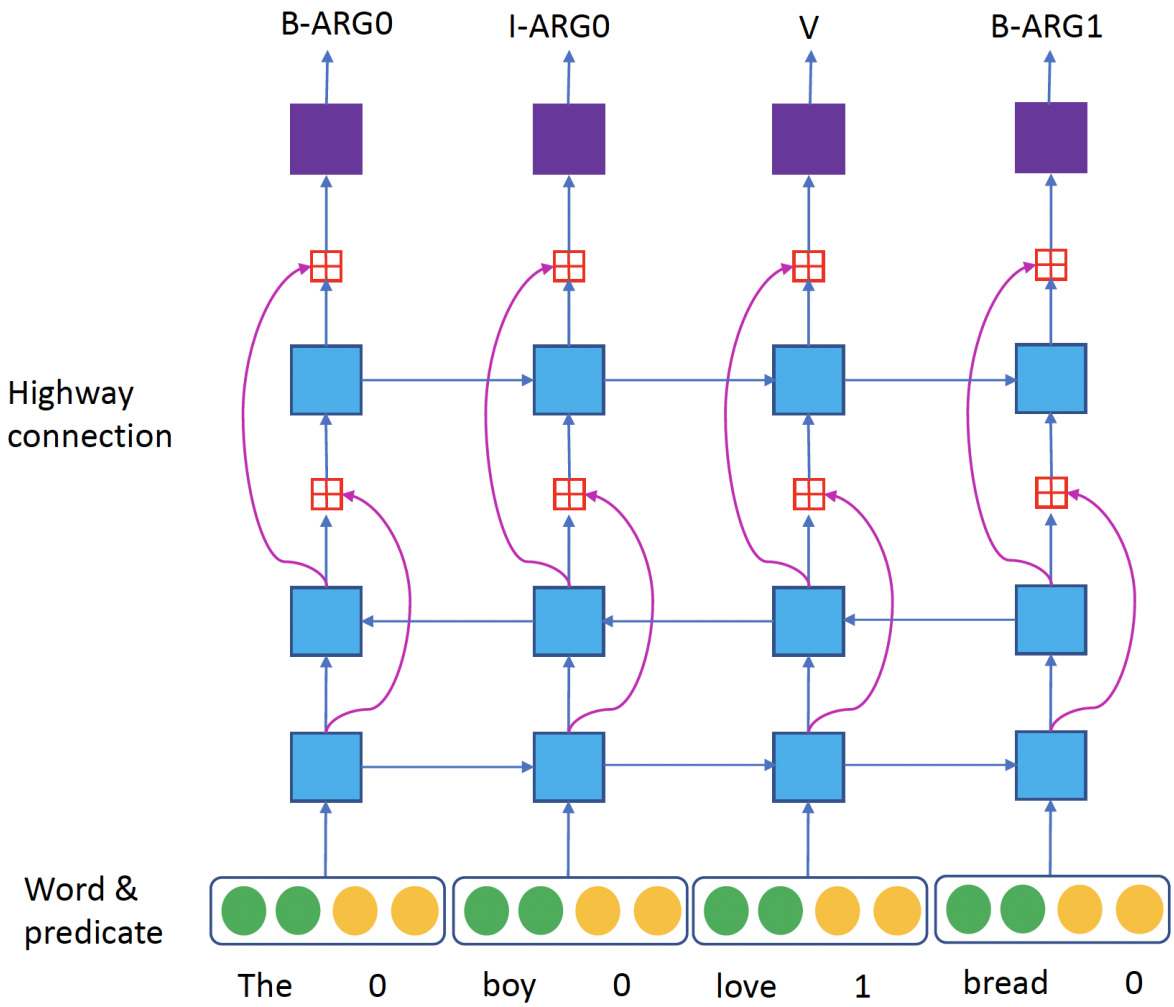
$$t_p = \text{argmax}_{t \in T} f(s, t) \tag{4.1}$$

The loss of BiLSTM is defined as:

$$lstm\_loss = -\sum_{i}^{n} \log p(t_i|s) \tag{4.2}$$

(4.2) is sequence cross entropy, n is the length of the tag sequence

Figure 4.1: 3 layer BiLSTM with highway connections, pink arrow denotes the highway connection; the red boxes are the gates that control the integration of highway knowledge and hidden states.



4.1.2 BiLSTM

Long short term memory network is a variant of recurrent neural network (RNN); it tries to address the problem that long-term dependencies are hard to catch by RNNs. For example, in a language modeling task, where the goal is to predict the next word according to previous information, a text could be “Mike is a professional writer ... Mike spends most time of his day

doing [some action to predict]." If we do not consider the context of the whole text, then the [to-be-predict] could be anything categorized as an action. However, if our model could capture the long-term dependencies, it will utilize the fact that Mike is a writer to infer that the action here is very likely to be writing.

The LSTM is built based on RNN, for RNN, it takes one input (a feature vector, for NLP tasks it is typically a word embedding) at a time, computes a linear transformation followed by a non-linear function to form a hidden state, then the current output is computed by apply another linear transformation followed by a *softmax function*. Start from the second time step, the hidden state computed from last time step is also feed into the computation step of new hidden state. (but really what happens in the start time step, a dummy initial hidden state is served to compute the first hidden state) the last hidden state is applied a linear transformation categorized by a distinct parameter matrix and the result is added to the linearly transformed input embedding, also a bias term is added. Following is the math formulation (4.3) of what I just described:

$$\begin{aligned} a^t &= b_1 + Wh^{t-1} + Ux^t \\ h^t &= \tanh a^t \\ o^t &= b_2 + Vh^t \\ y^t &= \text{softmax}(o^t) \end{aligned} \tag{4.3}$$

Here, h represents the hidden state, y is the output state, a and o are intermediate state for computing hidden state and output state. t stands for the current time step. W is the parameter matrix linear transformation of last hidden states, U is the parameter matrix for linear transformation of current input, V is the parameter matrix for applying linear transformation on the hidden state. b_1 and b_2 are bias terms, they along with *W, V, U* can be optimized (learned) through gradient descent or its variant.

The LSTM, [Hochreiter and Schmidhuber, 1997] uses some tricks to prevent long-term dependency problem. The most important one is **cell state**. It supposed to be a flow of information that the current process could use its information, add information to it, forgets some information as

necessary, and passed it to the next time step. The adding and forgetting process is made possible based on "gate", which is a composed of a sigmoid layer and a point-wise multiplication that could decide a number between 0 and 1 as needed, and multiply it to current information flow to be able to achieve adding and forgetting.

The first step computes the gate value of how much information we want to forget on the cell state. The computing of this process is performed on the concatenation of input data and the last hidden states. Then another gate value is computed to determine how much information to add to the cell state; also the information come from last hidden state and new input is combined and passed through a linear transformation followed by a non-linear function(tanh) to form the candidate values that can be added to the cell states. Next, the cell state is updated based on the computed two gate values and the candidate value information to be added. Finally, what information is being outputted is determined by yet another gate value computed based on the last hidden state and input embedding. Finally, the new hidden state is updated by selecting some information from the cell state using the last computed gate value. The process is tedious to describe, but the essence is: the LSTM has a cell state that preserves information from the very beginning of the process and can be updated based on previous hidden state and incoming input. This alleviates the long-term dependency problem. The math formulation

Finally, the BiLSTM [He et al., 2017] uses a stacked LSTM that the hidden states computed from the last layer are served as the new input for inputting into the next layer of LSTM. Also, note here that the direction of LSTM is different for every pair of adjacent layers of LSTM. If there are four layers of LSTM, then the first and third layers have a direction from left to right, the direction of the second and fourth layers are from right to left.

4.1.3 Highway Connections

The depth is one of the most crucial properties of neural networks; you could catch it from the headlines, like "deep neural network revolution", and it has been a decisive factor for network performance. However, it has been a known issue that deep networks are hard to train. Nowadays,

we know that the most important reason is probably the vanishing gradient problem. A portion of the gradient updates the parameter, and gradients of parameters of different layers need to be multiplied together for computing new gradients for previous layers. Therefore, sometimes, if somewhere, a gradient becomes too small, it will be used to multiply other gradients. As a result, the overall updates on parameter become tiny. It can cause learning to be slow or even stop altogether.

Highway connection [Srivastava et al., 2015a; Zhang et al., 2016] provides a solution by allowing the information from input of current layer to be leaked to next layer so that deeper networks can be trained. The "leaking" functionality is supported by a gating function like LSTM. Highway connection gives the input and the output of the cell (obtained by passing input through a linear transformation followed by a non-linear function) a weight , to control how much information from input and output of the cell should be preserved respectively, the weight can be learned and is characterised by a parameter matrix. And I present the math formula here (4.4): Highway equations:

$$y = H(x, W_H) \tag{4.4a}$$

$$y = H(x, W_H) \cdot T(x, W_T) + x \cdot C(x, W_C) \tag{4.4b}$$

$$y = H(x, W_H) \cdot (x, W_T) + x \cdot (1 - T(x, W_T)) \tag{4.4c}$$

$$y = \begin{cases} x & \text{if } T(x, W_T) = 0, \\ H(x, W_H) & \text{if } T(x, W_T) = 1 \end{cases} \tag{4.4d}$$

$$y_i = H_i(x) * T_i(x) + x_i * (1 - T_i(x)) \tag{4.4e}$$

(4.4a) is the original equation for the cell to compute the output from input, H is typically a linear transformation followed by a non-linear function, x is the input, and W_H is the weight matrix. In (4.4b), T is the transform gate, and C is the carry gate; they control how much information of output and input to pass down to the next layer, respectively. T and C are parameterized by W_T and W_C, respectively. Here C is set to $1 - T$ [Srivastava et al., 2015a] and we have (4.4c). As (4.4d)

shows, the output of highway connection could alter between passing the original input down or just act as a normal cell that computes linear transformation followed by a non-linear function. And finally, in [Srivastava et al., 2015a], they introduce block state so that all existing computation results of cells (neurons) of the current computation module is recognized as block outputs, then pass to the next layer. In 4.4e the subscript i stands for i^{th} block.

4.1.4 Decoding

I used Viterbi decoding [Forney, 1973] to resolve inconsistency problem of predicted BIO formatted results, which is to reject any sequence that is not a valid BIO format, for instance, **I-AR0** without **B-AR0** preceding it, or **I-ARG0** preceded by **B-ARG1**.

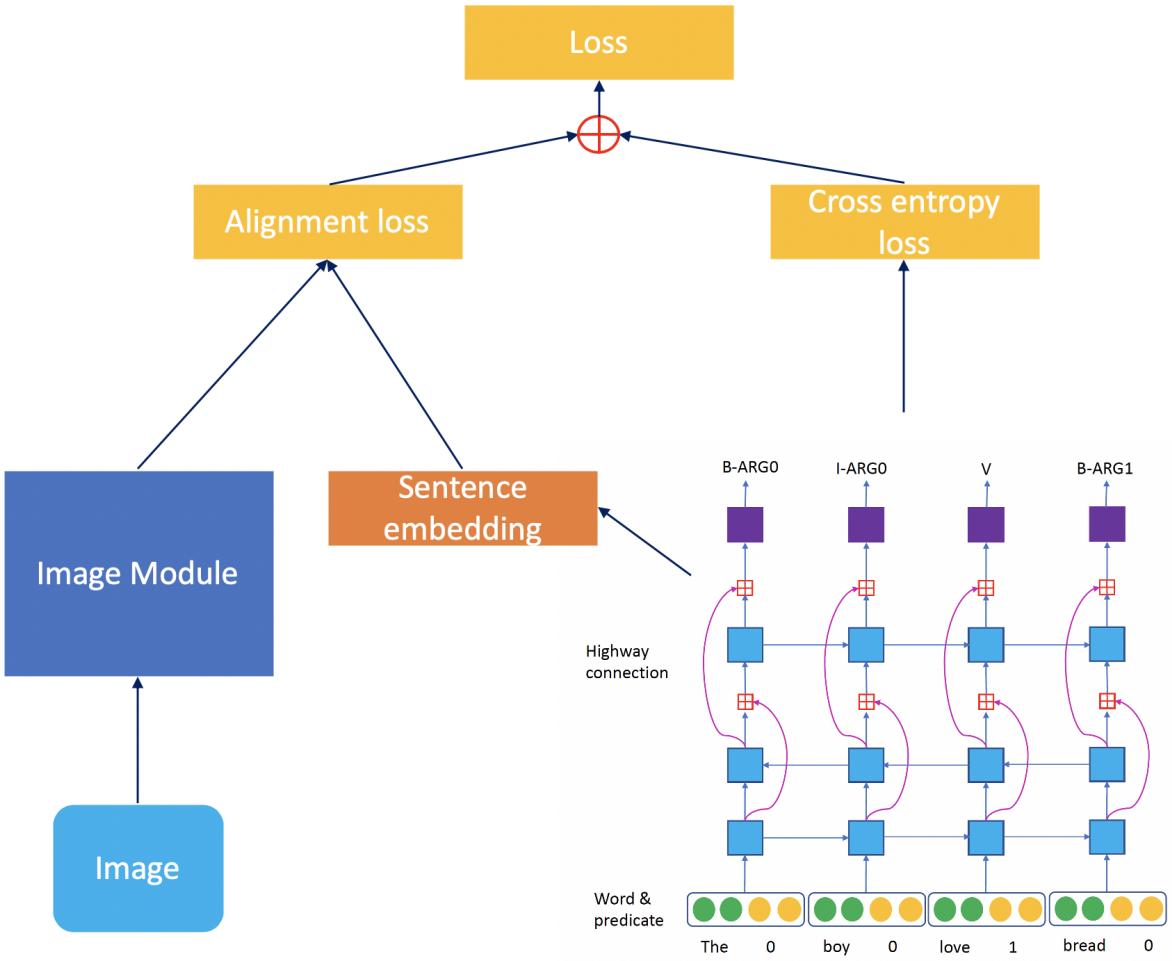
4.2 Alignment Model

The alignment SRL model (al-SRL) consists of three parts, the text module, image module, and alignment module. The text module is adapted from [He et al., 2017], which is the t-SRL I described in 4.1. the image module is a simple feed-forward neural network; the alignment module is realized through a triplet ranking loss function [Wang et al., 2014; Shi et al., 2019]. The task is mostly identical to what defined in 4.1, except that for al-SRL, the whole image embedding of each caption is ready to use during training. The overall architecture of the al-SRL is in Figure 4.2.

4.2.1 Image Module

The image feature I used for this experiment is acquired from [Shi et al., 2019], that obtained from ResNet-101. The image module defined here is a simple feed-forward neural network and layer norm. This module is in charge of making the dimension of image embedding to consistent with the text embedding extracted from the final hidden states of last layer of text module, so that

Figure 4.2: The alignment model al-SRL: it consists of three parts: image module(bottom left, blue box), BiLSTM (bottom right), and alignment module (where the alignment loss is computed). A sentence embedding is extracted from BiLSTM hidden states and resized image embedding to be fed into the alignment module for computing alignment loss. Two losses are integrated to form final loss.



alignment score can be computed.

$$IE = \text{LinearTransformation}(x) \tag{4.5a}$$

$$NormalizedIE = \text{L2LayerNorm}(IE) \tag{4.5b}$$

As presented in (4.5a), the image embedding x originally is a 2048 long vector, it is passed

through a linear layer, then normalized by a *l2LayerNorm* module. The *l2LayerNorm* just normalize the features within a sample so that the features in a sample have zero mean and unit variance; it has been proved to be effective for training RNNs [Ba et al., 2016]. After this computation, the image feature is ready to use for the alignment module.

4.2.2 Alignment Module

The main idea of the alignment module is to align the corresponding image and caption into the same vector space. To achieve this, I use an alignment loss to encourage both the image part and the text part to be aligned. My hypothesis is through the alignment of two modalities, the image information could be indirectly leaked to text part, so that the final SRL prediction could utilize information from both channels.

The alignment module consists of a score function that computes the distance between image embedding and sentence embedding, it is characterized by a parameter matrix Θ, which is the linear module we used in the image module 4.2.1. The final hidden state of last layer of BiLSTM described above is extracted as sentence embedding. Then based on the score function, a loss function is built to encourage corresponding image and caption to be align together in vector space. Here is the math formula of my alignment module:

$$m(\mathbf{s}, \mathbf{v}; \Theta) = \text{cosine}(s, \Theta v) \tag{4.6}$$

$$\begin{aligned} L(v, s) = &\sum_{i,k \neq i,k} [m(s^k, v^i) - m(s^i, v^i) + \delta]_+ \\ &+ \sum_{i,k \neq i} [m(s^i, v^k) - m(s^i, v^i) + \delta]_+ \end{aligned} \tag{4.7}$$

In equation (4.6), s represents the sentence embedding described before. v is the whole image embedding obtained. Θ gives the score function the ability to first convert image embedding to the same dimension as the sentence embedding. Most importantly, it allows the parameter matrix to learn from the alignment encouraging signal from 4.7 (triplet ranking loss). In equation 4.7, δ denotes the margin parameter of the triplet function. The first part of the triplet function

encourages the embedding of image i v^i to be aligned with sentence embedding of captions that corresponding to image i. It discourages embedding of image i v^i to be aligned with sentence embeddings of other captions represented by s^k. Moreover, the alignment score between embedding of image i v^i and sentence embedding of caption i s^i should be larger than the alignment score between image i v^i and caption k s^k by a margin of δ. The second part of the equation 4.7 encourages the same thing, that alignment between the embedding of image i v^i and sentence embedding of caption i s^i. Nevertheless, it discourages the different thing: The alignment between other image k v^k and caption i s^i, and this should be smaller than the alignment between image i and caption i by a margin of δ.

4.2.3 Loss

The loss function consists of two parts, the t-SRL loss defined in 4.2 and the triplet randing loss described in 4.7. A hyperparameter λ is used for integrating those two loss functions:

$$loss = (1 - \lambda) \cdot lstm\_loss + \lambda \cdot triplet\_ranking\_loss \tag{4.8}$$

4.3 Attention Model

The attention model (at-SRL) integrates between t-SRL and an attention module. The attention module could choose to pay attention to certain bounding box features by responding to certain time steps' predication needs on certain hidden states. It will compute a context vector representing current attention information for each hidden state computed from BiLSTM. The computed context vectors are concatenated to their corresponding hidden states, respectively, passed to a linear layer followed by a softmax function to compute the logits needed for the loss function. The loss function for the attention model here is identical to the description in 4.2, which is a standard sequence cross-entropy loss. For a visual description, see 4.3. This model is quite straight forward, and yet, it consumes the bounding box features, and it is reasonable to believe that this

type of image feature contains fine-grained information compared to the whole image embedding.

4.3.1 Attention Module

The attention module I used here, is a standard *linear matrix attention*. To compute the bounding box feature vector, bounding box vector and bounding box coordinate are concatenated. And it is passed to the image module as describe in 4.2.1. For each combination of bounding box feature vector v_i and hidden state of BiLSTM h_j, the computation of the attention weights are in 4.9a

$$a_{i,j} = W^T(v_i; h_j) + b \tag{4.9a}$$

$$a_{i,j} = \frac{exp(a_{i,j})}{\sum_i exp(a_{i,j})} \tag{4.9b}$$

";" in equation (4.9a) denotes the concatenation of two vectors, W here is a weight vector for compute the attention weights. The $a_{i,j}$ computed is normalized across each bounding boxes by passing through a *softmax function* (4.9b). Having computed attention scores for all combinations, the context vector c_j can be computed as follows:

$$c_j = \sum_i a_{i,j} \cdot h_j \tag{4.10}$$

4.4 Training Setup

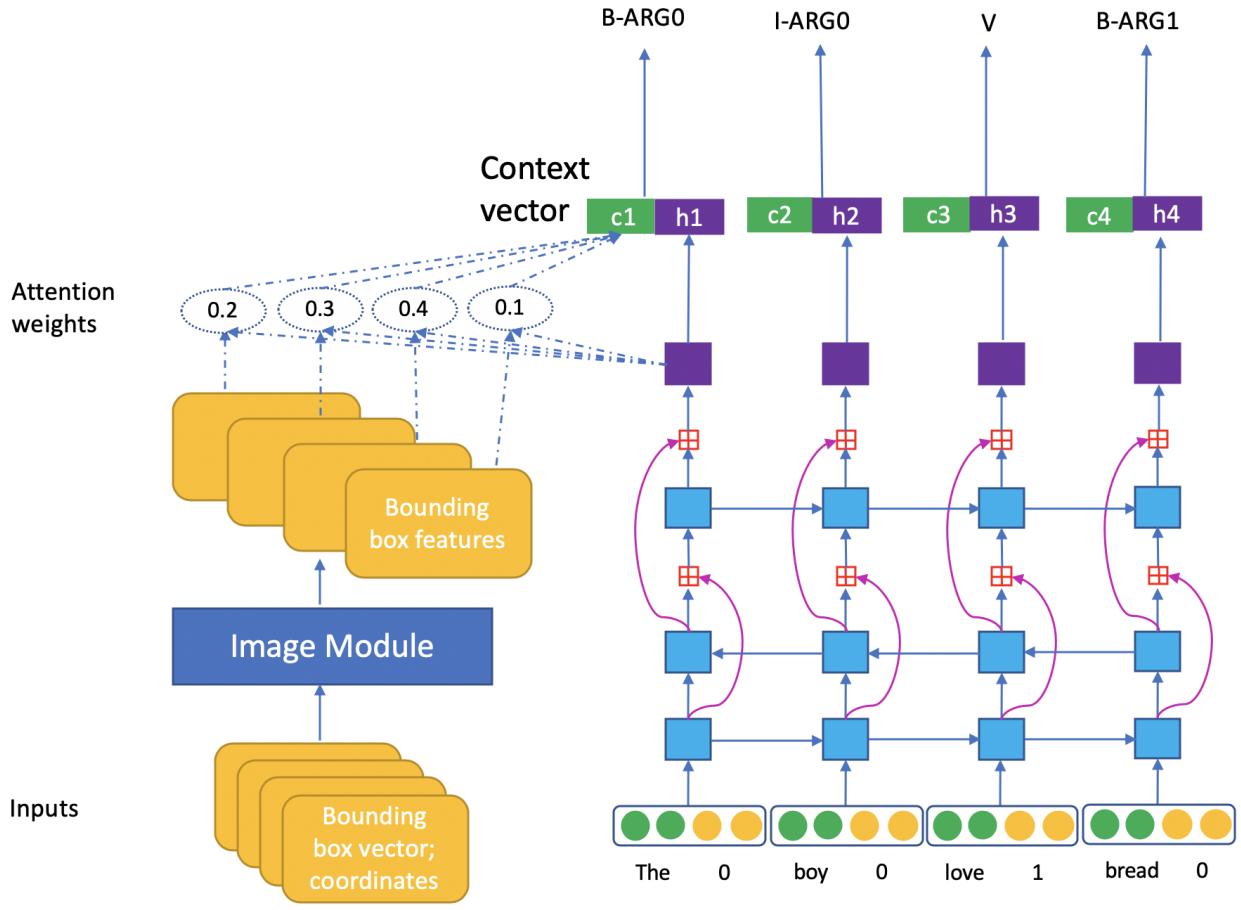
In this section, I will present my training parameters. I trained all three models for 20 epochs, the optimizer used is *Adadelta* [Zeiler, 2012], ϵ and ρ is set as $1e-6, 0.95$ respectively follows from [He et al., 2017], and initial learning rate is set to 1.0. the mini-batch size is set to 80. The gradient is clipped at 1. I use recurrent dropout [Gal and Ghahramani, 2015] for all three models, and the recurrent dropout probability is set to 0.1. The weight matrices in BiLSTM in all three models are initialized by random orthonormal matrices, follow from [Saxe et al., 2013], the weight matrices in image modules in al-SRL and at-SRL, along with weight matrix in attention module 4.3.1 are initialized with Xavier initialization [Glorot and Bengio, 2010]. All three models are trained with 1 RTX 2080.

The word embedding is initialized with GloVe 100-d vectors trained on 6B tokens [Pennington et al., 2014], and the embedding is set to be trainable. Out-of-domain vocabularies are initialized with a randomly generated embedding. All tokens are converted to lowercase before training.

For all three models, the input of BiLSTM is 200-dimensional vectors (100 for word embedding, and 100 for the verb indicator embedding), the hidden size is 300.

For al-SRL and at-SRL, there are some model-specific hyperparameters. The λ of al-SRL is set to 0.2. The feed-forward layer in the image module of al-SRL has an input size of 2048 and an output size of 300. The image module of at-SRL has an input size of 2052 (embedding size + coordinate size) and an output size of 300.

Figure 4.3: The attention module: the initial layers of text part are identical to t-SRL4.1. Bounding box features are obtained by passing the concatenation of bounding box vector and coordinates to the image module, as described in 4.2.1. Then context vector is computed for each hidden state by attention module. Finally, the context vectors and hidden states are concatenated together to make final predication.



Chapter 5

Results and Analysis

5.1 Results

As previously addressed in Chapter 3, the train, validation datasets of this task are automatically generated, and the original data is the MSCOCO dataset. The test data available is annotated by humans, but it is built based on the Flickr30k dataset. Since it is only reasonable to evaluate the prediction of model outputs against ground truth. All three models are evaluated on the Flickr30k test set. I use t-SRL as the baseline model. The results of comparing with baseline should reveal:

(1) The effectiveness of the other two models on the SRL task.

(2) The robustness of all three models when evaluating on data that come from a domain different with training data.

The models are evaluated by the Perl script *srl-eval.pl* included in CoNLL2005. Each model is trained for 5 times with different random initialization. Table 5.2, 5.3, 5.4 gives the resulting metrics of t-SRL, al-SRL, at-SRL respectively, based on precision, recall and F1-score, the results given here and all trained with the same random seed so that the comparison is reasonable. Table 5.1 presents the average scores overall 5 different training instances and the standard deviation across 5 training instances of each model.

From 5.1, we could see among three models, at-SRL consistently performs better on overall recall, precision, F1 compared to the other two models. Notably, it shows that our at-SRL model

is at least comparable to t-SRL, the pure text model, but al-SRL is inferior compared to the other two models. Also, from the standard deviation reported in 5.1, we could see all three models are stable with respect to different random initializations. Among them, t-SRL and at-SRL are more stable than al-SRL.

Table 5.1: This table gives the average precision, recall and F1 over five training instances with different random initialization for t-SRL, al-SRL, at-SRL. In side the brackets, the standard deviation across 5 training instances is reported

Model	Precision	Recall	F1
t-SRL	74.8% (±0.27)	75.3% (±0.50)	75.0 (±0.32)
al-SRL	73.9% (±0.65)	74.8% (±0.48)	74.3 (±0.57)
at-SRL	**75.1%** (±0.44)	**75.6%** (±0.36)	**75.3** (±0.32)

5.2 Analysis

As we saw in results 5.1. The at-SRL seems to surpass the other two models by a reasonable margin. Therefore, in this section, I will do some analysis centers around the at-SRL model. specifically, I will present the ablation analysis, the comparison analysis of before and after decoding results for all three models.

5.2.1 Ablation

For ablation analysis, it is only reasonable to analyze each module's contribution if the input data are the same. I focus on the analysis of at-SRL, for the same reason described in 5.2. Table 5.5 presents the F1, recall, precision of at-SRL, at-SRL with no highway connection, and at-SRL without decoding. All models presented in the table used the same random seed for the initialization of parameters. Moreover, for each ablation experiment, only one module is removed to show the performance difference of with/without that module.

We could see from 5.5, that without attention module, that is t-SRL, F1 dropped by 0.91, without highway connection, F1 dropped by 2.17, without decoding, F1 dropped by 2.23. Therefore, from ablation analysis, we could conclude that decoding and highway connection contribute most

Table 5.2: This table gives the precision, recall, and f1 for each semantic role label, along with overall score for t-SRL model results (This is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	74.53%	74.77%	74.65
ARG0	84.29%	88.59%	86.38
ARG1	77.30%	83.21%	80.14
ARG2	74.89%	63.53%	68.75
ARG3	33.33%	13.04%	18.75
ARGM-ADV	49.07%	41.15%	44.76
ARGM-DIR	52.00%	59.09%	55.32
ARGM-GOL	12.50%	6.25%	8.33
ARGM-LOC	65.71%	71.51%	68.49
ARGM-MNR	44.30%	52.63%	48.11
ARGM-TMP	70.18%	73.21%	71.66

to the model performance, but the attention module is also important.

5.2.2 Confusion Matrix

Here, I present the confusion matrix of three models in table 5.1, table 5.2, table 5.3. The confusion matrix presented here is a subset of high frequent semantic role labels of the complete matrix (for illustration purpose).

Overall, Table 5.1, table 5.2, table 5.3 gives the confusion matrix for t-SRL, al-SRL, at-SRL respectively, and we could see, for t-SRL, ARG0 and ARG1 confuses with each other. This might happen since many inanimates are involved in the event of an image described by a caption, which might lead to a higher probability of confusion between ARG0 and ARG1, as described in [Bonial et al., 2012]. Also, for t-SRL, ARG2 confuses with ARG1, DIR, LOC. [He et al., 2017] pointed out that ARG2 and LOC, DIR could be easily confused, as many verb frames use ARG2 to represent location and direction; ADV confuses with MNR, TMP quite often, it is reasonable since they are all modifiers to describe how an action is performed; LOC confuses with ARG2 a lot, for the same reason of confusing ARG2 with LOC and DIR. For al-SRL, it also confuses between ARG1 and ARG0, and ARG2 confuses with ARG1, DIR, LOC; ADV confuses with MNR and TMP, also LOC confuses with ARG2, DIR, and MNR. Then, for at-SRL, ARG1 and ARG0 are confusing,

Table 5.3: This table gives the precision, recall, and f1 for each semantic role label, along with overall score for al-SRL model results (This is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	72.71%	73.94%	73.32
ARG0	83.81%	88.47%	86.08
ARG1	76.98%	82.29%	79.54
ARG2	70.54%	60.90%	65.37
ARG3	28.57%	8.70%	13.33
ARGM-ADV	41.03%	33.33%	36.78
ARGM-DIR	50.00%	66.67%	57.14
ARGM-GOL	25.00%	12.50%	16.67
ARGM-LOC	61.73%	69.67%	65.46
ARGM-MNR	40.34%	53.38%	45.95
ARGM-TMP	67.69%	74.16%	70.78

ARG2 largely confuses with DIR and LOC; ADV confuses with MNR and TMP, LOC confuses with ARG2.

Through comparisons across confusion matrices as well as from 5.4 and 5.5, we could observe that for at-SRL vs. t-SRL, at-SRL tends to make fewer confusions of LOC with ARG2, also less confuse TMP with ADV, and less confuse ARG0 with ARG1.5.4 However, at-SRL confuses more of MNR with LOC, ARG2 with LOC. The conjecture here is that the at-SRL model tends to correctly label location related entities as location information is rich in image, so at-SRL might effectively utilize that information from the image. However, then, since ARG2 is known to be easily confused with LOC, DIR [He et al., 2017], with more location information induced, it might become more difficult not to confuse ARG2 with LOC. al-SRL performs worse than baseline consistently as presented in 5.1, and this is testified by comparing between the confusion matrix of t-SRL and al-SRL, although there is some variance for each specific label comparison, it is reasonable to say that al-SRL consistently confuses semantic role labels more severe than t-SRL.

5.2.3 Attention and Decoding

In this section, I compare the resulted prediction with and without decoding for all three models. My hypothesis is that due to the induced information from the image, the decoding step is

Table 5.4: This table gives the precision, recall, and f1 for each semantic role label, along with overall score for at-SRL model results (This is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	75.64%	75.47%	75.56
ARG0	85.62%	89.34%	87.44
ARG1	80.08%	84.59%	82.27
ARG2	74.70%	61.78%	67.63
ARG3	8.33%	4.35%	5.71
ARGM-ADV	52.76%	34.90%	42.01
ARGM-DIR	51.05%	61.11%	55.63
ARGM-GOL	33.33%	12.50%	18.18
ARGM-LOC	65.22%	74.82%	69.69
ARGM-MNR	42.18%	46.62%	44.29
ARGM-TMP	70.26%	77.99%	73.92

less necessary for at-SRL and al-SRL than t-SRL since the induced information might help resolve some inconsistencies.

For with and without decoding comparison, first, note here, for results with decoding, the consistency problem across labels are mostly resolved by decoding module, for example, the error 'B-ARGM-LOC' followed by 'I-ARGM-TMP' will not exist in the results after decoding. I want to look for here: Without decoding module explicitly solving the inconsistency problem, how much could the models perform worse, could at-SRL and al-SRL address this problem better due to external image information? For this analysis, we need to focus on the reported scores, as previously mentioned, the decrease of confusion matrix might just due to not taking into account of inconsistent results; therefore, we mainly look at the reported scores and propose some conjectures. In table 5.9. The performance drop with decoder removed for three models are reported, and we could see, although after removing the decoder, at-SRL do drop less than other two models, the performance drop is not significant, so it is not valid enough to say that inducing image information with attention module could help resolve inconsistency problem.

From cross confusion matrix comparison between the without decoding confusion metrics, we could come to the same conclusion described in 5.2.2, that for t-SRL vs. al-SRL, al-SRL consistently performs worse by confusing more of most of the semantic role labels with others.

Table 5.5: This table gives the result of ablation, involved models are the baseline model t-SRL, at-SRL, at-SRL without highway connection (no hi), and at-SRL without decoding (no decoding), the precision, recall and F1 for each model is reported. The number in brackets represent the F1 drop compared to at-SRL

Model	Precision	Recall	F1
t-SRL (no attention)	74.53%	74.77%	74.65 (-0.91)
at-SRL	75.64%	75.47%	75.56
at-SRL (no hi)	71.83%	75.01%	73.39 (-2.17)
at-SRL (no decoding)	72.02%	74.69%	73.33 (-2.23)

Moreover, for t-SRL vs. at-SRL, similarly, I found at-SRL confuses less of LOC with ARG2, and less of TMP with ADV, this means, the less confusion of at-SRL than t-SRL, is not due to the decoding module, but due to the superiority of the at-SRL, the image information incorporation with attention module.

Table 5.6: This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for t-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	70.53%	73.90%	72.17
ARG0	83.14%	88.30%	85.64
ARG1	74.97%	82.86%	78.72
ARG2	71.09%	63.16%	66.89
ARG3	33.33%	13.04%	18.75
ARGM-ADV	33.03%	37.50%	35.12
ARGM-DIR	47.92%	58.08%	52.51
ARGM-GOL	9.09%	6.25%	7.41
ARGM-LOC	61.40%	69.30%	65.11
ARGM-MNR	36.26%	49.62%	41.90
ARGM-TMP	59.75%	68.90%	64.00

Table 5.7: This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for al-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	67.74%	72.77%	70.17
ARG0	82.42%	88.01%	85.12
ARG1	73.56%	81.60%	77.37
ARG2	64.95%	59.90%	62.32
ARG3	22.22%	8.70%	12.50
ARGM-ADV	28.99%	31.25%	30.08
ARGM-DIR	43.16%	62.12%	50.93
ARGM-GOL	18.18%	12.50%	14.81
ARGM-LOC	57.10%	68.01%	62.08
ARGM-MNR	32.35%	49.62%	39.17
ARGM-TMP	54.89%	69.86%	61.47

Figure 5.1: The confusion matrix for t-SRL, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j

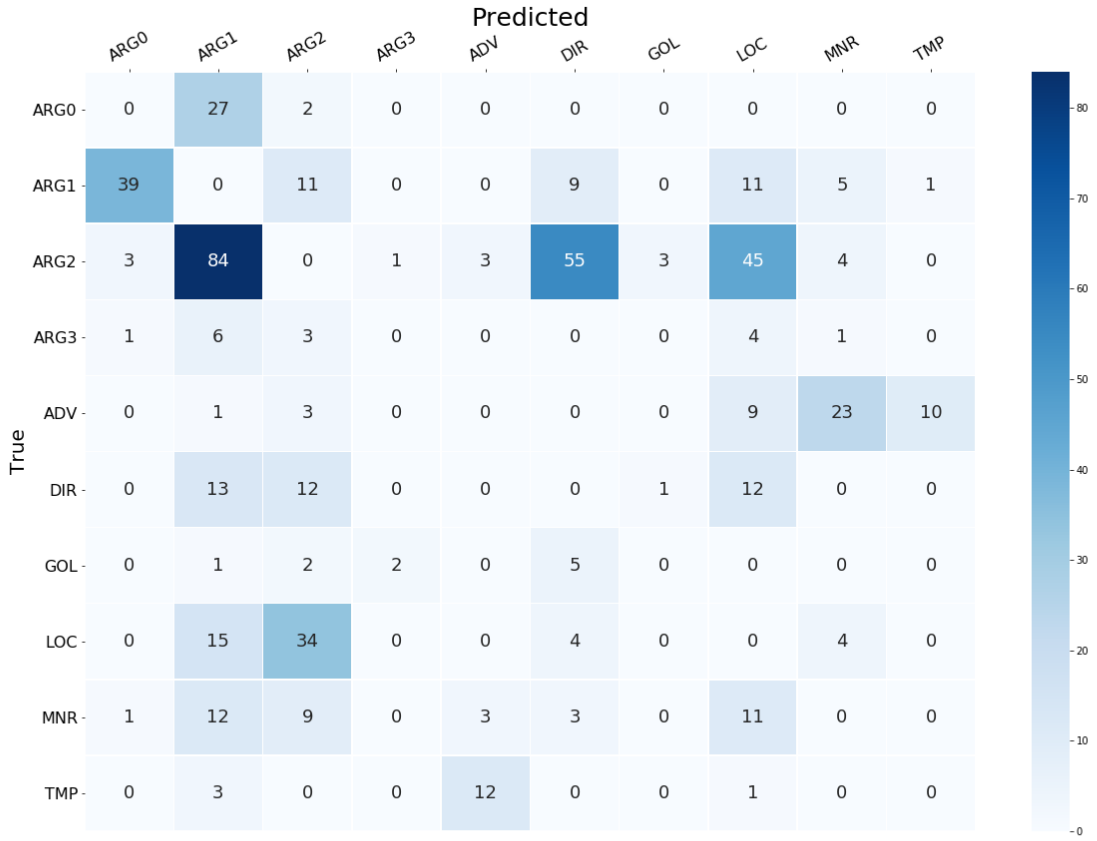


Figure 5.2: The confusion matrix for al-SRL, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j

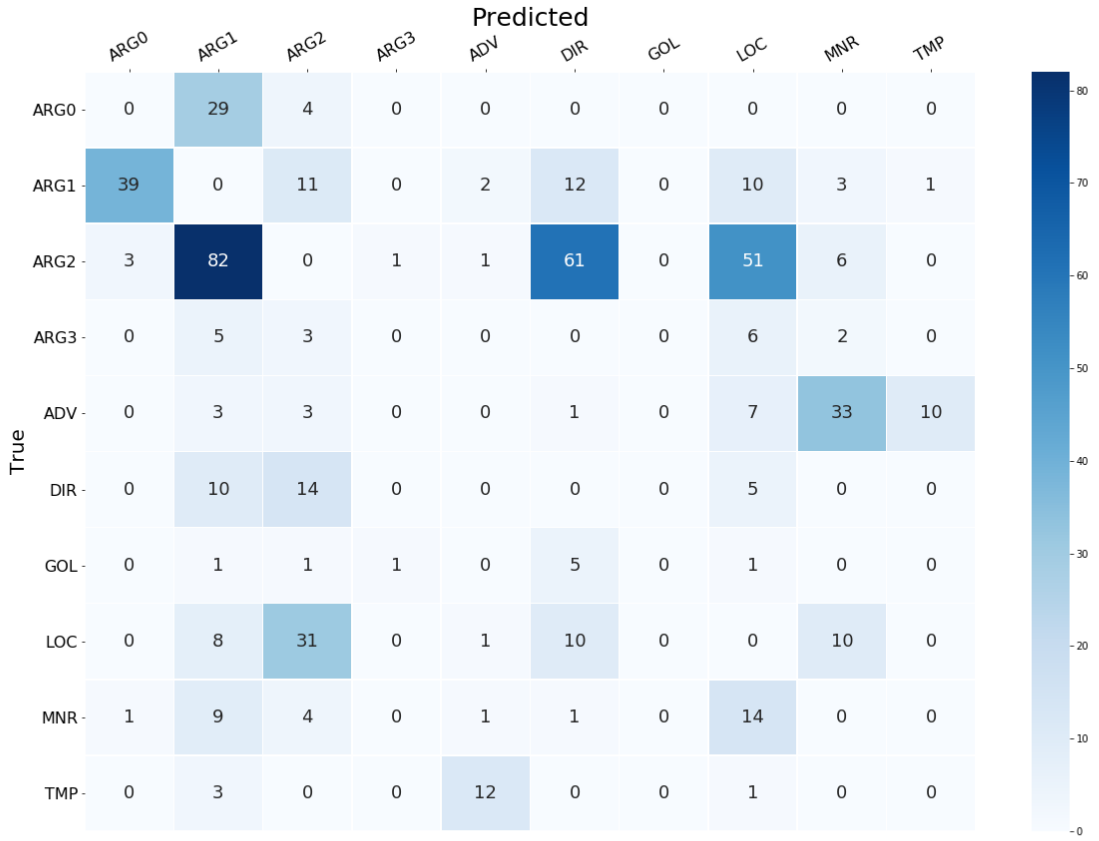


Figure 5.3: The confusion matrix for at-SRL. (note it is a simplified version of the complete confusion matrix.) Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j

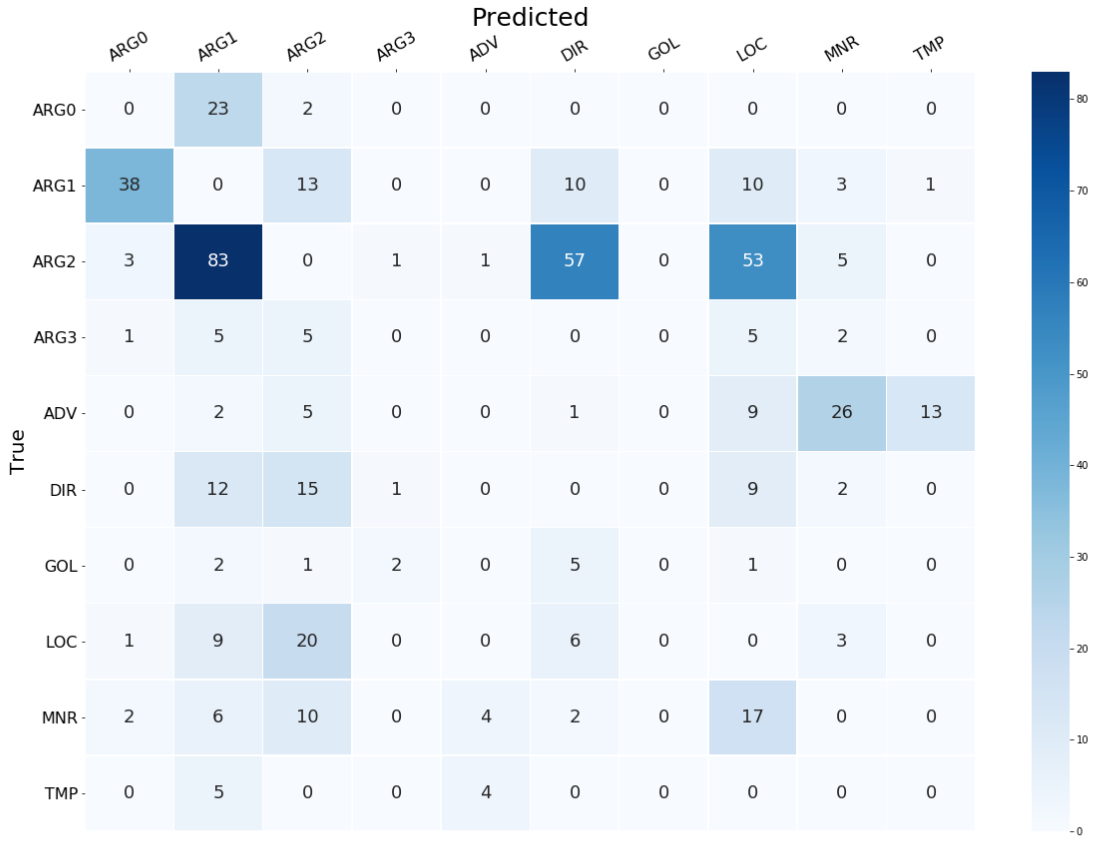


Figure 5.4: This heatmap represents the confusion difference between t-SRL and at-SRL without decoding. It is computed by subtracting the confusion matrix of at-SRL (no decoding) from t-SRL (no decoding). Blue color represents the confusion decrease for at-SRL compared with t-SRL; for example, $cell_{LOC,ARG2}$ represents at-SRL is less confused about LOC with ARG2 by 11. Furthermore, red color means the confusion increase for at-SRL compared with t-SRL.

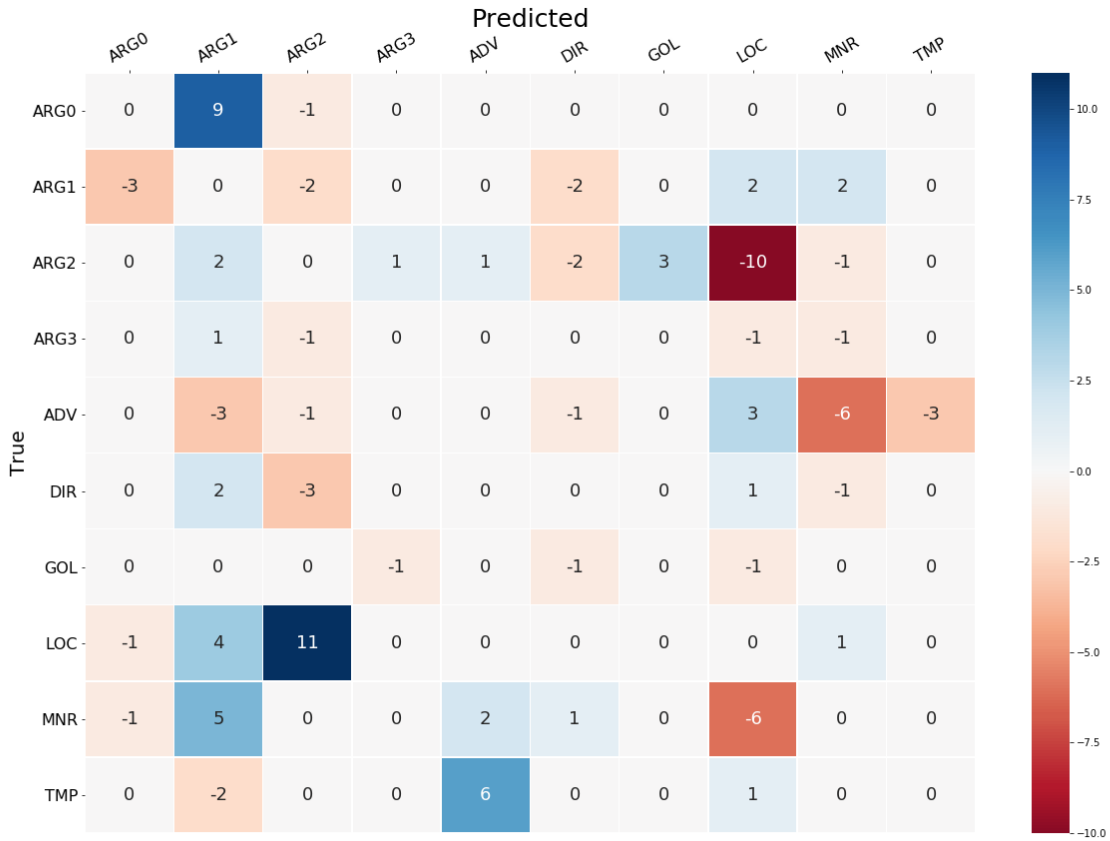


Figure 5.5: This heatmap represents the confusion difference between t-SRL and at-SRL with decoding. It is computed by subtracting the confusion matrix of at-SRL from t-SRL. Blue color represents the confusion decrease for at-SRL compared with t-SRL; for example, $cell_{LOC,ARG2}$ represents at-SRL is less confused about LOC with ARG2 by 14. Furthermore, red color means the confusion increase for at-SRL compared with t-SRL.

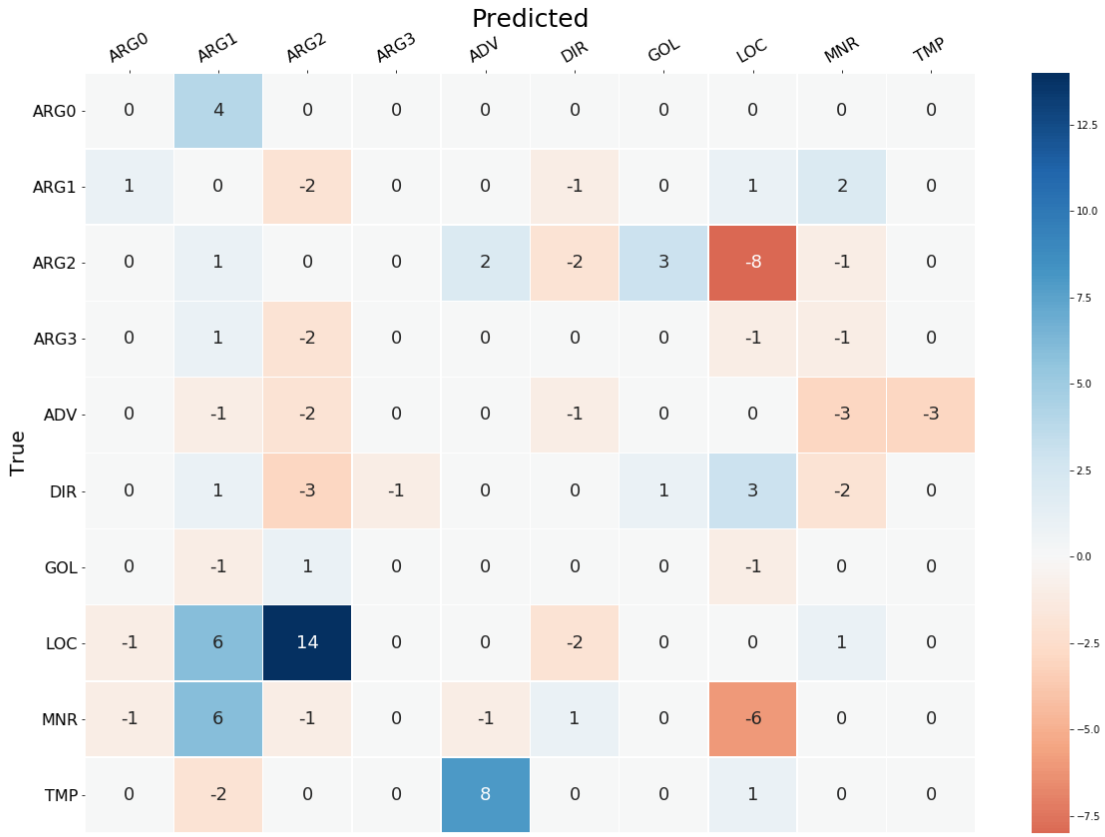


Table 5.8: This table gives the precision, recall and F1 score with respect to individual semantic role labels as well as overall results for at-SRL. It is the computed based on the predicted results **without** decoding (it is not a complete table)

	Precision	Recall	$F_{\beta=1}$
Overall	72.02%	74.69%	73.33
ARG0	84.40%	89.05%	86.66
ARG1	77.70%	84.53%	80.97
ARG2	70.10%	60.53%	64.96
ARG3	6.25%	4.35%	5.13
ARGM-ADV	42.07%	31.77%	36.20
ARGM-DIR	47.58%	59.60%	52.91
ARGM-GOL	16.67%	12.50%	14.29
ARGM-LOC	60.95%	73.16%	66.50
ARGM-MNR	36.42%	44.36%	40.00
ARGM-TMP	65.57%	76.56%	70.64

Table 5.9: Performance drop without decoder on precision, recall, F-1, for t-SRL, al-SRL, and at-SRL

Performance drop	Precision	Recall	F1
t-SRL	4%	0.87%	2.48
al-SRL	**4.97%**	**1.17%**	**3.15**
at-SRL	3.62%	0.78%	2.23

Figure 5.6: The confusion matrix for t-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j

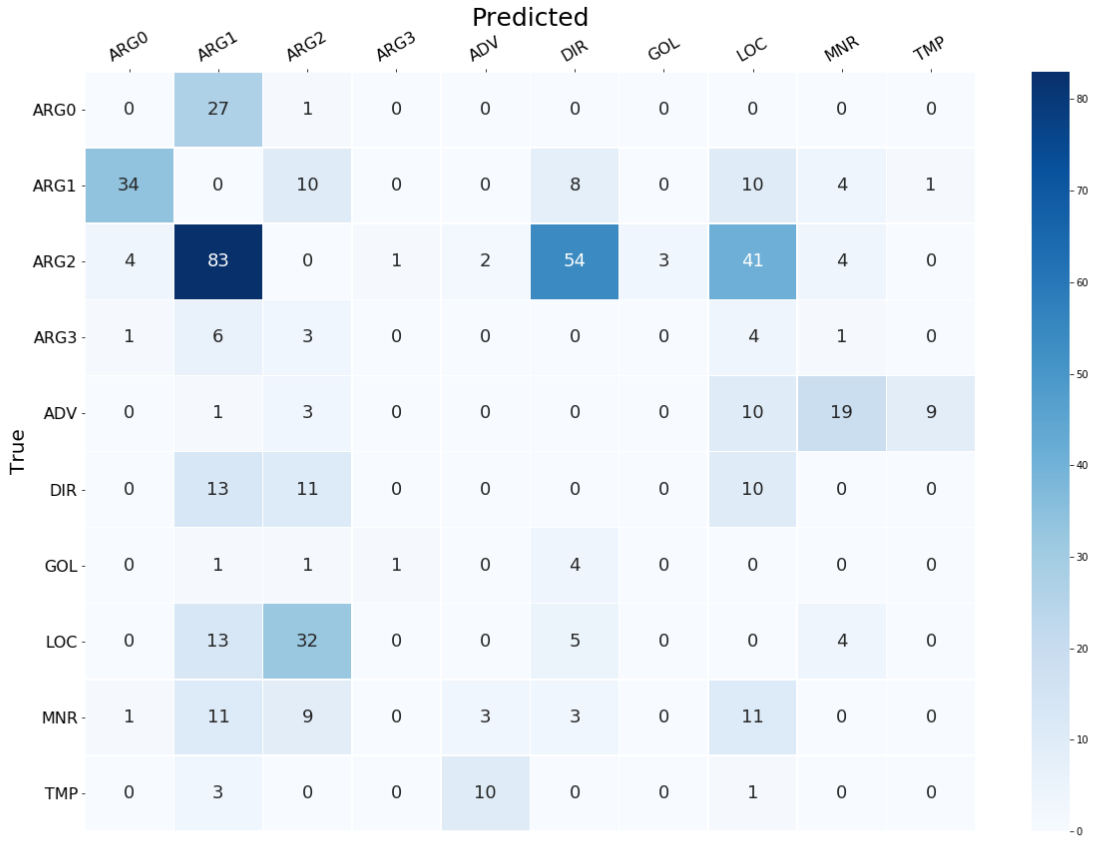


Figure 5.7: The confusion matrix for al-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j

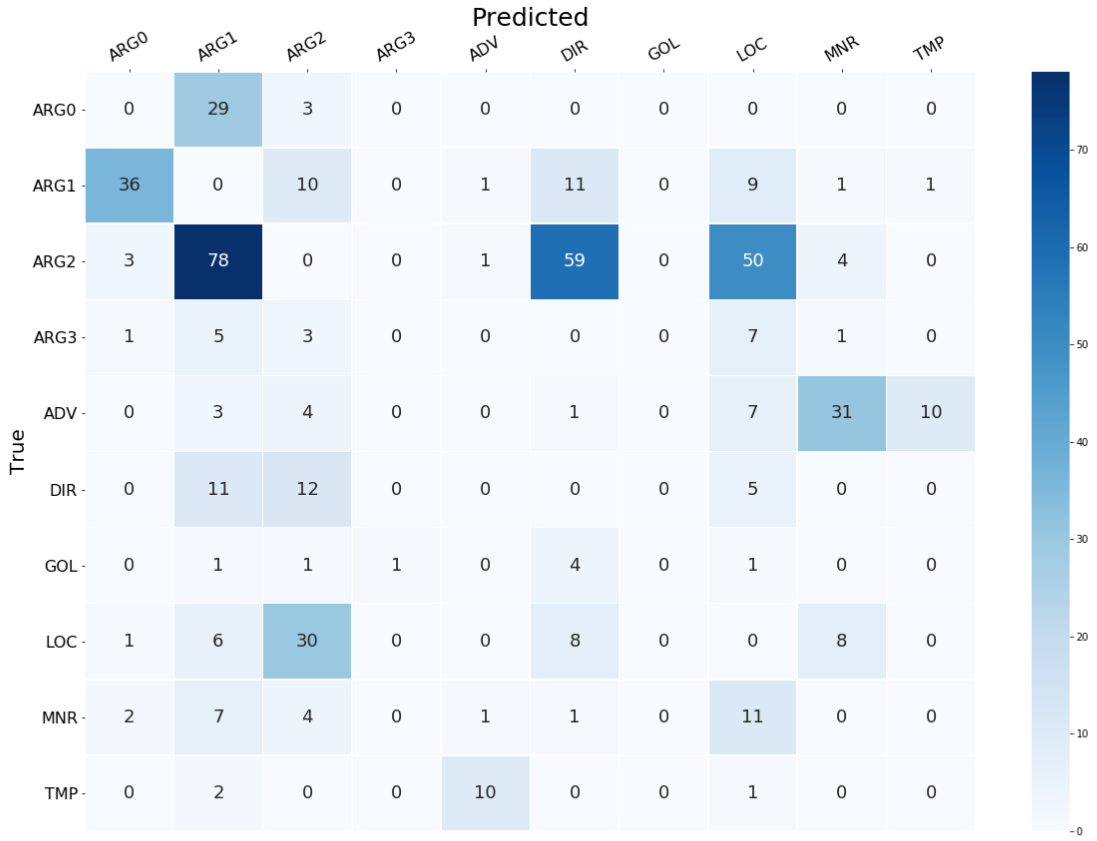
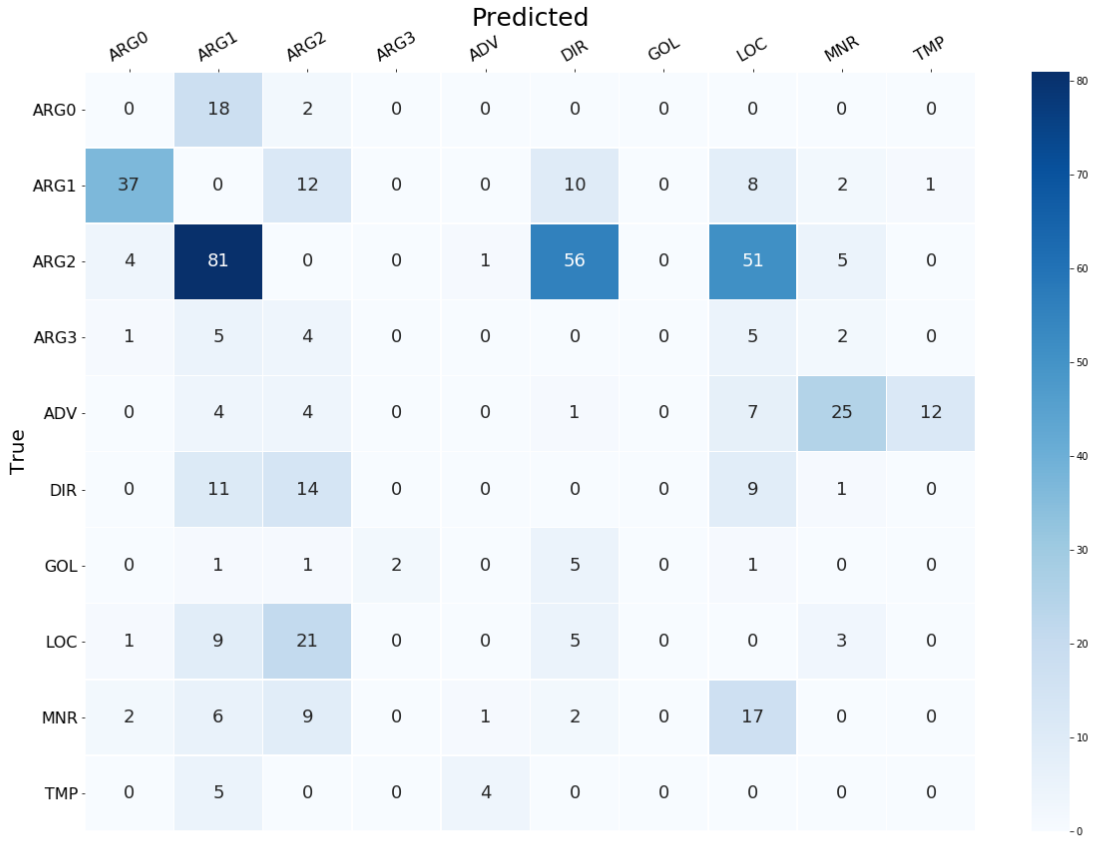


Figure 5.8: The confusion matrix for at-SRL **without** decoding, (note it is a simplified version of complete confusion matrix. Darker color denotes more mistakes, each $cell_{i,j}$ represents should be label i, but predicted as label j



Chapter 6

Conclusion and Future Work

This chapter concludes with a discussion of the results and further analysis that points to future directions.

6.1 Conclusion

This thesis presents three models that test the effectiveness of two ways of building multimodal representation towards incorporating image information into the SRL system. I created and obtained novel datasets for training and evaluation of SRL models. In addition, the model is evaluated on the out-of-domain dataset. I reported the results for t-SRL, al-SRL, at-SRL, and analyzed to compare across the performance of three models, and investigated how at-SRL achieved its performance. Specifically, first, from 5.1, we saw that at-SRL performs better than or at least comparable with the baseline t-SRL, and al-SRL performs consistently inferior to t-SRL or al-SRL. From ablation analysis 5.2.1, we saw the highway connection, decoder, contribute most to the at-SRL performance, but attention module also brought about 0.91 F1 increasing. Through comparing across confusion matrix, at-SRL show its superiority in that it confuses much less of LOC with ARG2, also confuses less of TMP with ADV compared with t-SRL. Then, combining the results from both reported metrics scores, the al-SRL is inferior among the three models. The attention module does not help resolve the inconsistency problem of BIO-tagging from current results from the results of the current experiment. 5.2.3

6.2 Future Work

The benefits of incorporating image information could be observed for at-SRL. Through analysis, at-SRL shows its superiority by confusing LOC less with ARG2. It tends to be more sensitive to location-related entities (therefore, label them correctly), which could be an important property of the proposed model as ARG2 is often confused with LOC and DIR in SRL systems, as pointed out by [He et al., 2017]. However, at the same time, with more location-related information, the model tends to confuse more of ARG2 and MNR with LOC. Therefore, one possibility for future work is to investigate why the phenomenon happens, and how to suppress the at-SRL's inclination to label ARG2, MNR with LOC. Also, I would like to: increase the training data size; currently, I am using the MSCOCO split defined in [Karpathy and Fei-Fei, 2017], but scale up the dataset is possible. Also, I would like to try the newly developed co-attention mechanism as well as use the transformer model to capture long-term dependency better. For al-SRL model, as described before, I believe the performance drop is largely due to the integrated loss function, next step, I would try to align image and caption implicitly without incorporating alignment loss directly into the objective function, also, develop a new model design for alignment model to utilize the bounding box features and align the objects with entities in the caption.

Currently[1] , limited by computing resources I have access to, each experiment instance is trained for 20 epochs, which is not sufficient enough comparing with reported experiment settings from some recent papers. They typically train the model for hundreds of epochs. Especially considering our at-SRL and al-SRL takes image data rather than just text, the model could require more training epoch to converge eventually. So it would be reasonable to expect that training for more epochs would bring about new insights and more robust results. Besides, this thesis project takes a unique training and evaluation combination: 1. the training dataset and evaluation dataset are coming from different data distributions. 2. The training dataset is automatically generated, and the evaluation dataset is ground truth dataset. To further investigate this experiment procedure,

[1] Many thanks to Professor James Martin, Professor Martha Palmer, Professor Chenhao Tan

a future direction could be controlling the amount of training dataset and analyze the relationship between the amount of auto-generated training dataset and the effectiveness of the model (measured by model performance). Finally, the assumption for this thesis project is that syntactic information could be extracted from the image. However, what type of syntactic information exists in image information? Currently, based on the analysis given above, I believe the spatial related information, like "LOC", "DIR", also the temporal information like "TMP" might be available in the image. Therefore, one assumption is that the static image contains the type of syntactic information that we could observe, but to induce the part of the information that depends on the interaction between entities, like "ARG0", "ARG1". We need to incorporate the "interaction" into our information resources, videos might be a possible resource. Furthermore, for the information that mostly resides in our mind, like syntactic information related to "goal", "intent". It requires a higher level of information understanding, so that we could obtain commonsense knowledge, and make causal inference; it requires more effective representation of our multimodal information resources, more data, even new tasks for encouraging models to acquire such capacity.

Bibliography

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In Proceedings of the IEEE international conference on computer vision, pages 2425–2433, 2015.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

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

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1, ACL '98/COLING '98, page 86–90, USA, 1998. Association for Computational Linguistics. doi: 10.3115/980845.980860. URL https://doi.org/10.3115/980845.980860.

Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41 (2):423–443, Feb 2019. ISSN 1939-3539. doi: 10.1109/tpami.2018.2798607. URL http://dx.doi.org/10.1109/TPAMI.2018.2798607.

Marzieh Bazrafshan and Daniel Gildea. Semantic roles for string to tree machine translation. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 419–423, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-2074.

Claire Bonial, Jena Hwang, Julia Bonn, Kathryn Conger, Olga Babko-Malaya, and Martha Palmer. English propbank annotation guidelines. Center for Computational Language and Education Research Institute of Cognitive Science University of Colorado at Boulder, 48, 2012.

Xavier Carreras and Lluís Màrquez. Introduction to the CoNLL-2004 shared task: Semantic role labeling. In Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004, pages 89–97, Boston, Massachusetts, USA, May 6 - May 7 2004. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W04-2412.

Xavier Carreras and Lluís Màrquez. Introduction to the conll-2005 shared task: Semantic role labeling. In Proceedings of the ninth conference on computational natural language learning (CoNLL-2005), pages 152–164, 2005.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

Maxwell Forbes, Ari Holtzman, and Yejin Choi. Do neural language representations learn physical commonsense?, 2019.

G. D. Forney. The viterbi algorithm. Proceedings of the IEEE, 61(3):268–278, 1973.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In Advances in neural information processing systems, pages 2121–2129, 2013.

Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks, 2015.

Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. Computational linguistics, 28(3):245–288, 2002.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the thirteenth international conference on artificial intelligence and statistics, pages 249–256, 2010.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task, pages 1–18, Boulder, Colorado, June 2009. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W09-1201.

Stevan Harnad. The symbol grounding problem. Physica D: Nonlinear Phenomena, 42(1-3):335–346, 1990.

Zellig S. Harris. Distributional structure. ¡i¿WORD¡/i¿, 10(2-3):146–162, 1954. doi: 10.1080/00437956.1954.11659520. URL https://doi.org/10.1080/00437956.1954.11659520.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. Deep semantic role labeling: What works and what's next. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 473–483, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1044. URL https://www.aclweb.org/anthology/P17-1044.

Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. Jointly predicting predicates and arguments in neural semantic role labeling. arXiv preprint arXiv:1805.04787, 2018.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. Ontonotes: The 90In Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, NAACL-Short '06, page 57–60, USA, 2006. Association for Computational Linguistics.

Dan Jurafsky. Speech & language processing. Pearson Education India, 2000.

Samira Ebrahimi Kahou, Xavier Bouthillier, Pascal Lamblin, Caglar Gulcehre, Vincent Michalski, Kishore Konda, Sébastien Jean, Pierre Froumenty, Yann Dauphin, Nicolas Boulanger-Lewandowski, et al. Emonets: Multimodal deep learning approaches for emotion recognition in video. Journal on Multimodal User Interfaces, 10(2):99–111, 2016.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4):664–676, Apr 2017. ISSN 2160-9292. doi: 10.1109/tpami.2016.2598339. URL http://dx.doi.org/10.1109/TPAMI.2016.2598339.

Andrej Karpathy, Armand Joulin, and Li F Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In Advances in neural information processing systems, pages 1889–1897, 2014.

C. Kong, D. Lin, M. Bansal, R. Urtasun, and S. Fidler. What are you talking about? text-to-image coreference. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 3558–3565, 2014.

Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel Gershman. Building machines that learn and think like people. The Behavioral and brain sciences, 40:e253, 2018.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. ArXiv, abs/1612.07182, 2017.

Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. Dependency or span, end-to-end uniform semantic role labeling. Proceedings of the AAAI Conference on Artificial Intelligence, 33:6730–6737, Jul 2019. ISSN 2159-5399. doi: 10.1609/aaai.v33i01.33016730. URL http://dx.doi.org/10.1609/aaai.v33i01.33016730.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. Lecture Notes in Computer Science, page 740–755, 2014. ISSN 1611-3349. doi: 10.1007/978-3-319-10602-1_48. URL http://dx.doi.org/10.1007/978-3-319-10602-1_48.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 11–20, 2016.

Diego Marcheggiani, Anton Frolov, and Ivan Titov. A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling. Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), 2017. doi: 10.18653/v1/k17-1041. URL http://dx.doi.org/10.18653/v1/K17-1041.

Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations, 2017.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank: An annotated corpus of semantic roles. Comput. Linguist., 31(1):71–106, March 2005. ISSN 0891-2017. doi: 10.1162/0891201053630264. URL https://doi.org/10.1162/0891201053630264.

Martha Palmer, Daniel Gildea, and Nianwen Xue. Semantic role labeling. Synthesis Lectures on Human Language Technologies, 3(1):1–103, 2010.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. International Journal of Computer Vision, 123(1):74–93, Oct 2016. ISSN 1573-1405. doi: 10.1007/s11263-016-0965-7. URL http://dx.doi.org/10.1007/s11263-016-0965-7.

Vasin Punyakanok, Dan Roth, and Wen-tau Yih. The importance of syntactic parsing and inference in semantic role labeling. Computational Linguistics, 34(2):257–287, 2008.

Michael Roth and Mirella Lapata. Neural semantic role labeling with dependency path embeddings. arXiv preprint arXiv:1605.07515, 2016.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks, 2013.

Karin Kipper Schuler and Martha S. Palmer. Verbnet: A Broad-Coverage, Comprehensive Verb Lexicon. PhD thesis, USA, 2005. AAI3179808.

Dan Shen and Mirella Lapata. Using semantic roles to improve question answering. In Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL), pages 12–21, 2007.

Chen Shi, Shujie Liu, Shuo Ren, Shi Feng, Mu Li, Ming Zhou, Xu Sun, and Houfeng Wang. Knowledge-based semantic embedding for machine translation. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2245–2254, 2016.

Haoyue Shi, Jiayuan Mao, Kevin Gimpel, and Karen Livescu. Visually grounded neural syntax acquisition. In ACL, 2019.

Peng Shi and Jimmy Lin. Simple bert models for relation extraction and semantic role labeling. arXiv preprint arXiv:1904.05255, 2019.

Carina Silberer and Mirella Lapata. Learning grounded meaning representations with autoencoders. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 721–732, 2014.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks, 2015a.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks, 2015b.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. Linguistically-informed self-attention for semantic role labeling. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018, pages 5027–5038. Association for Computational Linguistics, 2018. URL https://www.aclweb.org/anthology/D18-1548/.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019. doi: 10.18653/v1/p19-1644. URL http://dx.doi.org/10.18653/v1/p19-1644.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In Proceedings of the Twelfth Conference on Computational Natural Language Learning, CoNLL '08, page 159–177, USA, 2008. Association for Computational Linguistics. ISBN 9781905593484.

Hao Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In EMNLP/IJCNLP, 2019.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. Deep semantic role labeling with self-attention. In Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in neural information processing systems, pages 5998–6008, 2017.

Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. Machine comprehension with syntax, frames, and semantics. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 700–706, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2115. URL https://www.aclweb.org/anthology/P15-2115.

Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1386–1393, 2014.

Rui Wang, Hai Zhao, Sabine Ploux, Bao-Liang Lu, and Masao Utiyama. A bilingual graph-based semantic model for statistical machine translation. In IJCAI, pages 2950–2956, 2016.

Martin Wöllmer, Angeliki Metallinou, Florian Eyben, Björn Schuller, and Shrikanth Narayanan. Context-sensitive multimodal emotion recognition from speech and facial expression using bidirectional lstm modeling. In Proc. INTERSPEECH 2010, Makuhari, Japan, pages 2362–2365, 2010.

Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. In International conference on machine learning, pages 2397–2406, 2016.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In International conference on machine learning, pages 2048–2057, 2015.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1316–1324, 2017.

Matthew D. Zeiler. Adadelta: An adaptive learning rate method, 2012.

Y. Zhang, G. Chen, D. Yu, K. Yaco, S. Khudanpur, and J. Glass. Highway long short-term memory rnns for distant speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5755–5759, 2016.

Zhuosheng Zhang, Yuwei Wu, Zuchao Li, and Hai Zhao. Explicit contextual semantics for text comprehension, 2018.

Jie Zhou and Wei Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1127–1137, 2015.