Implementation and Evaluation of the DANCER Framework for Academic Article Summarization

Aathira Manoj
New York University
am10245@nyu.edu

Levente Szabo
New York University
ls5122@nyu.edu

Minji Kim
New York University
mk7773@nyu.edu

Abstract

Document summarization for academic articles is complicated by the document length, diversity and specialization of its vocabulary. The DANCER (Divide ANd ConquER) framework exploits the discourse structure of the document and uses sentence similarity to break a long document and its summary into multiple source-target pairs, which are used to train a summarization model. We assess the efficacy of the DANCER framework by using Pointer Generator as the summarization model and pairing it with different scoring metric (ROUGE-L, BLEU) for generating training samples. We evaluate its performance against a baseline model, which does not use DANCER. We use a subset of the publicly available arXiv dataset for our experiments. Code is available at https://github.com/ls5122/ArxivSummary.

1 Introduction

Automatic summarization has been recognized as one of the most important natural language processing tasks, as information in the modern digital age is very enormous requiring well designed storage, retrieval and compression. Being able to produce informative and well-written document summaries has the potential to greatly improve the success of both information discovery systems and human readers that are trying to quickly skim large numbers of documents for important information.

Therefore, in previous years, there are various and plenty of progress in neural summarization, but the majority of them have been focusing on short pieces of text such as news articles (CNN/Daily Mail dataset (Hermann et al., 2015)), for which input texts are on average 600 words long and abstracts are less than 100 words long. Summarization of longer documents, such as research papers, is very different from newswire summarization, with input text ranging from 2,000 to 7,000 words and the summaries exceeding 200 words. Seq2seq models tend to struggle with longer sequences because at each decoding step, the decoder needs to learn to construct a context vector capturing relevant information from all the tokens in the source sequence (Sha et al., 2016) leading to much higher noise.

The Divide-ANd-ConquER (DANCER) (Gidiotis and Tsoumakas, 2020) framework exploits the discourse structure of long documents, by working on each section separately. It does so by decomposing the summary of the document into sections and pairing it with the appropriate sections in order to create distinct target summaries. A neural network model is then trained to learn to summarize each part of the document separately using these target summaries. Finally, the partial summaries generated by the model are then combined to produce a final complete summary. This leads to a decomposition of the long document summarization problem into simpler summarization problems that are easier to learn and is less computationally complex and in the process creating more training examples having lesser noise in the target summaries when compared to the standard approach.

Academic articles are an ideal candidate for long document summarization because the paper abstract can serve as a label or a candidate summary for the paper. Additionally many papers follow a general structure with Introduction, Methods, Results and Conclusion being common sections in most articles. We can improve the summarization output and reduce computational costs by taking advantage of this common structure.

2 Related Works

The majority of past work in text summarization has been extractive (Aakash Sinha, 2018),

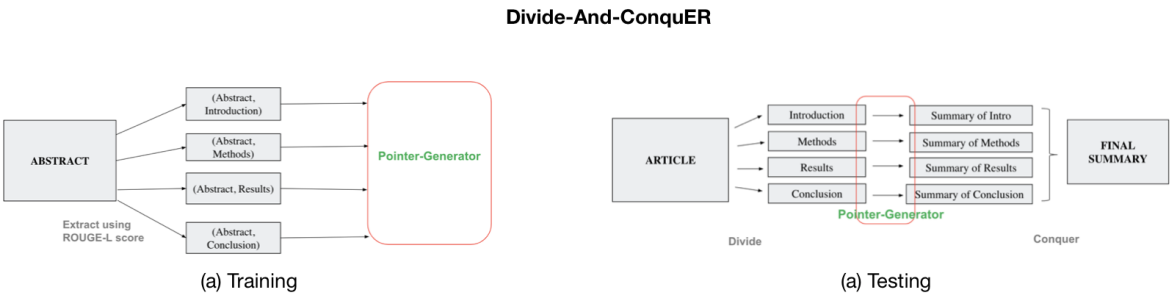

Figure 1: (a) Generating samples for training DANCER (b) Generating summary using DANCER

(Yong Zhang, 2016) where systems form summaries by copying and rearranging passages from the original text. Abstractive summarization where novel text is generated is a more challenging task with a number of advances so far. These include abstract summarization under DANCER (Gidiotis and Tsoumakas, 2020), BERT (Yang Liu, 2019), Pointer Generator Networks (See et al., 2017). For an overall review of extractive and abstractive summarization, see (Mehta, 2016).

3 Methods

3.1 Problem Statement

Our goal is to evaluate and compare the performance of DANCER by using different scoring metrics for splitting the document summary into different sections. In particular, we use ROUGE-L (Lin, 2004) and BLEU (Papineni et al., 2002) to generate the section-wise summaries from the main summary for training. We use the Pointer Generator, based on the sequence-to-sequence RNN paradigm, as the main summarization model. Finally, we compare the performance of the DANCER model with a baseline model, which also uses Pointer Generator, but does not split the main summary into section-wise summaries. We use a subset of the publicly available pre-processed ArXiv dataset (Cohan et al., 2018) for our experiments. The performance of different models is evaluated using ROUGE scores.

3.2 Approach

A general outline of our approach is given below.

1. Use the DANCER method to generate training datasets using the ROUGE-LCS score and the BLEU-Corpus score. Generate a baseline dataset which does not use this approach.

2. Train Pointer Generator Seq2Seq summarizer separately on all these datasets.

3. While testing, use the trained weights of the Pointer Generator model to generate section-wise summaries and concatenate them to generate the final summary .

4. Evaluate the different models according to different ROUGE scores.

DANCER: The steps for training using the DANCER framework are outlined in Figure 1. During training the DANCER approach requires some data pre-processing steps. We wish to divide the abstract (gold label summary) into summaries for the Introduction, Literature, Methods, Results and Conclusion. Since the section names of most academic articles do not line up perfectly with our desired sections our first step is to try to label and potentially merge the article sections.

To do this we use simple keyword matching by labelling each section according to the maximal matches from a fixed set of keyword for each of our desired sections. Keywords used are shown in Table 1. If a section name does not match any of the keyword then we skip it. Additionally, any sections with the same label are merged.

Once we have isolated our desired sections we need to partition the abstract into multiple summary labels (one for each section). To do this we focused on two specific metrics, the ROUGE-LCS score and the BLEU-Corpus level score. Between each sentence of the abstract a_i and each sentence of a specific section s_j^k The ROUGE-LCS is the length of the longest common subsequence divided by the length of the the section sentence.

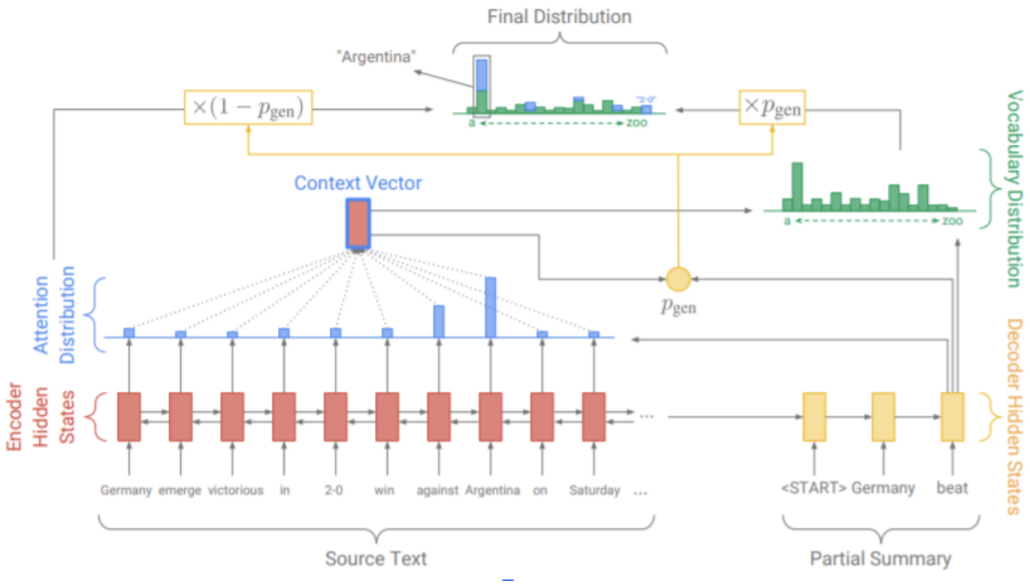

Figure 2: Pointer Generator Summarization Model

$$\textit{ROUGE-LCS}(a_i, s_j^k) = \frac{|LCS(a_i, s_j^k)|}{|s_j^k|}$$

We assign summary sentence a_i to the section s^k* where the section contains the sentence producing the maximal ROUGE-LCS precision.

The BLEU-Corpus level score consists of calculating n-gram level precisions over multiple sentences. For every sentence in the abstract we calculate its BLEU score for each section. The BLEU score uses n-gram precision scores and corresponding weights along with a penalty term to compute the final score.

The ROUGE and BLEU scores produce different splits in the abstract (Figure 3). While a majority of the abstract sentences are assigned to the Introduction there is some variation in how other sections are represented, notably using the BLEU score resulted in lower counts for the Introduction. To produce more coherent results we excluded the Literature section.

Pointer Generator: The Pointer Generator is a sequence to sequence neural model that provides abstract text summarization (Figure 2). The model consists of an encoder and decoder phase. Through a combination of pointing at words in the source and generating words from the vocab distribution this model can be seen as a hybrid summarization model. The model uses a coverage mechanism to minimize the repetition of copied words.

The input source text is embedded and fed into a bidirectional LSTM, the red network in Figure 2. This network serves as the encoder and produces a set of hidden states h_i. Decoding takes place one token at a time. For each timestep the decoder gives rise to decoder states s_t which during training correspond to the word embedding of the previous word. Together these combine to create the attention distribution using the parameters W_h, W_s, b_{attn}, v

$$e_i^t = v^T \textit{tanh}(W_h h_i + W_s s_t + b_{attn}) \quad (1)$$

$$a^t = \textit{softmax}(e^t) \quad (2)$$

Next using the attention distribution the context vector h_t^* is constructed as a dot product between the attention distribution and the hidden states. The attention distribution guides the decoder towards the next word while the context vector serves to produce the distribution over all words in the vocabulary p_{vocab} as well as the pointer generator probability p_{gen} which gives us the option to copy words from the source text.

By concatenating the context vector with the decoder state s_t we get p_{vocab}

$$p_{vocab} = \textit{softmax}(V'(V[s_t, h_t^*] + b)b') \quad (3)$$

With V', V, b, b' all parameters. The pointer generator takes as input the context vector and the decoder states and decoder inputs. The parameters are $w_h^T, w_s^T, w_x^T, b_{ptr}$

Section	Keywords
introduction	introduction , case
literature	background , literature , related
methods	method , methods , technique , techniques , methodology
results	result , results , experimental , experiment , experiments
conclusion	conclusion , conclusions , concluding , discussion , summary , limitations

Table 1: Section Keywords

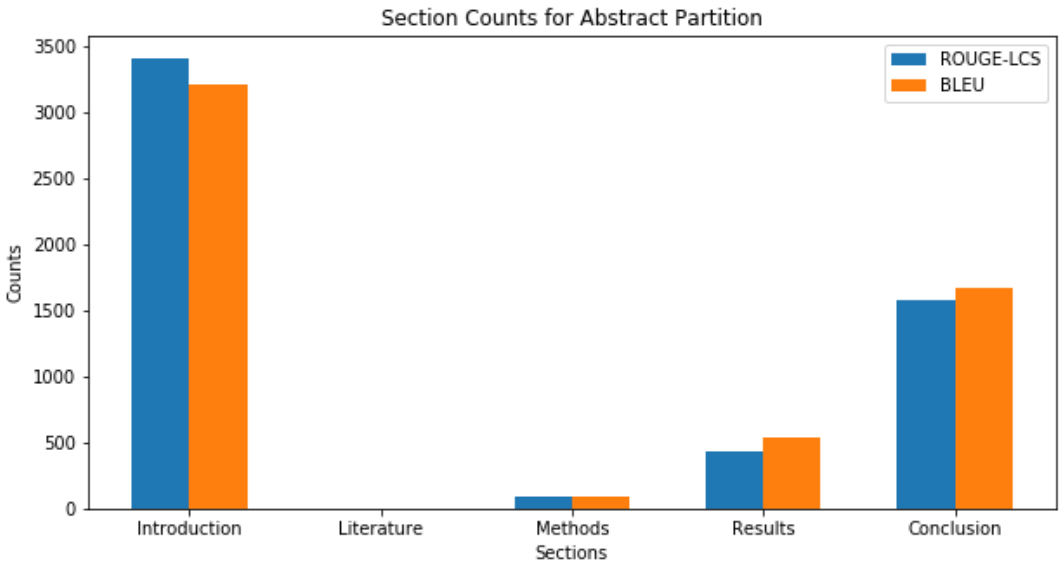

Figure 3: Partition of the document abstract into different sections

$$p_{gen} = \sigma(w_h^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr}) \quad (4)$$

The source distribution of a word is computed $p_{source}(w)$, this allows us to select ideal word to copy from the source text according to the attention distribution. It is a result of summing the attention distribution over all words $w_i = w$. Finally the decoder, the yellow network in Figure 2 produces a token according to

$$p_(w) = p_{gen} \cdot p_{vocab} + (1 - p_{gen}) \cdot p_{source} \quad (5)$$

Decoding During the testing phase the DANCER framework applies the trained Pointer Generator summarizer in parallel to the article sections. Then the section summaries are concatenated to construct the final article summary which is tested against the abstract along a variety of metrics to determine the effectiveness of the summary.

4 Experimental Setup

4.1 Dataset

We use structured long documents from the arXiv, a free distribution service for scholarly articles. For training and testing, we use a subset of pre-processed, publicly available arXiv documents. provided courtesy of Cohan et al. (Cohan et al., 2018). The entire document set is 7.07 GB and contains 215,913 documents in addition to a vocabulary file. We use a subset of 8000 articles for training and 1000 articles for validation and testing. Each document contains the abstract, the article and a list of section names.

We have grouped similar sections together by the keywords used to describe them, as shown in Table 1. All sections do not equally contribute towards the final summary, as shown in Figure 3. Usually, the literature section is not essential when trying to summarize the main points of the article. On the other hand, sections like the introduction and conclusion usually include quite a lot of the important information that we want to include in

Model	ROUGE-1	ROUGE-2	ROUGE-L
DANCER with ROUGE-L and Ptr Gen	**37.01**	**12.77**	**21.21**
DANCER with BLEU and Ptr Gen	34.78	12.30	20.44
Baseline	0.62	0.001	0.60

Table 2: ROUGE F1 results for different models.

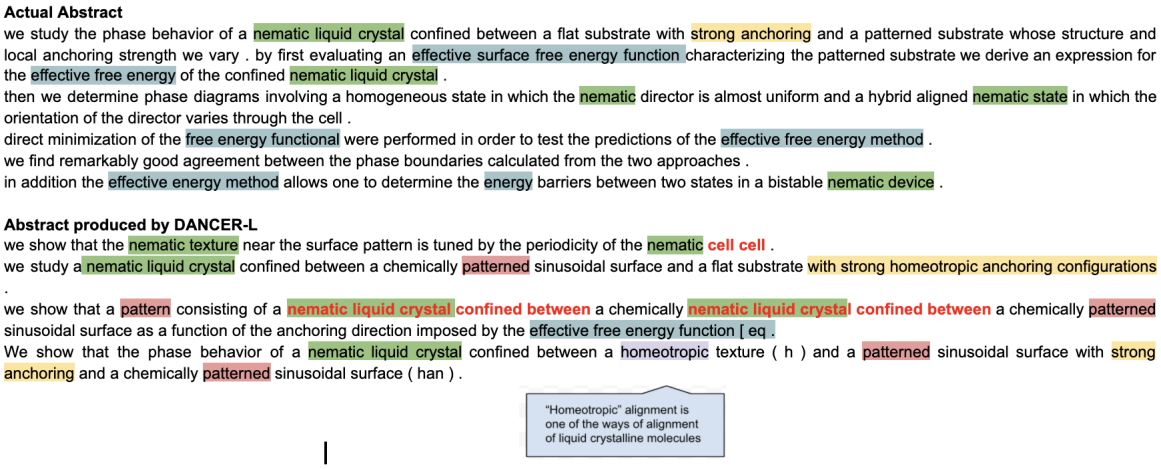

Figure 4: Example abstract generated using DANCER framework

the summary. Therefore, we have used only those sections while generating the training samples as well as during testing.

4.2 Implementation Details

All the 3 datasets were trained using Pointer Generator model.Our Pointer-Generator model is implemented in Pytorch and is based on the original Tensorflow implementation (See et al., 2017). The hyperparameter selection is similar to the setup suggested in (See et al., 2017). Our model has a bidirectional LSTM layer of 256 units for the encoder and a unidirectional LSTM layer of 256 units for the decoder. We restrict the vocabulary to 50,000 word tokens for both the input and output and use word embeddings of size 128. No pre-trained word-embeddings are used. A batch size of 16 is used for training. We used *Adagrad* (Duchi et al., 2011) with 0.15 learning rate and initialize the accumulator to 0.1. We clip the gradients to have a maximum norm of 2, but avoid using any regularization.

We trained all the three models for $50,000$ epochs with coverage turned on after $40,000$. For training, input sequences are truncated to 400 word tokens while padding the shorter ones with zeros to the same length. The decoder outputs were restricted to 100 words. For the prediction phase, we use beam search decoding with 4 beams and generate a maximum of 100 tokens per section.

The models were evaluated using ROUGE (Lin, 2004) F1 scores. The results are can be seen in Table 2.

5 Results

Based on the results shown in Table 2, it is evident that DANCER provides a significant improvement in ROUGE F1 scores. Generating section-wise summaries using ROUGE-L for training produces better ROUGE F1 results than by generating section-wise summaries using BLEU scores. The summaries produced by both the metric are more similar to each other than to the actual summary.

The summary generated by DANCER as shown in Figure 4, after training for $50,000$ epochs on just 8000 samples using ROUGE-L to generate section is quite impressive. It manages to generate a lot of words occurring in the actual abstract e.g. *"nematic liquid crystals", "strong anchoring", "effective free energy", "patterned"* etc. It also generates words not in the actual abstract e.g. *"homeotropic"*, which is one of the ways of alignment of liquid crystal molecules, seemingly relevant to the topic of the research article. The word is not a common one and not part of the vocabulary and must be copied from the article. We suspect the *pointer*

mechanism to be responsible for it.

However we also see some repetitions (marked in red) e.g *"cell cell", "nematic liquid crystal"* . We suspect the lack of training with the coverage turned on (only 10,000 epochs) to be the cause of this.

Refer to Appendix A for example summaries produced when different metrics (ROUGE-L and BLEU) are used with DANCER.

6 Conclusion and Future Work

Summarizing long documents is a very different problem to newswire summarization, requiring much higher computational complexity, making it extremely hard to train models that have enough capacity to perform this task. DANCER is a simple yet effective extension that can boost the performance of different summarization models with minimal additional effort and resources. DANCER framework allows different methods to be used for section-wise summary generation for training and can be used with various summarization models. In this paper, we have evaluated its performance when using ROUGE-L and BLEU for generating training samples and Pointer Generator section-wise summarization against a baseline model. Based on our experiments, we observed a notable improvement in ROUGE F1 scores when using DANCER. ROUGE-LCS produces better ROUGE F1 scores than BLEU score.

In future work, we would like to:

- Try more complex summarization models like Transformers based models. It can also be combined with other sophisticated methods that perform sentence extraction before the main summarization process, since it has been observed that pointer neural networks sometimes struggle at selecting relevant parts of the input.
- Exploring more sophisticated methods that use machine learning to identify the type of each section. These methods become more relevant when not dealing with academic documents and needs to be extended to other domains.
- Use a machine learning model to make the decision if a given section should be included in the summary

References

Akshay Gahlot Aakash Sinha, Abhishek Yadav. 2018. Extractive text summarization using neural networks.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. 12(null):2121–2159.

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of long documents.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28, pages 1693–1701. Curran Associates, Inc.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Parth Mehta. 2016. From Extractive to Abstractive Summarization: A Journey. pages 100 – 106, Berlin, Germany. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. ACL '02, page 311–318, USA. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks.

Lei Sha, Jing Liu, Chin-Yew Lin, Sujian Li, Baobao Chang, and Zhifang Sui. 2016. RBPB: Regularization-based pattern balancing method for event extraction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1224–1234, Berlin, Germany. Association for Computational Linguistics.

Mirella Lapata Yang Liu. 2019. Text summarization with pretrained encoders.

Mahardhika Pratama Yong Zhang, Joo Er Meng. 2016. Extractive document summarization based on convolutional neural networks. IEEE.

A Appendices

Examples of Generated Summaries

Original Abstract	we show how to control spatial quantum correlations in a multimode degenerate optical parametric oscillator type i below threshold by introducing a spatially inhomogeneous medium , such as a photonic crystal , in the plane perpendicular to light propagation .we obtain the analytical expressions for all the correlations in terms of the relevant parameters of the problem and study the number of photons , entanglement , squeezing , and twin beams . considering different regimes and configurations we show the possibility to tune the instability thresholds as well as the quantumness of correlations by breaking the translational invariance of the system through a photonic crystal modulation .
DANCER with ROUGE-L summary	we show that the periodic modulation of the refractive index leads to the formation of the refractive index in the single photon regime .we show that the presence of the periodic modulation of the refractive index leads to the origin of the electromagnetic field .we study the effect of an intracavity pc in a typical device displaying quantum light spatially correlated in continuous variables .we show that in presence of a deep translational symmetry in the instability process the scenario is lowered .
DANCER with BLEU summary	the degrees of freedom where quantum correlations between components of a system separated in a optical parametric oscillator (opo) are considered in semiconductor microcavities .the quantum - enhanced modulation of the refractive index leads to the formation of the photonic band - gap .we study the effect of an intracavity pc in a typical device displaying quantum light spatially correlated in continuous variables , as well as with respect to the threshold can improve correlations at a given pump with respect to the opo , at least when threshold is lowered (pump modulation) .

Original Abstract	the production of @xmath0 bosons in association with jets is an important background to new physics at the lhc .events in which the @xmath0 carries large transverse momentum and decays leptonically lead to large missing energy and are of particular importance .we show that the left - handed nature of the @xmath0 coupling , combined with valence quark domination at a @xmath1 machine , leads to a large left - handed polarization for both @xmath2 and @xmath3 bosons at large transverse momenta .the polarization fractions are very stable with respect to qcd corrections .the leptonic decay of the @xmath4 bosons translates the common left - handed polarization into a strong asymmetry in transverse momentum distributions between positrons and electrons , and between neutrinos and anti - neutrinos (missing transverse energy) .such asymmetries may provide an effective experimental handle on separating @xmath0+jets from top quark production , which exhibits very little asymmetry due to c invariance , and from various types of new physics ..3 cm"
DANCER with ROUGE-L summary	we study one - to - leading order (nlo) matrix emission in top - quark pair production in short - distance standard - model interactions .we calculate the kinds of distinguish different sources pushes directly in an electroweak process .we show that the unambiguous identification of a recoil partner in the splitting process is exactly mapped onto emitter and spectator partons , are shifted in the assignment of a common renormalization and factorization scale .we study the polarization dominance of the electroweak charged - current interaction in the @xmath1 initial state (lhc) @xmath0 vector bosons of both signs in the partonic center of the @xmath1 initial state .
DANCER with BLEU summary	the large - handed polarization of high-@xmath6 @xmath0 bosons moving primarily along the beam axis and @xmath3 vector bosons produced directly in short - distance standard - model interactions are presented .the effect is distinct from the well - known @xcite - leading order cross sections .we study the polarization coefficients of both nlo parton - level qcd and the me+ps framework of the phase - space integration .the polarization fractions are quite insensitive to variations of a common renormalization and factorization scale , so varying the scale does not offer a reasonable estimate of the theoretical uncertainty .we presented a detailed study of the polarization dominance of the electroweak charged - current interaction in the @xmath1 initial state of the electroweak charged - current interaction .the prevalence of valence quarks in the @xmath1 initial state of the dominant process are discussed .

Original Abstract	we consider the type i see - saw model with two right - handed neutrinos and a normal neutrino mass hierarchy and impose a zero coupling between the right - handed neutrino mainly responsible for the atmospheric neutrino mass and the electron neutrino .we derive a master formula which relates see - saw input parameters in a one to one correspondence with physical neutrino observables . using the master formula we search for simple ratios of couplings consistent with current data on neutrino mass and lepton mixing .we discover a minimal predictive example in which the right - handed neutrino mainly responsible for the atmospheric neutrino mass has couplings to @xmath0 proportional to @xmath1 and the right - handed neutrino mainly responsible for the solar neutrino mass has couplings to @xmath0 proportional to @xmath2 or @xmath3 , with a relative phase @xmath4 , providing the link between leptogenesis and cp violation in neutrino oscillation experiments .we show how these patterns of couplings could arise from an @xmath5 family symmetry model of leptons which predicts all the pmns parameters in terms of the neutrino mass ratio @xmath6 , corresponding to tri - bimaximal - cabibbo mixing , accurate to one degree , with the prediction @xmath7 .
DANCER with ROUGE-L summary	we present a detailed study of the statistics of a sequential dominance (sd) neutrino yukawa couplings in the heavy right - handed majorana neutrino masses .in particular , we find that the right - handed neutrino mass @xmath10 and the mass of the lightest neutrino mass @xmath12 and the mass of the lightest neutrino mass matrix with respect to the normal neutrino mass matrix and the mass hierarchy of the lightest neutrino mass .we show that sequential dominance follows automatically and a second texture zero yukawa coupling of the electron neutrino to the measurement of the minimal see - saw model which relates see - saw parameters in this case .it is shown that the bound on the reactor angle @xmath287 is saturated by the measurement of a minimal see - saw model with normal neutrino mass hierarchy consisting of two right - handed neutrinos with a zero yukawa coupling in both reactor type i see - saw model with hierarchical strength .
DANCER with BLEU summary	it is caused by the effective neutrino mass matrix from the see that the lightest neutrino mass matrix with hierarchical strength , leading to a normal mass hierarchy of physical neutrino masses .the neutrino of mass @xmath11 is mainly responsible for the lightest neutrino mass .we study the effective neutrino model with normal neutrino mass hierarchy consisting of two right - handed neutrinos with a zero yukawa coupling of the `` dominant '' right - handed neutrino to the electron neutrino mass matrix with hierarchical strength , leading to the limiting case of a three right - handed neutrino model with sequential dominance .