Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

TRANSFER LEARNING: BRIDGING THE GAP BETWEEN DEEP LEARNING AND DOMAIN-SPECIFIC TEXT MINING

by
Chaoran Cheng

Inspired by the success of deep learning techniques in Natural Language Processing (NLP), this dissertation tackles the domain-specific text mining problems for which the generic deep learning approaches would fail. More specifically, the domain-specific problems are: (1) success prediction in crowdfunding, (2) variants identification in biomedical literature, and (3) text data augmentation for domains with low-resources.

In the first part, transfer learning in a multimodal perspective is utilized to facilitate solving the project success prediction on the crowdfunding application. Even though the information in a project profile can be of different modalities such as text, images, and metadata, most existing prediction approaches leverage only the text modality. It is promising to utilize the visual images in project profiles to find out how images could contribute to the success prediction. An advanced neural network scheme is designed and evaluated combining information learned from different modalities for project success prediction.

In the second part, transfer learning is combined with deep learning techniques to solve genomic variants Named Entity Recognition (NER) problems in biomedical literature. Most of the advanced generic NER algorithms can fail due to the restricted training corpus. However, those generic deep learning algorithms are capable of learning from a canonical corpus, without any effort on feature engineering. This work aims to build an end-to-end deep learning approach to transfer the domain-specific knowledge to those advanced generic NER algorithms, addressing the challenges in low-resource training and requiring neither hand-crafted features nor post-processing rules.

For the last part, transfer learning with knowledge distillation and active learning are utilized to solve text augmentation for domains with low-resources. Most of the recent text augmentation methods heavily rely on large external resources. This work is dedicates to solving the text augmentation problem adaptively and consistently with minimal resources for token-level tasks like NER. The solution can also assure the reliability of machine labels for noisy data and can enhance training consistency with noisy labels.

All the work are evaluated on different domain-specific benchmarks, respectively. Experimental results demonstrate the effectiveness of those methods. The advantages also indicate promising potential for transfer learning in domain-specific applications.

TRANSFER LEARNING: BRIDGING THE GAP BETWEEN DEEP LEARNING AND DOMAIN-SPECIFIC TEXT MINING

by
Chaoran Cheng

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Science

Department of Computer Science

May 2020

APPROVAL PAGE

TRANSFER LEARNING: BRIDGING THE GAP BETWEEN DEEP LEARNING AND DOMAIN-SPECIFIC TEXT MINING

Chaoran Cheng

Dr. Zhi Wei, Dissertation Advisor — Date
Professor of Computer Science, NJIT

Dr. James M. Calvin, Committee Member — Date
Professor of Computer Science, NJIT

Dr. James Geller, Committee Member — Date
Professor of Computer Science, NJIT

Dr. Senjuti Basu Roy, Committee Member — Date
Assistant Professor of Computer Science, NJIT

Dr. Wenge Guo, Committee Member — Date
Associate Professor of Mathematical Sciences, NJIT

BIOGRAPHICAL SKETCH

Author: Chaoran Cheng

Degree: Doctor of Philosophy

Date: May 2020

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Science,
  New Jersey Institute of Technology, New Jersey, US, 2020
- Master of Science in Computer Science,
  Beijing Forestry University, Beijing, China, 2012
- Bachelor of Science in Computer Science,
  Beijing Forestry University, Beijing, China, 2009

Major: Computer Science

Presentations and Publications:

Chaoran Cheng, Cheng Zhong, Jie Zhang, and Zhi Wei. "Consistency-based Unsupervised Data Augmentation for Named Entity Recognition with Minimal Resources." *under review.*

Chaoran Cheng, Fei Tan, and Zhi Wei. "DeepVar: An End-to-End Deep Learning Approach for Variants Identification in Biomedical Literature ." *In the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI), 2020.*

Chaoran Cheng, Fei Tan, Xiurui Hou, and Zhi Wei. "Success Prediction on Crowdfunding with Multimodal Deep Learning." *In the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, 2019.

Fei Tan, Chaoran Cheng, and Zhi Wei. "Modeling and Elucidation of Housing Price." *Data Mining and Knowledge Discovery (DMKD)*, 33.3 pp. 636-662. 2018.

Fei Tan, Chaoran Cheng, and Zhi Wei. "Time-aware Latent Hierarchical Model For Predicting house prices." *IEEE International Conference on Data Mining (ICDM)*, pp. 1111-1116, 2017.

Fei Tan, Chaoran Cheng, and Zhi Wei. "Modeling Real Estate for School District Identification" *IEEE International Conference on Data Mining (ICDM)*, pp. 1227-1232, 2016.

Kai Zhang, Shandian Zhe, Chaoran Cheng, Zhi Wei, Zhengzhang Chen, Haifeng Chen, Guofei Jiang, Yuan Qi, and Jieping Ye. "Annealed Sparsity via Adaptive and Dynamic Shrinking" *In the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1325-1334. ACM, 2016.

To my parents and Lifeng, for their unconditional love.
To my dear JJ and Jenny, I love you.

ACKNOWLEDGMENT

The past five years at NJIT have been an unforgettable experience for me. When I first started my Ph.D. in 2014 with limited software engineering experience, I had never heard of the term "deep learning" and knew very little about machine learning. The word "data science" was not cool in public media yet. Back then, I could barely understand how to build an artificial intelligence model to understand human language. It is unbelievable that I have been doing research about deep learning over the last couple of years and developing state-of-the-art systems to solve hard natural language processing problems and other applications as well. I would not have been able to do this and be part of this AI trend without the help and support from many people, and I feel deeply grateful for them.

First and foremost, my greatest thanks go to my advisor Professor Zhi Wei. He is an extremely kind, caring, and supportive advisor that I could not have asked for more. He always has a very insightful, high-level view and understands the nature of problems when I was trapped in the mess of details. He also had the greatest patience for me when I needed more time, even for my delayed schedule or on those bad days. More importantly, he always encouraged me though I am not that confident about myself. I am forever grateful to him and glad I learned so much from him.

I would like to express my sincere gratitude to Professor James Geller, Wenge Guo, Senjuti Basu Roy, and James Calvin - for serving on my dissertation committee and a lot of support and guidance for my dissertation. Professor Geller is an extremely charming and enthusiastic person. I enjoyed and learned a lot from his class. He sets a high standard for students' work in every aspect and I appreciate his attitude for the long-term goods of students. Professor Guo, Basu Roy, and Calvin brought immense knowledge in different professional aspects. Their patience, inspiration, and knowledge helped me in all the periods of research and writing of this dissertation.

Besides, I would like to thank my peers Dr. Fei Tan and Dr. Tian Tian in our group for a lot of guidance and help through my Ph.D. studies. Both of them are role models for students in our group. They are both geeks and extremely enthusiastic and knowledgeable in their research fields. Fei has a high-level view and is also very detail-oriented. He has a very clear sense of how to define an impactful research project. Tian always has the magic to explain complicated technical terms clearly and map them to real domain-specific problems. In particular, they gave me a lot of support on my research even though their schedules were extremely tight considering how many things they had accomplished. I keep learning from them and always feel my passion getting ignited after having discussions with them (part of the honest reason is because of peer pressure). I enjoyed the teamwork with them, although I wish I could have done a better job.

I thank the whole deep learning journal club, especially Xiurui Hou, Kuang Du, Haoran Liu, and Cheng Zhong. They gave me a lot of help and support at various times. They are the bridge connecting me as the isolated and exhausted nursing mommy to the live and energetic research communities. I shared a lot of joyous moments with them, and those made my most vivid memories in my Ph.D. life.

During my PhD, I have received four years of teaching assistant support from the Computer Science department and one year of research assistant support from the New Jersey Innovation Institute. Their financial support was critical for me to continue my PhD program and I am very grateful for that. I also appreciate the timely support from faculty and staff in the Department of Computer Science when I had questions and challenges.

Special thanks go to my parents: Xichao Cheng and Yufeng Li, who worked very hard to give me the best education. My parents made me who I am today, and I will never know how to repay them. Like most Chinese parents, they are not good

at expressing their emotions but only wish the best for their kids. I know that they are at least a little proud of me for what I have been through so far.

Lastly, I would like to thank Lifeng for his love and support. For the past nine years, he is not only my classmate, my partner, my friend, but also the father of our lovely kids. Without him, I would not have come to study abroad. Without him, I would also not have made this journey at NJIT. I thank him for everything he has done for our family and me.

TABLE OF CONTENTS

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

Though the generic deep learning models perform well in canonical tasks, there is less evidence of them being utilized well in the improvement of domain-specific applications due to the unique challenges in the varied background and restricted resources. Nevertheless, the significant success of deep learning in generic tasks demonstrated it is capable of learning tremendous knowledge given large corpus, and it is promising to exploit that knowledge to solve other domain-specific problems. In this dissertation, we propose to use transfer learning to bridge the gap between generic deep learning models and domain-specific tasks.

For the first part, we consider the problem of project success prediction on crowdfunding platforms by introducing a multimodal solution. Despite the fact that the information in a project profile can be of different modalities such as text, images, and metadata, most existing prediction approaches leverage only the text dominated modality. Little study has been conducted to evaluate the effects of visual images on success prediction. One focus is to transfer the information learned from heterogeneous modality and to find a principle framework of combining different modalities. Moreover, meta information has been exploited in many existing approaches to improve prediction accuracy. However, such meta information is usually limited to the dynamics after projects are posted. Such a requirement of using after-posting information makes both project creators and platforms not able to predict the outcome in a timely manner. We aim to design and evaluate advanced neural network schemes that combine the transfer learning knowledge from different modalities to study the influence of sophisticated interactions among textual, visual, and metadata on project success prediction. Our approach requires only information collected from the pre-posting profile, which makes pre-posting prediction possible.

For the second part, we consider the problem of Named Entity Recognition (NER) in biomedical scientific literature, more specifically in genomic variants recognition. Most of the advanced generic NER algorithms can fail due to the out-of-vocabulary words in the biomedical literature and low-resource corpus. Efforts are needed to incorporate the domain-specific knowledge into those advanced generic NER algorithms. Our focus in this research is to transfer the domain knowledge learned from a large collection of scientific literature to facilitate the variants recognition tasks. On the other hand, the state-of-art approaches in most of the domain-specific applications are still heavily relying on feature engineering. The hand-crafted features only work well on specifically customized methods but cannot be generalized to other data, which most likely has out-of-scope rules, even sharing the same domain-specific background. We aim to investigate the end-to-end deep learning approaches to transfer the generic NER algorithms to genomic variation recognition.

For the third part, we target the text augmentation problem for general token-level natural language processing tasks by utilizing knowledge distillation and other techniques. Data augmentation has proved its effectiveness in promoting performance in computer vision and speech recognition applications. However, it is non-trivial to do text augmentation, and it is more sophisticated than image augmentation due to the fact that text augmentation is not an invariant transformation. One focus is on how to select the non-informative words for replacement adaptively. The other one is on how to train a model with a combination of clean data and noisy data consistently and confidently as well as lifting the performance. Text augmentation is an emerging field, and only a few approaches have been proposed recently. Most of them require massive external resources, which poses impassable obstacles for domains with restricted resources. We introduce, for the first time, a consistency-based unsupervised data augmentation method for the token-level task and a model to

automatically infer true labels for the new noisy unlabeled dataset. We also consider the low-resource domains, in which often large annotated sets are not available or external resources are restricted.

In the end, we conclude with a discussion of the role of transfer learning in those domain-specific tasks and some potential directions for future research towards a fully automated approach in other domain-specific applications.

CHAPTER 2

TRANSFER LEARNING IN MULTIMODAL DEEP LEARNING FOR SUCCESS PREDICTION ON CROWDFUNDING PLATFORM

2.1 Background

Crowdfunding platforms, like Kickstarter (kickstarter.com), IndieGoGo (indiegogo .com), and GoFundMe (gofundme.com), are emerging portals for designers, artists, startups, small businesses and entrepreneurs to raise funds for their projects through the internet. Such platforms provide opportunities for all the people who have creative ideas to pitch a campaign to gather capital and bring their ideas into reality. The fundraisers seek funding and will provide certain rewards depending on the amount of money provided by the backer, either in the form of future tangible products, experiences, services, or just having their names listed on a thank-you-board. Moreover, the audience demographics, aesthetic design, and terminologies are entirely different across varied fields from arts to sciences on crowdfunding platforms. For example, the owners of a local restaurant started a campaign to rebuild their business, which was devastated by Superstorm Sandy, and in return, the backers would have their names listed on the restaurant's website or a free dinner for the family as acknowledgement[1]; a teenager created a project to raise money for her college education and will give self-designed apparel back to donors[2]; a technician posted a profile explaining his innovative product, monetary goal, and timeline to deliver the goods while the backers could have the product after the project is completed[3].

Particularly, crowdfunding collects perks from individuals in the crowd rather than a large amount of funds from traditional fundraising professionals. Along with other financial alternatives such as microfinance and peer-to-peer lending,

[1]www.kickstarter.com/projects/573995669/rebuild-a-better-bait-and-tackle (accessed on Mar 31, 2020)
[2]www.kickstarter.com/projects/pythontear/pt-apparel (accessed on Mar 31, 2020)
[3]www.kickstarter.com/projects/jalousier/flipflic (accessed on Mar 31, 2020)

crowdfunding platforms have quickly risen to prominence due to their promising and attractive capital raising ability. Tens of thousands of innovative projects were fostered in the past few years. With the *JOBS ACT* and its subsection the *CROWDFUND ACT*[4] [8] signed into law by former President Obama, several new crowdfunding sites are expected to emerge very soon.

Project success prediction in crowdfunding is very challenging. There are recent efforts to elucidate contributing factors to make a successful project. The major existing studies consider only dynamically factors after projects are posted. Several factors relevant to success have been identified including the number of backers [34, 114], promotion on social media [34, 59], comments and replies [120], or funds pledged dynamic during a campaign [120]. All those post-launch factors show powerful predictive potential. Nevertheless, both project creators and platforms can predict the outcome only after the project has been posted for a certain period. It is desirable to predict project outcomes even before a project is posted. For platforms, they can reserve spaces for the projects more likely to succeed; for creators, they may revise their profiles proactively and save their time of going through predicted failure.

Moreover, most existing research focuses primarily on the text in profiles. Visual images as a critical modality in pre-launch profiles have not been studied yet, to the best of our knowledge. In most contexts, images are used to deliver ideas in a more effective way than text. For example, the product designers have to describe their concrete idea about the prototype by images, while artists need to demonstrate their artifacts by showing visible sketches. Some funding campaigns illustrate their blueprint merely by images instead of words[5]. With the huge collection of images available, it is appealing to ask: can we improve project success prediction accuracy by leveraging image information? There are two major challenges, however. First, as

[4]http://www.gpo.gov/fdsys/pkg/BILLS-112hr3606enr/pdf/BILLS-112hr3606enr.pdf (Accessed on Mar 31, 2020)
[5]www.kickstarter.com/projects/414768297/keepers-of-the-moonandsun-english-edition (Accessed on Mar 31, 2020)

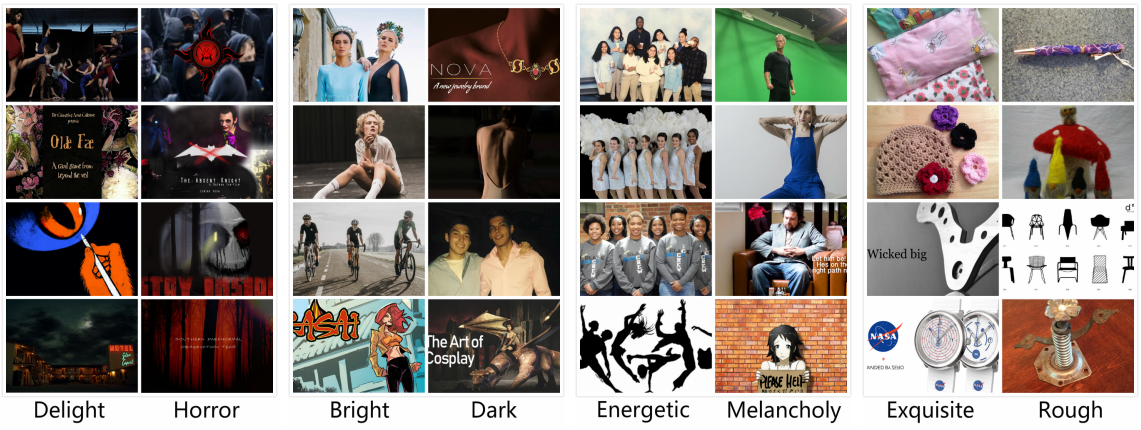

Figure 2.1 Examples of profile images on crowdfunding platform. For each case, images in the left column are from the successful projects, while the ones in the right column are from the failed projects. There is a clear difference in visual style between successful and failed projects.
Source: kickstarter.com

shown in Figure 2.1, image content understanding in the context of crowdfunding is far from trivial. Second, there may exist complex interactions among different modalities (image, text, animation, etc.) in a profile that works together to deliver success. The key solution for the first challenge is to transfer the knowledge from ImageNet, the canonical task in computer vision. Then, we also need to investigate and evaluate different multimodal representations and find a principled way to integrate the heterogeneous textual and visual information, as well as other modalities.

In this research, we introduced a Multimodal Deep Learning (MDL) model to predict the success of crowdfunding projects. To the best of our knowledge, our work was the first attempt so far to introduce the image factor to crowdfunding success prediction. The major contributions of this chapter can be summarized as follows.

- We designed the multimodal feature representation for the profile with textual, visual contents, and metadata. We investigated fusion schemes with different modalities and evaluated our multimodal architecture on the real crowdfunding dataset.
- We systematically investigated the contribution of images to project success. Our extensive experiments proved the effectiveness of images for promoting the outcome prediction from different aspects.

- Our approach requires only the information collected from the pre-launch profiles, which makes early prediction of project outcome possible. Such early predictions will benefit both platforms and creators.

The rest of this chapter is organized as follows. (1) We summarize previous studies related to crowdfunding, multi-sources applications in social media, and compare our work with the previous studies. (2) We highlight our proposed model for analyzing and predicting crowdfunding success, as well as demonstrate our feature representation for heterogeneous data sources. (3) We present our dataset, preprocessing steps, and experimental settings. (4) We report the results, examine to what extent the images could improve the performance of project success prediction, investigate the specific components from images, and discuss the implications of our findings. (5) We conclude with remarks and future directions of our research work.

2.2 Related Work

This work is connected to areas in project success prediction, and multimodal analysis.

2.2.1 Project Success Prediction

The booming trend of crowdfunding has drawn much attention from academia. Early research in crowdfunding has been performed by scholars in economics, management, and business, who primarily explored it from a financial perspective [1, 6, 71] or investigated its impacts on public sectors like education, business, and healthcare. Later researchers in fields of Computer Supported Cooperative Work (CSCW) and Human-Computer Interaction (HCI) studied the profile design and reward design of the campaign and tried to find why and how those crowdfunding sites are motivating extrinsic participants to post or fund projects [39, 49, 4, 69, 3, 91]. Most of the research in these fields is conducted from the perspective of project creators. Helpful supporting tools and strategies to reach a higher success rate are developed for both platforms and founders. Hui et al. [49] studied the work of creators on

their preparation, marketing, and follow through with projects. Similarly, Daoyuan [25] discovered that the creator's previous reputation and reciprocal history could positively contribute to the success of projects on Kickstarter. Gerber et al. [39] revealed that although the ideas on the crowdfunding platforms span across fields and vary in scope the anticipated motivators are connected by the commitment to an idea and a community with similar interests.

More recently, the project success prediction of crowdfunding projects has become a hot topic in NLP fields. The conventional approach was to build a machine learning classifier like SVM based on meta features from the campaign profile. Greenberg et al. [40] showed an improvement by utilizing various decision tree algorithms and SVM trained with features such as whether the video was present, the sentence count in the profile, project goals, project duration, and other possible additional factors like creators' demographic attributes. Some advanced approaches utilize textual descriptions while certain models additionally exploit dynamic information by monitoring social media or crowdfunding campaigns. Mitra and Gilbert [69] analyzed the linguistic features with 59 other common features to predict project success. Yuan et al. [112] proposed a text analytic topic framework to predict the fundraising success by extracting latent semantics from the text description with a combination of common numerical features. Etter et al. [34] and Zhao et al. [120] studied dynamic time-series factors by tracking social media and monitoring the dynamic features like backers and money pledged status during the campaign. Zheng et al. [121] found the degree of the creator's social network is positively associated with project success since creators can broadcast their crowdfunding projects to a broader audience through their social network. Li et al. [59] formulated the success prediction problem from the aspect of censored regression and achieved better performance utilizing temporal features from pledging and social media dynamics.

From the prior descriptions, we can observe that most of the earlier works focused on textual profile and post-launch information, which inhibits both project creators and platforms from being able to predict the outcome early. Therefore, to make pre-posting prediction possible, our approach focused on the joint analysis of the textual and visual information collected from the pre-launch stage, which has not been fully explored yet in previous studies. The dynamic conditions during the funding process, like changes of the money pledged, creators' social media accounts, the number of comments and backers, are beyond the scope in our research. Daoyuan [25] also discovered that the category could moderate the effect of the content related factors, like narrative richness. Furthermore, none of them explored the role of images in profiles. Introduction of the visual modality could leverage textual data and their associated meta information to identify extra engaging underlying factors.

2.2.2 Multimodal Analysis

The multimodal approaches using joint text and image analysis have been explored in social media for quite a while, e.g., multimodal semantic analysis [82], multimedia market evolution monitoring [114], and multimodal news analysis [80]. To encode visual information, most of the earlier approaches relied on hand-crafted features combined with methods to aggregate manually engineered descriptors before the rise of CNNs. The Bag-of-Visual-Words (BoVW) [21] model was the common choice of image feature representation. It would collect codewords from feature descriptors like Scale-Invariant Feature Transform (SIFT) [62], and then learn the codebooks from unsupervised learning. The feature descriptors for BoVW usually are extracted from local extrema, like edges and corners, inside a small sub-region. It ignores the global semantic relations and can work effectively only if the provided feature descriptors are well-matched. It is more suitable for tasks of classification or identification on small data sets. However, it is hard to obtain the optimal size and computationally

infeasible on large datasets [17]. As the generation of large codebooks requires enormous computational resources, it fails on large datasets with more than millions of descriptors as our case, and its performance is variable and less satisfactory due to the complexity and diversity of image patterns [36]. In more recent works, a hybrid architecture was introduced to leverage the best of the BoVW and deep neural networks. One approach was to combine the pre-trained deep features with BoVW [114]. In contrast, some other approaches projected the hand-crafted feature descriptors to lower dimensionality and feed them to neural networks [77].

A CNN pre-trained with a large database, like ImageNet, can be used as off-the-shelf feature extractor. The transfer feature learning from existing deep convolutional neural networks showed promising results in varied research projects [87, 51, 63, 99]. Nevertheless, this approach has not been proved to be an effective method in crowdfunding project success prediction. Moreover, the adoption of the existing approach was hindered due to the differences of scale of amount of images per profile and much diverse visual context, which needs to be carefully addressed in this problem.

2.3 Problem Formulation

As a crowdfunding project example illustrated in Figure 2.2, the typical structure of the campaign page includes a goal, a project description (red box at the bottom) embedded with tens of figures (red box in the middle), perk structure (right column of the webpage), links to the creator's social media platforms, and some metadata (red box on the top) like category, location.

Let $\mathcal{D}$ represent the crowdfunding dataset with N profiles. The definition of success is that the creators can reach their initial goal by the end of the limited campaign period. Then $y_k \in \{\pm 1\}$ specifies the ground-truth outcome whether project $k \in [1, N]$ is successful or failed. Our goal is to learn a multimodal feature

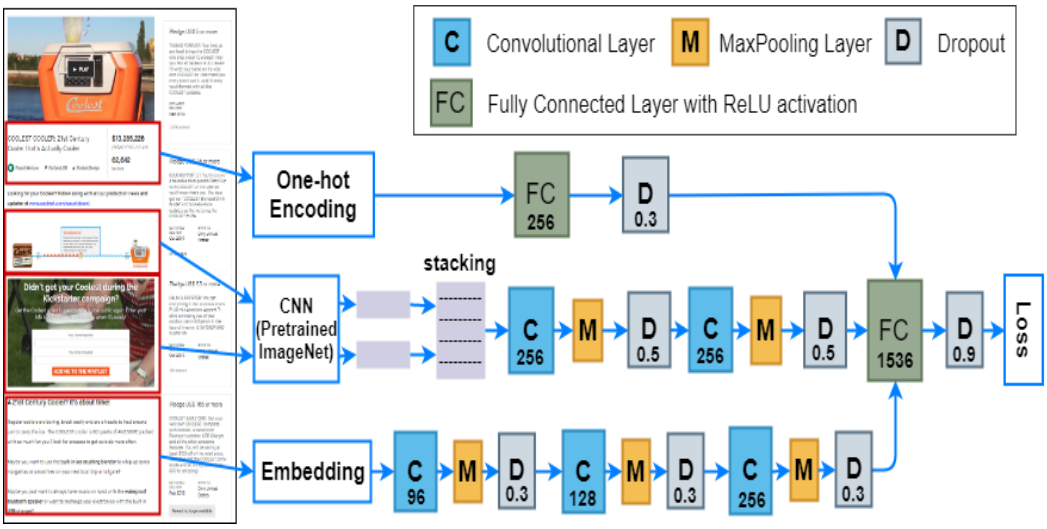

Figure 2.2 Framework Overview of Our Multimodal Deep Learning Framework. The input is from pre-posting profile features. There are three pipelines in our framework: (1) top branch for meta modality; (2) middle branch for visual modality; (3) bottom branch for textual modality.

map $F(X)$ for given $\mathcal{D}$ to predict the success outcome $\mathbf{Y}$. The feature mapping is defined as:

$$F(X) = f(W_F \cdot (\phi(X_T) \oplus \phi(X_I) \oplus \phi(X_M)) + b_F) \tag{2.1}$$

where the symbol $\oplus$ means concatenation, $f(\cdot)$ is a non-linear activation function such as rectified linear unit (ReLu), and W_F, b_F are weight and bias, respectively. In this equation, $\phi(\cdot)$ can be considered as a feature mapping of modalities for text X_T, images X_I, and metadata X_M.

In our research, the training objective function is based on cross entropy:

$$\mathcal{L} = -\sum_{k=1}^{N} [\delta(y_k = 1) \log(p_k) + \delta(y_k = -1) \log(1 - p_k)] \tag{2.2}$$

In the above, $\delta(\cdot)$ is the indicator function, and $p_k \in [0, 1]$ is the estimated probability for the class with label $y_k = 1$. And our implementation of loss layer combines the sigmoid operation for computing p_k given $F(X)$.

2.4 Joint Fusion of Heterogeneous Features

Figure 2.2 illustrates how our system computes multimodal representations. As shown in Figure 2.2, our MDL model has three branches: (1) bottom branch for encoding textual input $\mathcal{T}$; (2) middle branch for encoding visual content of images $\mathcal{I}$; (3) top branch for encoding meta information $\mathcal{M}$. Each branch is composed of either a CNN subnet or fully connected hidden layers. In general, each branch can have a different number of layers, and the inputs for the three branches could be produced by their own upstream networks, such as word embedding or pre-trained ImageNet. At the end of each branch, the feature maps from three streams are concatenated into one feature map.

Textual Feature. We applied two popular feature representations for the textual input: Bag of Words (BoW) [10] with Term Frequency-Inverse Document Frequency method (TF-IDF) [84], and word embedding. In BoW, the text in a given profile is encoded as a histogram of weights for the words appearing in the $\mathcal{T}$, and the weights computed with the TF-IDF weighting scheme. Despite the fact that the generated feature vector ignores the order and semantics of the word, the BoW model shows its power in many of varied NLP applications. Moreover, its sparsity and high dimensionality would lead to computational issues in some applications. In contrast with the sparse BoW representation, a word embedding encodes text as the continuous distributed representation of a short fixed-size vector. It is semantically compatible with representing both words and their related context. The embedding model we used is GloVe [75] with 300-dimensional vectors.

Visual Feature. The pre-trained ImageNet is used to extract the feature map for each individual image. Specifically, we use a pre-trained 16-layer VGG model [89] and take its output from the fully-connected layer (fc6). For any project k, $I_k \subseteq \mathcal{I}$ is used as its input of images, and additionally $\ell_\imath \in I_k$ in which $\imath$ is the index of images in

profile k which may contain multiple images with varied size n. Given an image ℓ_i, it is rescaled to 224×224 and represented by a 4096-dimensional vector extracted from the VGG16 model. Image feature maps for any $\ell_i, i \in [1, n]$ in I_k are used as visual input for profile k. Then we applied two popular approaches to generate the visual representations for profiles: BoVW and CNNs. The pre-trained deep features are used as descriptors and clustered using mini-batch k-means model to generate the codebook for BoVW. Each profile can then be represented by a bag of visual words. For CNNs, those pre-trained deep features are aggregated in different ways, like averaging, flattening, and stacking. Kornblith et al. [55] suggested ResNets the best feature extractor for transfer learning tasks. However, we observed better performance using the VGG16 model. Considering our feature vectors are generated from a large group of sparse images, and the styles of images on crowdfunding platforms differ from ImageNe as shown in Figure 2.1, this hypothesis may justify the observation in our experiments to some extent.

Meta Feature. Metadata in our experiment is composed of campaign category and funding goal extracted from the profile. Funding category was converted by the one-hot encoder directly, while the funding goal is converted with the binning transformation[6]. We grouped numerical funding goal values into a set of customized discrete ranges and then assigned each numerical value to a range bin. Specifically, we summarized the data and used (1) fewer bins to encode numeric values falling inside the lower and higher quantile; (2) more bins for other values belonging to the in-between quantile. Then, each numeric value of the funding goal would be assigned to a binary vector. More specifically, we set three bins for the lower 0 - 0.3 quantile, four bins for the upper 0.9 - 1 quantile, and 50 bins for the values in between.

[6]https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/group-data-into-bins (Accessed on Mar 31, 2020)

2.5 Experiments

2.5.1 Data Set

Data Collection. The evaluation of our MDL framework was done on a dataset scraped from Kickstarter. Kickstarter has become the largest crowdfunding platform in the U.S. since it was launched in 2009. The campaign profiles may be modified during the funding stage and thus not exactly the same as the pre-launch profiles. Yet, it is likely to serve as a broadly useful model for examining crowdfunding efforts. We crawled the data using the seed from webrobots[7]. We ran the scraping script for two weeks to collect the data. For each campaign webpage, we extracted the textual profile, images, category, funding goal, funding start date, and end date. The textual profile includes the title, summary, funding description, and risk/challenges. Visual contents include all the jpg format images but not the png or gif format. Furthermore, the campaigns before 2015 were discarded.

Preprocessing. Before feature extraction, we carried out the following data preprocessing procedures:

1. *Project status.* The status of the funding campaign on Kickstarter includes successful, failed, canceled, suspended, removed, live, and others. We only kept projects with the status of successful or failed.
2. *Textual profile.* We removed the stop words, punctuation and digits from the text, and further removed the projects with text length less than 12 words.
3. *Visual profile.* We removed small images whose row dimension size is less than 200. We observed the small images function as banners that are the aesthetic shape or section headers. They might increase the aesthetic value of the profile but they cause excessive extra computational costs.
4. *Category.* We also removed the projects in skewed categories. That means if the projects listed in a given category less than 1/10 of the largest category in our data, we removed those categories and projects.

[7]https://webrobots.io/kickstarter-datasets/ (Accessed on Mar 31, 2020)

Data Splitting. To train our model to select the best parameters and evaluate the performance, we split the dataset into three parts, as shown in Table 2.1, which describes the statistics of our dataset grouped by campaign year and funding outcome. The reported overall success rate of Kickstarter was 43% in 2011[8], and increased to 75% in 2017[9]. To evaluate the predictive potential on a temporally changing distribution, we used campaigns launched in 2015 and 2016 as the training set, campaigns in 2017 as the validation set, and campaigns in 2018 as the testing set.

Table 2.1 Statistics of Our Dataset.

Data Set	Year	*No. of Samples*			*No. of*
		Success	*Fail*	*Total*	*Images*
Training	2015/16	8705	9806	18511	135404
Validation	2017	4655	2687	7342	65907
Testing	2018	5429	2830	8259	83102

To evaluate the performance, the data is split by year.

2.5.2 Experimental Settings

Evaluation Metrics. We evaluated the prediction performance of all methods in terms of Recall, Precision, F1-Score, and AUC@ROC (AUC), respectively.

Evaluation Algorithms. Two important goals of our work were to introduce image as an additional modality and to limit the analysis to pre-posting profiles. Neither perspective has been addressed in previous work. Thus, the focus of our evaluation lies in the investigation of different modalities and feature representations for project success prediction without any post-posting information. We compared

[8]www.kickstarter.com/blog/happy-birthday-kickstarter (Accessed on Mar 31, 2020)
[9]www.kickstarter.com/blog/happy-8th-birthday-kickstarter (Accessed on Mar 31, 2020)

Table 2.2 Statistics about Number of Words and Images in Profiles.

Level	*Quantile*	*No. of Words*	*No. of Images*
1	- 0.25	150	1
2	- 0.5	260	4
3	- 0.75	450	10
4	- 1 (0.9)	4572 (725)	113 (18)

We used 0.9 quantile as a cut-off for the feature dimension in the MDL model, so the values inside the parentheses are the actual values for text length and image numbers in our experiments.

our framework with the linear SVM in all investigated cases. Our research explored the following questions and corresponding methods:

1. *Which performance levels can be achieved by textual profile information only?* We studied classifications based on textual information only: **SVM-BoW** and **MDL-Text**, and evaluated the boundaries of textuality only. The best text-based approach serves as the baseline.

2. *Which performance level is achievable by visual information?* We investigated the suitability of the visual modality only and evaluated different aggregation strategies for **SVM-BoVW** and **MDL-IMG**. The best image-based approaches are reported.

3. *Which multimodal representation performs best? Does the multimodal combination of different feature representations facilitate classification?* We compared the methods with all modalities with Text, Image, and Metadata (TIM), and reported the results of **SVM-TIM** and **MDL-TIM**. Moreover, further studies on a varied combination of different modalities (**MDL-Text/IMG, MDL-Text/Meta, MDL-IMG/Meta**) were conducted to learn the most contributing element. In addition, the parameters were tuned for each model, like the number of CNN layers, the value of kernel size for CNN layer, the pooling size for max-pooling layers, the number of fully connected layers and the value of neuron units, and the dropout rate for each dropout layer.

4. *How sensitive is the performance to the modalities?* More specifically, is the performance consistent on different project profiles with varied textual lengths and image numbers? We performed an ablation analysis to different granularity to test this.

Hyper-Parameter Tuning. For each case evaluated, we varied the most influential parameters to train the models. The models are tuned based on the validation, and the optimal parameters are reported accordingly based on the F1 measure. Our MDL model is implemented in Python using Keras with TensorFlow backend. We used the RMSprop [97] optimizer and the learning rate is set to 1e-5 for 100 epochs. We set the batch size to 128 campaign projects and employed early stopping with 20 epochs and dropout [92] to prevent overfitting. To reduce the computational cost of training CNN, we truncated the length of text description and the number of images in the profiles. The cut-off value we used is 0.9 quantile for both. Specifically, it is 725 for words in the text and is 18 for images as shown in Table 2.2. Nevertheless, we are still dealing with a large set of images for each profile.

2.6 Results and Discussion

Baseline. As shown in Table 2.3, compared with MDL-Text, SVM-BoW achieved decent performance in terms of all metrics, especially precision and recall. Thus, SVM with textual modality only is the baseline.

Table 2.3 Results of Single Modality on Text Only

Methods	*Recall*	*Precision*	*F1 Score*	*AUC*
SVM-BoW	**0.7356**	**0.7424**	**0.7387**	**0.7356**
MDL-Text	0.6831	0.7153	0.6920	0.7788

Visual Modality. We trained a linear SVM classifier with the BoVW feature obtained by mini-batch k-means with cluster sizes varied from 30 to 300. With respect to the visual representation for MDL-IMG, we tried different approaches to aggregate the feature map $\ell_i, i \in [1, n]$ where $n = 18$ for a given I_k, e.g., average pooling, flattening, and stacking. The stacking approach performed best on the MDL-IMG model. As we can see from Table 2.4, the MDL-IMG model yields better visual representations than BoVW. This demonstrated the deficiency of BoVW while dealing with large datasets due to the complexity and diversity of image patterns and confirmed the superiority of state-of-the-art feature transfer learning from computer vision. However, its performance is worse than MDL-Text.

Table 2.4 Results of Single Modality on Images Only

Methods	*Recall*	*Precision*	*F1 Score*	*AUC*
SVM-BoVW	0.6570	0.6623	0.6592	0.6569
MDL-IMG	**0.6738**	**0.6809**	**0.6768**	**0.7340**

Multimodal Combination. We utilized all collected textual, visual, and meta information to evaluate their effects on performance and investigated different feature fusing approaches. The fusion techniques could be complicated as [95], and we found the FC layer achieved the most effective results in our experiments. As demonstrated in Table 2.5, both SVM-TIM, and our MDL-TIM outperformed baseline method SVM-Text, which confirms the motivation outlined in the introduction: images as an essential component in profile could help to improve the predictive performance. Meanwhile, our proposed model MDL outperforms SVM, which demonstrated the superiority of deep transfer learning in computer vision over the BoVW model.

Table 2.5 Comparison with All Modalities

Methods	*Recall*	*Precision*	*F1 Score*	*AUC*
SVM-TIM	0.7411	**0.7595**	0.7483	0.7411
MDL-TIM	**0.7505**	0.7568	**0.7534**	**0.8326**

Ablation Analysis. Additionally, we investigated the most contributing modality by removing one factor at a time. The performances of different modality combinations (Text/Image, Text/Meta, Image/Meta) is reported in Table 2.6. As we can see that IMG/Meta performs the worst after removing text from the model, thus text still carries more predictive messages in general. Meanwhile, we observed that metadata could introduce more distinctive signals. It becomes clear after checking the images on Kickstarter because the patterns in different categories vary from each other, e.g., styles of product design do differ tremendously in all respects.

Table 2.6 Ablation Analysis of MDL

Combinations	*Recall*	*Precision*	*F1 Score*	*AUC*
Text/IMG	0.7335	0.7241	0.7278	0.7995
IMG/Meta	0.7108	0.7191	0.7143	0.7807
Text/Meta	0.7385	0.7320	0.7348	0.8162

Table 2.7 Correlation between accuracy and images/text levels

Correlation	*Text+*	*Image+*	*All*
Text Level	0.618	—	—
Image Level	—	0.776	0.570

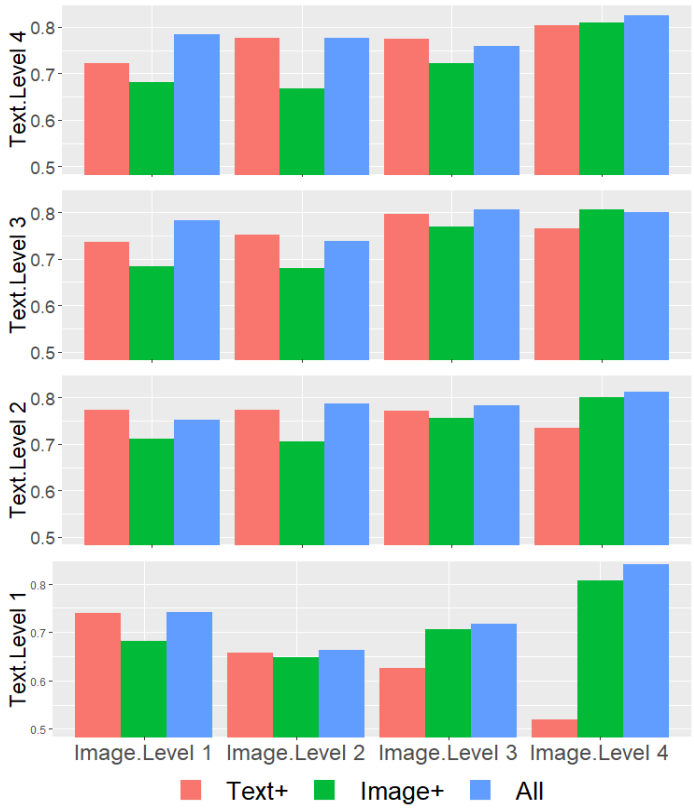

Figure 2.3 Investigation on the ablation of modalities. Results are grouped by the length of text profile and numbers of images. Accuracy is reported here by split Levels of Word and Image Number in Profile for models MDL-Text/Meta (Text+), MDL-Image/Meta (Image+) and MDL-Text/Image/Meta (All). The profiles are split into groups by definition in Table 2.2. The higher the level is, the more words or images are in the profiles.

Sensitivity analysis of the modalities. Furthermore, we investigated the effects of textual length and image numbers to test the sensitivity of performance on different modalities. Firstly, we split profiles into four levels as statistics shown in Table 2.2. Then we analyzed the accuracy of models MDL-Text/Meta (Text+), MDL-Image/Meta (Image+), and MDL-TIM (All) from Table 2.5 and reported the fine-sorted results in Fig. 2.3. In general, MDL-TIM outperforms significantly in most cases. Moreover, we surprisingly found that the MDL-Image/Meta outperforms MDL-Text/Meta in the last column (image-level four across all text levels). This indicates better prediction can be achieved if given more images than text. Interestingly, we

found that better success prediction will not necessarily be achieved if given more text by checking the trend from bottom row (text level one across all image levels) to the top row (text level four across all image levels). The correlation between accuracy and text/image level reported in Table 2.7 also supported our observation. The entries with p.value > 0.05 are removed from the table. Giving this important revelation, it is suggested that the performance of success prediction will be enhanced by integrating images despite limited text description. Yet to have the best outcome, all modalities should be considered. To this end, we demonstrated the prevailing role of images in crowdfunding success prediction.

2.7 Conclusions

In this chapter, we utilized the missing visual modality in previous approaches and developed a multimodal deep learning model for the project success prediction problem. The visual feature representation is built upon transfer learning representations from ImageNet. The extensive experiments were carried on a real-world data set collected from Kickstarter. The empirical studies illustrated that visual images are as important as text and superior performance could be achieved by incorporating them. The corresponding results showed that our model can deliver the best performance over alternative methods. The ablation analysis of the modalities also provided useful insights for project creators.

CHAPTER 3

DEEPVAR: AN END-TO-END DEEP LEARNING APPROACH FOR VARIANTS IDENTIFICATION IN BIOMEDICAL LITERATURE

3.1 Background

Due to the sheer volume of new biomedical literature in the last decade, automated information extraction tools are critical for researchers to access the explosive amount of published research. One of the most distinguishing features of scientific biomedical literature is that it contains a large number of terms and entities, in which some are explained in public electronic databases if researchers manually built professional knowledge for them. In the biomedical context, entities would typically be short phrases as the representations of a specific object, e.g., names of genes or proteins, genetic variants, gene products, genetic diseases, drugs, etc. It is impossible to access all that information among up-to-date published research manually. Information extraction is a critical factor for efficiently accessing and integrating such knowledge. The ultimate goal of information extraction is to extract knowledge automatically. Still, usually, the first task is to identify name entities from text, more formally as Named Entity Recognition (NER) task.

To identify named entities present in the text, statistical approaches, such as Maximum Entropy (ME) [9] and Conditional Random Fields (CRFs) [56], are used in most of the previous works by either learning patterns associated with a particular type of entities or hand-built rules. The performance of such algorithms heavily depends on the design of hand-crafted features, and the number of features could be so large that the models are prone to overfit on training corpus and fail for practical use.

Recently, the Deep Neural Network (DNN) models have increasingly been used in generic Natural Language Processing (NLP) areas and achieved significant success,

pushing most of the benchmarks to a new level of state-of-the-art. More importantly, those models minimized the feature engineering efforts by learning the hidden patterns from the large volume of labeled samples. However, due to the high cost of expert curation, the size of curated training data is often restricted in biomedical domains. As shown in Table 3.1, the benchmark data set of variants is much smaller than other works. Moreover, we can also observe that the variant entity names contain more diverse orthographic and morphological alterations. The heavier linguistic heterogeneity exacerbates the challenge of this problem.

Our goal in this research is to develop an end-to-end DNN NER model that automatically identifies variants in biomedical literature and classifies them into one of a set of predefined types. Despite many attempts on other biomedical benchmarks in the past, it is the first attempt to use a deep learning approach for the genomic variants recognition, and it remains a challenging task. The main challenges are:

- To minimize feature engineering effort, automatically generalizing hidden diverse linguistic patterns is harder from limited training resources.
- To differentiate the ambiguous entities or synonyms, learning some effective feature representation is harder with shallow networks from limited trainable resources.
- To limit the false positive errors, both the entity identification and the entity boundaries need to be accurately inferred since it is critical for downstream applications such as relation extraction.

In this research, we took full advantage of the generic state-of-the-art deep learning algorithms and introduced our Deep Variant (DeepVar) Named Entities Recognition model. We tried to find a principle way to transfer domain knowledge in the biomedical literature and built an end-to-end DeepVar model. Our results showed that our DeepVar could achieve better performance than the state-of-the-art algorithms

Table 3.1 Comparison of Other Data Sets with Ours to Demonstrate The Extreme Low-resource Situation in Our Research

Data Set	*Size*	Entity types and counts	*Named Entity Example*
BC2GM	20,000 sentences	Gene/Protein (24,583)	S-100; Cdc42; RecA; ROCK-I
BC4CHEMD	47,402 sentences	Chemical (84,310)	(25)MgPMC16; SAHA
BC5CDR	30,677 sentences	Chemical(15,935)	cyclosporin A; L-dopa
		Disease(12,852)	cardiovascular arrhythmias; swelling
JNLPBA	13,484 sentences	Cell Line(4,330)	Jurkat T-cells; Hsp60-specific T cells
		DNA (10,589)	cytokine gene; human interleukin-2 gene;
		Gene/Protein (20,448)	NF-kappaB site; Hsp60; retinoic acid receptors
		Cell Type (8,649)	16HBE human bronchial epithelial cells
		RNA(1,069)	GR mRNA; glucocorticoid receptor mRNA
NCBI-Disease	8,336 sentences	Disease(6,881)	MCF-7 tumours; breast and ovarian cancer
tmVar - Ours	4,783 sentences	Protein Mutation (653)	p.Pro246HisfsX13; S276T; Arg987Ter
		DNA Mutation (751)	c.399_402del AGAG; Ex2+860G>C; -866 promoter(G/A);
		SNP (136)	rs2234671; rs1639679

without any feature engineering. Meanwhile, our results from extensive empirical studies may shed light on other low-resource applications.

3.2 Related Work

Statistical machine learning systems have proven their success for NER in earlier works. However, almost all these approaches relied on feature engineering to some extent. They learned patterns associated with individual entity classes by many hand-crafted features such as internal linguistic features or external knowledge. In Biomedical Named Entity Recognition (BioNER), which extracts important entities such as genes and proteins, various similar machine learning-based approaches have also been applied and achieved good performance. The widely used hand-crafted features include different types of linguistic features such as syntactic and semantic information of words, as well as domain-specific features from biomedical terminologies such as BioTesaurus [60] and Toxicogenomics Database [26].

With respect to the genomic variation recognition, all the previous work including MutationFinder [12], EBNF [58], OpenMutationMiner [73], tmVar [103], SETH [96], and NALA [13] employed dozens of regular expressions to build orthographic and morphological features, like word shape, prefixes, and suffixes, for their variants entities identification systems. Since the regular expressions used for generating customized hand-crafted features are fixed and can only describe limited patterns for variants, all the previous work mainly focused on techniques improving the regular expression to capture more patterns [103, 96]. Nevertheless, they still need to add a few post-processing steps to achieve better results [103, 13].

More recently, due to the development of deep learning techniques, it has become a fashion in NER applications to minimize the efforts in feature engineering and build an end-to-end system. The first attempt to use deep learning in the NER task was the SENNA system [20], which still utilized lots of hand-crafted features. Then,

varied works were applied at different levels to abolish the hand-crafted features. The current state-of-the-art approaches now regulate both the word level and character level representations intertwined by both Bidirectional Long Short-Term Memory (LSTM) Neural Network [47] and Convolutional Neural Network (CNN) [113]. Some works focused on building the shallow word-level representations with character-based features through CNN [20, 117, 52, 94], or bidirectional LSTM [48, 64, 94]. The majority of work combined both word-level and character-level features to achieve the best performance. Nevertheless, some still applied slight pre-processing steps to normalize digit characters [16, 57, 94], while some works employed marginal hand-crafted features [20, 48, 16, 94]. For the first time in NER literature, [57, 64] used the end-to-end structure without any hand-crafted features.

Various work has been done on varied BioNER domains to improve the effectiveness of the aforementioned models. Habibi et al. [43] investigated the effectiveness of the approach proposed in [57] for chemicals, diseases, cell lines, species, and genes name recognition, while Dernoncourt et al. [27] verified the same approach on patient notes. Yoon et al. [111] investigated the approaches of [64] on chemicals and disease entities. Wang et al. [102] utilized the multitask architecture similar to [61] and verified on chemicals, cell lines, disease, genes, and other name recognition. Xu et al. [107] proposed a modified framework based on [57] by adding extra sentence level representation as global attention information and verified on clinical NER task. Nevertheless, even some recent state-of-the-art BioNER works still need to elaborate marginal external information, the task of variant identification remains open in literature, and to build an end-to-end approach can be challenging.

Meanwhile, it is worth noting that most of those studies employed the word-level distributional representations, well-known as word embedding. Word embedding is proposed to polish up the deficiency of Bag-Of-Words (BOW) model and becomes a key component for word features in the NLP application. The most famous

model is word2vec [68]. However, embeddings from generic corpora fail to capture specific meanings within the biomedical domain. One of the common challenges is the Out-Of-Vocabulary (OOV) words, which can be rare terms like mutants or unseen forms of known words like disease names. Those entities are not typo and have high occurrence but cannot be found in the canonical pre-trained embeddings. Learning high-quality representations for biomedical literature can be challenging. Xu et al. [108] employed {left, right, and max}_surrounding_embedding from surrounding words as hand-crafted embedding features and achieved better accuracy in clinical NER task. [65] used some character-based 1-in-m encoding schemes to solve rare words or typos. Recently, the word representations pretrained on a large collection of domain-specific texts (PubMed, PMC, etc.) have been proved superior than generic word embeddings [43, 70]. It is promising in low-resources variant identification tasks, yet the evidence is still missing.

3.3 Deep Variants Identification Model

In this section, we presented our Deep Variants (DeepVar) NER model for identifying variants in a low-resource data set. We focused on neural sequence representation learning to capture contextual information and hidden linguistic patterns without hand-crafted features or regular expressions. The architecture is shown in Figure 3.1. The sentence "We identified T10191C(P.S45P) in ND3." used in the figure is for illustration purposes. As illustrated in Figure 3.1, our DeepVar model contains three parts:

Input Embeddings Each word in the sentence has two types of input: word-level (words in red color) and character-level (characters in green color). For character-level input, we applied one-hot encoding (green circle on top) as character embedding; with respect to word-level input, we used the word embedding (red circle icon; 3.3.2). It is noted that the word embeddings are pre-trained on a separate large collection of the

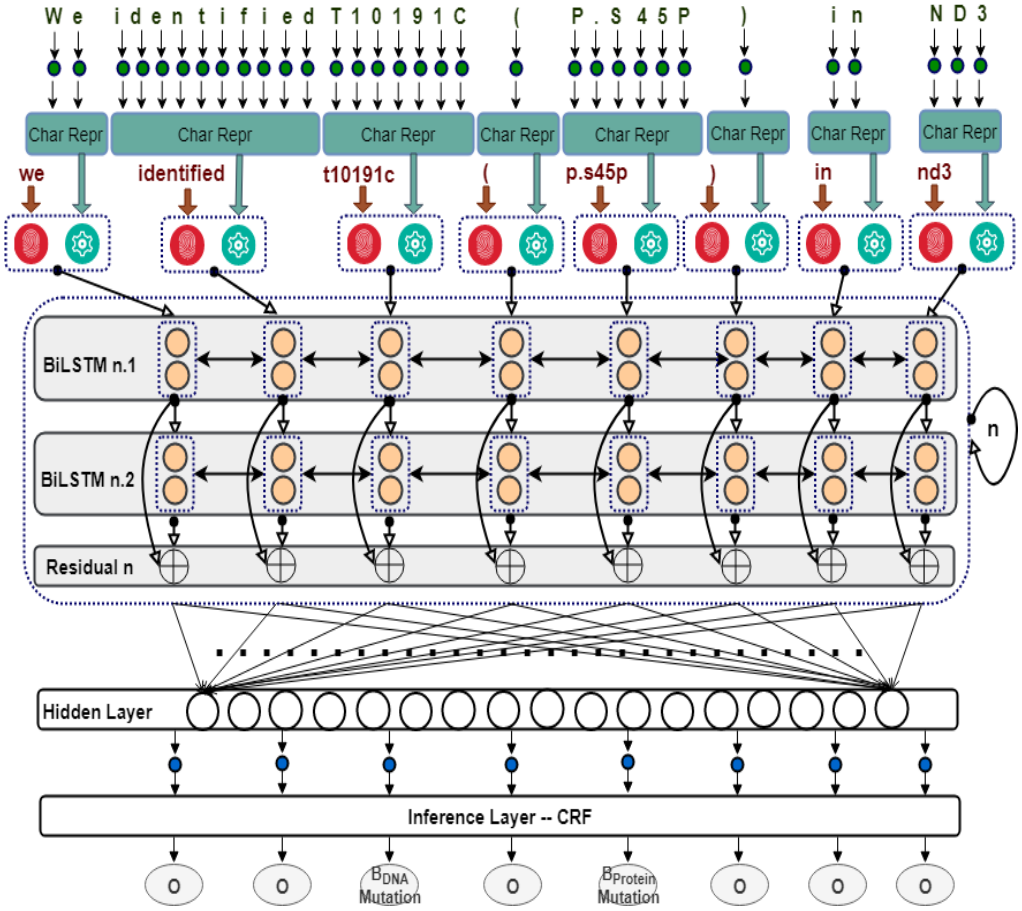

Figure 3.1 The Architecture of proposed DeepVar Model. The small green circle, green rectangle, and large green circle icon represent character embedding, character sequence representation learning module, and character sequence representation respectively; the red circle icon represents word embedding; the gray boxes including two BiLSTM layers and Residual represent the unit element of word sequence representation learning module which may have n unit; the small blue circle represents the hidden stats of word sequence representations from the hidden layer.

biomedical corpus, while character embeddings are built from our variant BioNER task.

Feature Representation Learning The character representation (green circle icon) is learned from modules with LSTM or CNN ("Char Repr"; Section 3.3.1). Then it would be concatenated to word embeddings as the input of word sequence representation learning module (gray boxes in the middle; Section 3.3.3). This module contains the stacked BiLSTM networks with integrated residual layer, and it is designed to capture long-term information and effective contextual representations.

Inference Module Finally, the final word feature representation for each word will be the hidden states from the hidden layer (blue circle). The CRFs inference layer will take it and assign labels to each word (Section 3.3.4).

3.3.1 Character Embedding and Feature Representation

Character information has been proven to be critical for entity identification tasks [16, 57, 64]. First of all, character embedding could handle the OOV words to some extent since it could enclose the morphological similarities to some established words. Moreover, it also could be able to insert the orthographic and linguistic patterns for variants such as prefix, suffix, and punctuation. For example, mutation names often contain alphabets, digits, hyphens, and other characters like "HIV-1", "IL2", "rs2297882", and "C>T". It is crucial to learn all those hidden morphological and orthographic patterns automatically for inference.

We represented the character embedding through a lookup table. The lookup table we used contains 70 characters, including 26 English letters, ten digits, 33 other characters, and one placeholder for the unknown character. The full alphabet is shown in Table 3.2. More specifically, we employed 1-of-m encoding in which each character is encoded as an m sized vector where all values are zero except the entry

Table 3.2 Look-up Table for Character Embedding

letters	abcdefghijklmnopqrstuvwxyz
digits	0123456789
others	,;.!?:‘’“”/|_@#$%ˆ&*˜+−=<>()[]{} UNKNOWN

corresponding to the found character with a value of one. Based on the lookup table, $m = 70$ is used. Subsequently, each instance is then represented by a sequence of $m = 70$ sized vectors with character sequence length l, where l is a hyperparameter.

Then LSTM and CNN are used to capture the hidden morphological and orthographic patterns and learn character-level representations:

Character CNN Chiu and Nichols [16], Ma and Hovy [64] have investigated the effectiveness of using the CNN structure to encode character sequences. In our research, we employed the same architecture as in [64]. More specifically, one CNN layer was used following with max-pooling to capture character-level representations.

Character BiLSTM In study from Lample et al. [57], the BiLSTM is utilized to model the global character sequence information. The final states from the *left-to-right* forward LSTM and *right-to-left* backward LSTM are concatenated as character sequence representations.

3.3.2 Word Embedding

Word embedding represents a word as a continuous dense vector with a low dimension at the lexical level. In addition to character representations, word embeddings are still crucial to represent semantic information. Habibi et al. [43], Mohan et al. [70] demonstrated that the word representations pretrained on a large collection

of biomedical literature (PubMed, PMC, etc.) could outperform the generic word embeddings. First of all, it can learn the good representations for biomedical named entities (e.g., "IL2" for "Interleukin 2"), which are not in the canonical newswire corpus. Moreover, it should be capable of capturing the syntactic and semantic information for the ambiguous entities (e.g., "TNF alpha" can refer to a protein or DNA) or different words referring to the same mutation (e.g., evolving of time or simply the preference of authors).

Recently, the contextual embeddings such as ELMO [78], Flair [2], and BERT [28] are proposed and achieved the state-of-the-art performance on all the generic NLP applications. The domain-specific embeddings, which are trained on the large biomedical corpus, well known as BioEmbedding, are simultaneously made available to the public. Various works [50, 74] showed that the BioEmbedding outperforms vanilla embedding on BioNER tasks. In our experiments, we also investigated different pre-trained BioEmbeddings.

3.3.3 Word Representation Learning

Although CNN could be utilized to model word-level representations [20, 94], we employed the BiLSTM in our research as it is more widely used [57, 64, 16, 48, 61] and more powerful to capture the contextual distributional sensitivity. A BiLSTM includes forward LSTM and backward LSTM. The forward LSTM captures the contextual information from left to right, while the backward LSTM extracts information in a reversed direction. The hidden states of the forward and backward LSTM are concatenated for each word and are given to the next layer.

Basically, the input to an LSTM network is a sequence of vectors $X = \{x_1, x_2, \ldots, x_T\}$, where x_t is a representation of a word in the input sentence x at a certain layer of the network. The output is a sequence of vectors $H = \{h_1, h_2, \ldots, h_T\}$, where h_t is a hidden state vector storing all the useful information at time t. At step

t of the recurrent calculation, the network takes x_t, c_{t-1}, h_{t-1} as inputs and produces c_t, h_t through the input (i_t), forget (f_t) and output (o_t) gates via the following intermediate calculations:

$$\mathbf{i}_t = \sigma(\mathbf{W}^i\mathbf{x}_t + \mathbf{U}^i\mathbf{h}_{t-1} + \mathbf{b}^i) \quad (3.1)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}^f\mathbf{x}_t + \mathbf{U}^f\mathbf{h}_{t-1} + \mathbf{b}^f) \quad (3.2)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}^o\mathbf{x}_t + \mathbf{U}^o\mathbf{h}_{t-1} + \mathbf{b}^o) \quad (3.3)$$

$$\hat{\mathbf{c}}_t = \sigma(\mathbf{W}^c\mathbf{x}_t + \mathbf{U}^g\mathbf{h}_{t_1} + \mathbf{b}^g) \quad (3.4)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + i_t \odot \hat{c}_t \quad (3.5)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot tanh(\mathbf{c}_t) \quad (3.6)$$

where $\sigma(\cdot)$ and $tanh(\cdot)$ is the element-wise sigmoid and hyperbolic tangent functions, and $\odot$ denotes element-wise product. $\mathbf{W}^i, \mathbf{W}^f, \mathbf{W}^o, \mathbf{W}^c$ denote the weight matrices of different gates for input $\mathbf{x}_t$, and $\mathbf{U}^i, \mathbf{U}^f, \mathbf{U}^o, \mathbf{U}^c$ are the weight matrices for recurrent hidden state $\mathbf{h}_t$. $\mathbf{b}^i, \mathbf{b}^f, \mathbf{b}^o, \mathbf{b}^c$ denote the bias vectors. As shown in formulation (3.1), we used the LSTM design [47] without peephole connections.

As shown in the gray boxes of Figure 3.1, our word representation learning module includes n units of modules in which n is a hyper-parameter. Each unit includes two BiLSTM layers stacked together followed by a residual layer. First of all, word representations are concatenated from the word embedding and character representations, then fed into the two-layer BiLSTM architecture. Then the residual layer would take the hidden states from both BiLSTM layers and apply the transformation.

The vanilla architecture in [57, 16, 64] includes only one BiLSTM layer. The semantic representations learned from the shallow network is not able to differentiate the variants apart from genes/proteins having similar orthographic

patterns. However, simply increasing the depth of a network will not necessarily improve the performance, and on the contrary, it often leads to a decline in performance beyond a certain point [93]. The introduction of residual could bridge some learned global information to lower layers and facilitate addressing the vanishing gradient problem when training a deeper network [44]. Specifically, we used the identity residual formulated as:

$$y(x) = F(x) + x \tag{3.7}$$

where $F(\cdot)$ is a nonlinear parametric function, and in our case it is the BiLSTM.

3.3.4 Inference Procedure

The CRFs is commonly used for labeling and segmenting sequences tasks, and also has been extensively applied to NER. It is especially helpful for tasks with strong dependencies between token tags. Reimers and Gurevych [81], Yang et al. [110] demonstrated that CRFs could deliver a larger performance increase than the softmax classifier across all NER tasks. Reimers and Gurevych [81] also suggested that a dense layer followed by a linear-chain CRF as a variant CRF classifier would be able to maximize the tag probability of the complete sentence. Specifically, we employed the same variant CRF classifier design as the last layer of the network.

First of all, the transformed representation from the last residual layer for the sequence is mapped with a dense layer and a linear chain CRF layer to the number of tags. The linear-chain CRF maximizes the tag probability of the complete sentence. More formally, given an input sentence x of length N $x = [w_1, w_2, \ldots, w_N]$ in which w_t is the a word in sentence, we predict corresponding variant types $Y = [y_1, y_2, \ldots, y_N]$. The score of a sequence of tags z is defined as:

$$S(x, y, z) = \sum_{t=1}^{N-1} \mathcal{T}_{z_{t-1}, z_t} + \sum_{t=1}^{N} \mathcal{U}_{x_t, z_t} \tag{3.8}$$

where $\mathcal{T}$ is a transition matrix in which $\mathcal{T}_{p,q}$ represents the score of transitioning from tag p to tag q and $\mathcal{U}_{x_t, z_t}$ represents the score of assigning tag z to word w given representation x at time step t. Given the ground truth sequence of tags z, we minimize the negative log-likelihood loss function during the training phase:

$$\begin{aligned} \mathcal{L} &= -log\mathcal{P}(z|x) \\ &= log \sum_{\hat{z} \in \mathcal{Z}} e^{S(x,y,\hat{z})} - S(x, y, z) \end{aligned} \tag{3.9}$$

where $\mathcal{Z}$ is the set of all possible tagging paths. For efficient training and decoding, the Viterbi algorithm is used.

3.4 Experiment Setup

3.4.1 Data

We trained and tested our model on the same datasets from tmVar [103], while 20% of the training is held out for validation.

Data Preprocessing The only preprocessing we performed on the data is tokenization. The conventional tokenization in generic NER tasks would split a sentence by the white space and remove all the digits and special characters. However, those digits and special characters like punctuation are part of the domain-specific entities in biomedical text. Moreover, due to the great variations of those entities, appropriate tokenization is an important preprocessing step for learning biomedical word embeddings. Experimental results show that tokenization can significantly affect the retrieval accuracy, and appropriate tokenization can significantly affect the performance. For example, whether the sequence "(lL-2)" is tokenized to {"(",

"IL-2" and ")"} or {"(IL", "-", and "2)"} would result in considerable difference in representation learning and accuracy. We first tokenize a sentence using white space and characters in "# & $ _ * " ' ; / \∼ ! ? = } { ", then for each token t, if there's any character from , . ':" at the end of t, then strip this character. Finally, strip the brackets if t is bracketed.

Annotation There is no consensus on which annotation scheme is better. The choice varied from applications. Chiu and Nichols [16] demonstrated that BIOES (for Begin, Inside, Outside, End, Single) could achieve considerable performance improvements over BIO (for Begin, Inside, Outside). Lample et al. [57] showed Using BIOES and BIO yields similar performance. Reimers and Gurevych [81] demonstrated that the BIO scheme is preferred over BIOES through extensive experiments on varied NER tasks. Therefore, we adopted the BIO scheme without comparing it with BIOES or other schemes.

3.4.2 Evaluation

One challenge for NER research is establishing an appropriate evaluation metric [72]. In particular, entities may be correctly delimited but misclassified, or entity boundaries may be mismatched. In some generic NER tasks, they would consider partial matching (text offsets overlap, e.g., left match or right match) or oversized boundaries as accurate tagging. However, same as [103], we only considered exact matching (two entities match if their boundaries are identical and tags are correctly classified), and any other prediction was considered as misclassification.

Moreover, there are three types of variants in the tmVar dataset: DNA mutation, protein mutation, and SNP. Therefore, the final set of tags used for training and prediction in our research are B-DNAMutation, I-DNAMutation, B-ProteinMutation, I-ProteinMuatation, B-SNP (no I-SNP), O, and _PAD_ (for padding purpose). To make a fair comparison with other works, we removed the

tag header B- and I-, and only used the tag body with their entity boundaries to calculate precision, recall, and F1 score.

3.4.3 Settings

We implemented our model using Keras with the TensorFlow backend. The computations for a single model are run on Tesla P100 GPU. Table 3.3 summarizes the chosen hyperparameter settings for all DNN models. Moreover, the embedding size for BioW2V is also a hyperparameter, which includes 50, 100, and 200. With respect to the SGD optimizer, besides the common settings, we also set the momentum to 0.9 and used Nesterov.

3.5 Results and Discussion

In this section, we report our experimental results and investigate some key components used in our experiments. We discuss their roles in the low-resource training process.

3.5.1 Results

We compared our proposed DeepVar with state-of-the-art NER systems [57, 64] and variant identification system tmVar [103] and nala [13]. We performed extensive parameter tuning for all generic DNN models using settings shown in Table 3.3, while for vanilla models we used the same setting in [57, 64] on character feature learning and greedily tuned other settings like the word embeddings and optimizer. For tmVar and nala, we quoted their experiment results directly.

The results are reported in Table 4.3. First of all, we observed that the DeepVar model achieves significantly higher F1 scores than state-of-the-art vanilla models, nala, and tmVar ([a,b] without post-processing). DeepVar also achieves appreciably higher F1 scores than generic DNN models. Meanwhile, the result of DeepVar is very close to the best record of tmVar ([c] with extensive hand-crafted features and

Table 3.3 Hyperparameters and Training Settings in Our Experiments

	Parameters	*Values*
char-level configuration	max char length	15, 30, 50
	char emb size	25, 50, 100
	char emb dropout	0, 0.25, 0.5
	char CNN filter size	30, 50, 70
	char CNN window	3, 5, 7
	char LSTM states	25, 50, 100
	char LSTM dropout	0, 0.25, 0.5
word-level configuration	max word length	115
	word emb	BioW2V, BioELMO, BioFlair, BioBert
	word repr. learning unit n	1, 2
	word LSTM states	50, 100, 200
	word LSTM dropout	0, 0.25, 0.5
hidden-layer	hidden states	50, 100, 200
	hidden layer dropout	0, 0.25, 0.5
training and optimization	batch size	32, 64, 128
	optimizer	SGD, RMSP, ADAM
	learning rate	1e-4
	learning rate decay	1e-5
	clipnorm	1.0
	epochs	100

Table 3.4 Results of Comparisons in Our Experiments

Model	*Char Repr*	*Word EMB*	*P(%)*	*R(%)*	*F(%)*
DeepVar	BiLSTM	BioW2V	91.72	89.86	**90.78**
	CNN	BioELMO	90.67	**90.48**	90.58
DNN†	BiLSTM	BioELMO	**91.84**	89.05	90.42
	CNN	BioELMO	90.91	89.25	90.07
Vanilla‡	BiLSTM[57]	BioELMO	88.76	89.66	89.20
	CNN[64]	BioELMO	90.32	87.02	88.64
tmVar (reported in [103])			85.81	80.82	83.24[a]
			92.01	83.72	87.67[b]
			91.38	***91.40***	***91.39***[c]
NALA (reported in [13])			87.00	92.00	89.00[d]

†with greedy tuning

‡same character learning settings, while greedy tuning other settings

[a]using BIO annotation scheme, without post-processing

[b]using 11 hand-crafted annotation scheme, without post-processing

[c]using 11 hand-crafted annotation scheme, with post-processing

[d]using partial match. Performance of the exact match would be lower.

post-processing). However, it is worth noting that DeepVar is a truly end-to-end system without any preprocessing, feature engineering, or post-processing.

Moreover, the DeepVar and generic DNN models differ at the introduction of the residual layer, which is designed to learn better semantic representations by training deeper networks. For the results reported in Table 4.3, the generic models achieved the best performance using the shallow network with one BiLSTM layer while $n = 2$ in DeepVar for word-level representation learning. We also investigated both BiLSTM

and CNN in learning the character-level representation and compared their role in different models. As we can see from Table 4.3, BiLSTM performs better than CNN in all scenarios.

3.5.2 Word Embeddings

In our experiments, we used the weights from pretrained models on biomedical literature to extract the word embeddings for the BioW2V, BioElmo[1], BioFlair[2], and BioBert[3], respectively. More specifically, BioW2V used CBOW word2vec [68] model and was pre-trained on the large up-to-date collection of PubMed corpus. While BioELMO used the concatenated representations from the last three layers, BioFlair took the stacked representations from pubmed-forward and pubmed-backward, while BioBert used the concatenated representations from the last four layers.

The best performance of DeepVar is reported on BioW2V, however, as shown in Table 3.5, the overall performances of BioELMO, BioBert, and BioFlair significantly outperform BioW2V in generic DNN NER models. This interesting observation demonstrated that word2vec can achieve compelling performance in deeper networks. Moreover, the performances of BioBert and BioELMO are very close and slightly better than BioFlair.

3.5.3 Optimizer

For DeepVar training, we observed that Rmsp slightly outperforms Adam while both of them significantly outperform SGD. For generic DNN models, we had the same observation over Rmsp and Adam while SGD has much worse performance. Moreover, SGD is easily failed on training a valid classifier on most settings if using generic models with BioW2V as word embedding input. This observation is significantly

[1]https://allennlp.org/elmo (Accessed on Mar 31, 2020)
[2]https://github.com/zalandoresearch/flair (Accessed on Mar 31, 2020)
[3]SciBert [7] https://github.com/allenai/scibert (Accessed on Mar 31, 2020)

Table 3.5 Comparisons on Pre-trained Word Embeddings

Model	Embedding	P(%)	R(%)	F(%)
DeepVar	BioW2V	91.72	89.86	90.78
	BioELMO	90.67	90.48	90.58
	BioBert	91.49	89.45	90.46
	BioFlair	91.27	89.05	90.14
DNN	BioW2V	87.52	89.47	88.49
	BioELMO	91.84	89.05	90.42
	BioBert	90.97	89.86	90.41
	BioFlair	90.22	89.86	90.04

divergent from knowledge learned from generic NER tasks [57, 64, 81, 110] in which SGD and Adam are preferred over Rmsp.

Table 3.6 Comparisons on Optimizers

Model	Optimizer	P(%)	R(%)	F(%)
DeepVar	SGD	87.45	85.86	86.65
	RMSP	91.72	89.86	90.78
	ADAM	91.84	89.05	90.42
DNN	SGD	82.52	82.35	82.44
	RMSP	91.84	89.05	90.42
	ADAM	88.36	90.87	89.60

3.6 Conclusions

In this chapter, we discussed our DeepVar neural network for biomedical variant entity identification. Despite being simple and not requiring any feature engineering, the

proposed approach achieved comparable performance to the state-of-the-art system. It outperformed other benchmark systems on the low-resource dataset. We also showed through detailed analysis that the performance gain is achieved by the introduced residual, which facilitates to train a deeper network and confirmed the domain-specific contextual word embeddings make significant contributions to the performance gain. Our investigation on key components may also shed light to other deep low-resource applications.

CHAPTER 4

CONSISTENCY-BASED UNSUPERVISED DATA AUGMENTATION FOR NAMED ENTITY RECOGNITION WITH MINIMAL RESOURCES

4.1 Background

Deep learning has accomplished revolutionary achievements on a wide range of Natural Language Processing (NLP) tasks in recent years due to its extraordinary language understanding capability from a large amount of data. Despite their success, those state-of-the-art deep neural networks are generally data-hungry, which builds impassable obstacles for domains without large corpus. The performance could be improved when deep neural models are trained with more data. In certain domains, the shortage of labeled data can be addressed by annotating unlabeled data with crowdsourcing [37, 19, 32] or regular expression matching [5, 115, 116]. However, the labeled data collection can be prohibitive in many real-world scenarios due to the high cost of annotation and/or the scare target of interests, especially when expert annotations are required (i.e. the disease or genomic variant names in biomedical applications), or when personal privacy issues are concerned (i.e. medical clinic notes or sensitive social user profiles). In those scenarios, data augmentation can be an appealing alternative means to produce an adequate amount of new labeled data in a more affordable way.

Various studies have demonstrated the benefits and pitfalls of data augmentation in computer vision [122, 76, 22, 15] and speech recognition [86, 104]. Hernández-García and König [45] exploited techniques on how to increase the number of training examples using domain knowledge, and showed its effectiveness on controlling overfitting as well as improving generalization capability. A common approach is to introduce more realistic-looking noise in data space through creating perturbed synthetic over-samplings from existing examples, like random flip, crop,

rotate, or zoom, RandAugment [23] or GridMask [14]. It is noted that most of the methods are invariant transformations for those applications (i.e., the class of an instance will not be changed after transformation).

However, data augmentation in NLP is a more sophisticated problem as there are no straightforward invariant transformations for texts. Despite the difficulty of obtaining universal invariant rules that can be applied easily and automatically in diversified NLP domains, a few works have been proposed recently to address the problem using various approaches. For example, some methods manipulated the augmentation on feature space [106, 41]. At the same time, more studies focused on augmenting data directly, such as word replacement with synonyms learned from language models or machine translation models. The augmentation policies varied in how to generate diverse paraphrases (e.g., by random sampling [105], word synonyms replacement [100, 35, 53], beam search [33, 54], or back translation [105, 119]).

It is worth noting that almost all of those methods utilized domain-dependent external resources, such as filtering the domain-related instances from an external reference set [66], translating the text by machine translation model [105], and utilizing the existing well-established model in the domain to assist annotation. Such domain dependencies introduce additional constraints on the capacity of adapting the source model to the new domains and can be prohibitively infeasible for under-represented applications with low resources, such as scientific terminologies or machinery logging, as shown in Table 4.1. Even for newswire articles, those external resources or machine translation tools are only regularly available for a few well-studied natural languages like English and French.

Little work in data augmentation has been performed for under-represented domains with low resources. Also this has rarely been considered by the mainstream data-hungry deep learning community. In addition, it is challenging to accomplish data augmentation for token-level tasks in a logical and discriminating manner,

Table 4.1 Examples of Broader Languages for NLP Tasks

Data Source	Example
Newswire Article	Influenza is still going strong in the United States and isn't expected to slow down for at least several more weeks
Scientific Article	The polymorphism rs2234671 at position Ex2+860G>C of the CXCR1 gene causes a conservative amino acid substitution (S276T)
Machinery Logs	214.1.211.251–[15/Apr/2011: 9:39:30 0700] "GET/modules .php?Name=Reviews & rop=post & title =% 253c scriptcomment>alert2528document.cookie%)% 253c/script> HTTP/1.0 "404 316" - "" - "

because it may result in even worse performance using inappropriately augmented data than just using clean data. As shown in Figure 4.1, the top and bottom examples illustrate augmentations on sentence-level and token-level tasks, respectively. For sentence-level tasks, only a single label would be affected for the sentence after augmentation, as shown by the orange color in Figure 4.1. While, for token-level tasks like part-of-speech tagging (POS) or named entity recognition (NER), every word in the corpus needs to be properly labeled. The objective of augmentation for the token-level task is slightly different from the sentence-level task but results in more demanding challenges.

To the best of our knowledge, no comprehensive solution has been proposed yet to tackle the challenges mentioned above in one task. In this research, we proposed a new consistency-based unsupervised data augmentation method as well as a model to train neural network with a combination of a limited amount of clean data and a

larger set of automatically augmented noisy samples in active learning settings. Our augmentation approach and training model employ the following "BCD" principles:

- **B**urden-free Augmentation: our augmentation with adaptive topic alignment is conducted in an unsupervised manner without any requirement of external resources, neither large unlabeled reference set nor machine translation tool. This greatly relieves the burden for under-represented domains with low resources.
- **C**onfidence-based Training: our model training is guided by a confidence-based annealing scheduler. It encourages the model to focus on learning more robust representation confidently without being distracted by the noisy information, which in return can control the gap of training signal learned between labeled clean data and unlabeled noisy data.
- **D**iversity-oriented Labeling: our labeling for noisy augmented data is prioritized by the diversity-based selection, and labels are actively propagated during the training process from teacher-student distillation. It lowers the demand for applications when no well-established model exists to do the machine annotation. It can acquire a higher quality of annotation during active distillation.

The proposed model, coupled with the BCD principles, applies to different token-level scenarios. In this work, we apply it to several NER tasks from varied domains to demonstrate its effectiveness. Our experimental results, on a variety of publicly available datasets, show that it steadily outperforms baselines with minimal resources. Our ablation analysis indicates that the performance improvement is obtained from training with both clean and noisy instances as well as from effectively handling the noise in the data. We also compare it with other recent proposed sentence-level augmentation strategies and discuss more insights on the components in our design.

4.2 Related Work

In this section, we briefly review the recent developments of several research lines we deem to be most relevant to our work.

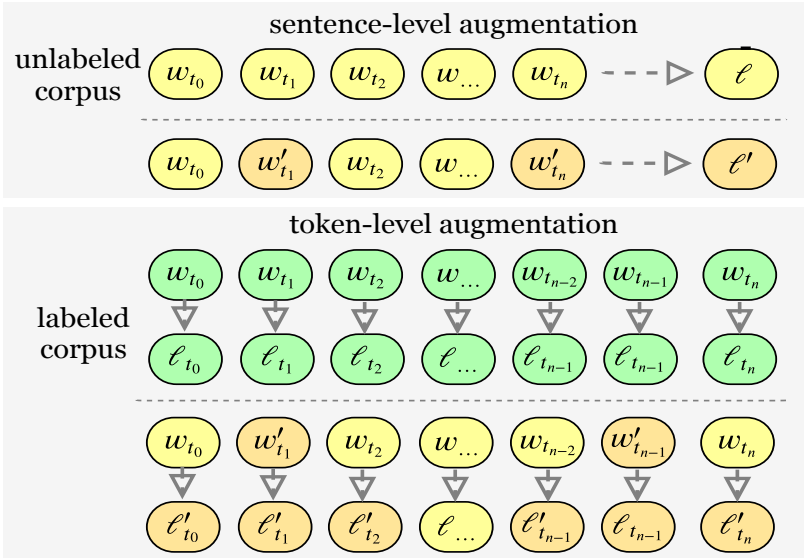

Figure 4.1 Illustration of text augmentation. The yellow color represents unlabeled data, orange color represents affected data by augmentation, and green color represents labeled data. w_t and l are the word and label, respectively.

Knowledge Distillation Knowledge Distillation (KD) was first introduced by Hinton et al. [46] to the deep learning communities, and various forms of extended techniques were discussed comprehensively by [83]. The effective recipe of KD is to compress the knowledge from the huge and computationally expensive teacher model to optimize a simpler student model. Typically, the teacher model can be a cumbersome ensemble of networks [24] or tasks [18]. In our research, we introduce it in the context of data augmentation. Rather than building multiple teacher models or groups of tasks, we particularly train a single (weak) teacher model to generate pseudo-gold labels for noisy augmented instances, and leave the student model to learn more generalized information progressively from differently selected high-valued augmented data on-the-fly.

Active Learning The ultimate goal of Active Learning (AL) is to maximize model performance with minimal labeling cost. It usually consists of iterative procedures that judiciously select unlabeled samples for labeling [38]. Approaches often differ

on sample selection criteria and weak labeling schema. A common practice is to uniformly select a small starting subset of data using heuristic rules for labeling. Human interventions [116] or ensembling [88] are then involved in the loop to acquire weak labels. In our research, we consider the same pool-based AL technique that has been used in Gao et al. [38], where a pool of unlabeled data is initially constructed, and then small batches are iteratively selected to label in conjunction with training. We are also intending to demonstrate that teacher-student distillation can automatically improve the accuracy of weak label propagation.

Semi-Supervised Learning Given the rich variety of Semi- Supervised Learning (SSL) techniques, we focus on recent developments in deep neural networks that are more relevant to our work. Many approaches for SSL have been developed for NLP tasks to train large models with massive data [109, 30, 79, 42]. Similar to AL, SSL can also describe algorithms that seek to improve learning with a small portion of labeled data and a comparably larger set of unlabeled data [118, 90]. Thus they are naturally related and can be combined to improve performance by learning meaningful data representations from the unlabeled pool [98].

However, only a few works have considered combining KD, AL and SSL during training. To the best of our knowledge, very few existing works, if any, focused on the data augmentation that coupled SSL with AL and KD as our approach in this work has been used.

4.3 Modeling and Proposed Framework

In this section, we formulate the model, discuss the details of how we solve those problems with the BCD principles, and train the model in active learning settings.

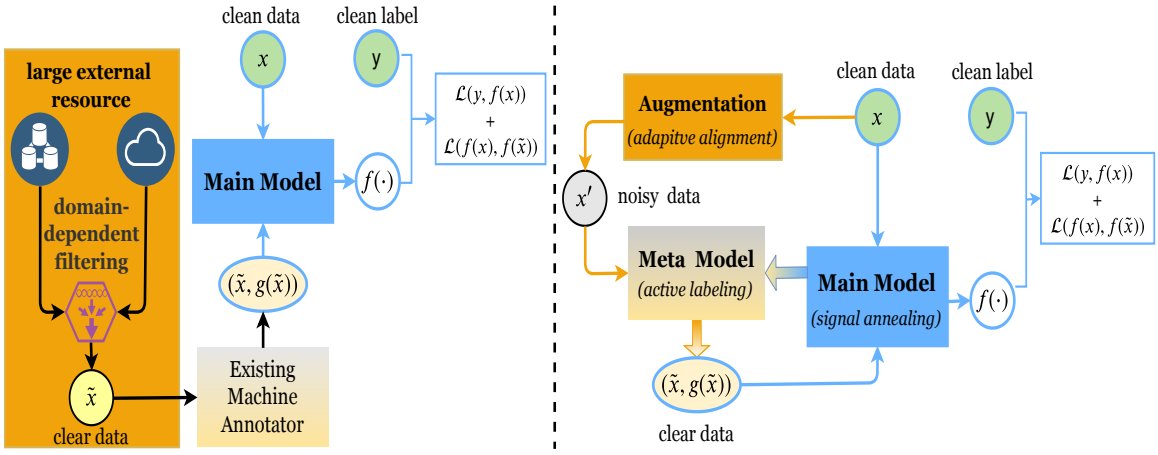

Figure 4.2 Comparison of existing approaches and our model. The green color represents clean data with ground truth, orange color represents component to generate unlabeled data, yellow color represents clear data which is filtered from unlabeled data with annotated weak labels. The pipeline on the left is an illustration of existing approaches (e.g.: [66, 105]) with dependency on large external resources and existing machine annotator; the framework on the right is our proposed model for token-level task without dependency on external resources nor existing machine annotators.

4.3.1 Problem Demarcation

Given a token-level NLP task, our problem of interest consists of three coherent sub-problems:

- *Valid Data Augmentation*: how to decide the proper tokens to be revised selectively to maximize the diversity of augmented context, while preserving the primary information, as well as choose the appropriate tokens to replace them to minimize the risk of altering the true label distribution;
- *Reliable Label Annotation*: how to annotate the new noisy instances in a reliable way to minimize the divergence between distributions of machine labels and true labels without oracle referenced resources nor robust machine annotator;
- *Enforcing Smoothness of Training*: how to enforce the deep neural network on training a robust model consistently and confidently in the combination of limited labeled data and larger noisy augmented data with weak labels.

We proposed three corresponding principles to tackle them, which were mentioned previously, BCD principles. More specifically, they are **B**urden-free

augmentation, **C**onfidence-based training and **D**iversity-oriented labeling, respectively. More importantly, in contrast to previous works, we propose to address the problem for the under-represented domains in low-resource settings. Particularly, we make no presumption that a large unlabeled external dataset is available or a well-established machine learning model should exist for machine annotation.

The high-level framework of our model about how these components are connected is shown on the right side of Figure 4.2. Nodes in green color represent the limited size of clean data D with ground truth, while the node in gray color denotes augmented noisy data A in which $|A| \gg |D|$, and the node in yellow color means distilled clear data B. The component in orange color represents burden-free adaptive augmentation and is discussed in Section 4.3.2; the meta model $g(\cdot)$ tries to annotate clear data from the noisy examples through teacher-student distillation; the main model (in blue color) $f(\cdot)$ is updated through active learning. How to train neural networks consistently, in the combination of limited labeled data and a large amount of noisy augmented data with weak labels, is discussed in Section 4.3.3. The semi-supervised objective function contains two parts: the supervised loss $\mathcal{L}(y, f(x))$ and the divergence $\mathcal{L}(f(x), f(\tilde{x}))$. We discuss the objective functions and the annealing technique, which is used to enforce confident training in Section 4.3.4. Note that the arrows in blue color represent the training procedure with back-propagation involved. It should be mentioned that the parameters for the main models are not changed when distilling the labels on the clear samples. The fully trained main model will be used to predict the unseen test data.

In the rest of this chapter, unless otherwise denoted, we will keep using D, A, B to denote the clean data, augmented noisy data, and distilled clear data, respectively.

4.3.2 Burden-free Augmentation

Intuitively, the goal of text augmentation is to introduce a more diverse context by revising some tokens, while maintaining the nature of original labels. The dilemma is that we want to maximize the possibility to introduce more diversity but, at the same time, minimize the risk of disturbing the original true distribution. Therefore, how to design the augmentation transformation has become critically important.

Formally, let's denote a labeled sentence by $x = \{w_0, w_1, \ldots, w_n\}$, where w_i is the ith word in the given sentence and n is the sentence length. Let $q(x'|x)$ be an augmentation transformation from which one can draw augmented samples $x' = \{w'_0, w'_1, \ldots, w'_n\}$. The desired instances are expected to be diverse and valid. Hence, the augmentation should retain primary information and only replace uninformative words with other words. For $q(x'|x)$, we customize the sentence-level word replacing method in [105] to token-level task, and further extend with adaptive topic alignment.

Our Adaptive Augmentation utilizes the topic model to do the topic alignment for each instance, and then adaptively draw the less informative candidate tokens $\{w_i\}$ for revision by the normalized TFIDF score. The detailed steps of Adaptive Augmentation with Topic Alignment method are shown in Algorithm 1. Specifically, we train a Latent Dirichlet Allocation (LDA) topic model [11] (Algorithm 1: line 1) to align the topic for each sample $x \in D$. Then, for each word w, we calculate a replacement probability based on the `tfidf(w)` score (Algorithm 1: line 5 to line 6). In addition, we further compute a discriminative weight for each word w in sentence x (Algorithm 1: line 8 to line 10) to measure whether a word carries primary information or not by aligning with $\texttt{topic}_x$. When the word $w \in x$ is going to be replaced, we randomly sample an uninformative word $v \in D$, with respect to $\texttt{topic}_x$, from the vocabulary based on the calculated discriminative weights (Algorithm 1: line 15).

Our augmentation transformation $q(x'|x)$, which is built upon the unsupervised TFIDF bag-of-word scheme and LDA topic model, is burden-free to the low-resource

Algorithm 1: Adaptive Augmentation with Topic Alignment $q(x'|x)$

Data: Labeled data $\boldsymbol{D}$;

Output: Augmented set $\boldsymbol{A}$;

Input:

Number of hidden topics $\boldsymbol{h}$; Augmentation temperature $\boldsymbol{t}$;

Number of augmentations (for each instance) $\boldsymbol{m}$;

Initialization: $A = \emptyset$;

Process:

$(\texttt{topics, scores}) \leftarrow LDA(D, h)$;

$(\texttt{tfidf, idf}) \leftarrow TF(D) \cdot IDF(D)$;

forall *sentence* $x \in D$ **do**

  `# compute probability for replacement in sentence` x

  $\texttt{C} = \max_{w \in x} \texttt{tfidf}(w)$; $\texttt{Z}_1 = \sum_{w \in x} (\texttt{C} - \texttt{tfidf}(w))$;

  $\texttt{replace\_prob} = (\texttt{C} - \texttt{tfidf}(w))/\texttt{Z}_1 \cdot \texttt{t}$ for $w \in x$;

  `# compute sampling score for` v `in vocabulary`

  $\texttt{topic}_x \leftarrow LDA.predict(x)$; $\texttt{S}_{\texttt{adapt2x}} \leftarrow \texttt{scores}(\texttt{topic}_x)$;

  $\texttt{Z}_2 = \sum_{v \in D} (\max(\texttt{S}_{\texttt{adapt2x}}(v)) - \texttt{S}_{\texttt{adapt2x}}(v))$;

  $\texttt{Sampling}_{\texttt{prob}} = (\max(\texttt{S}_{\texttt{adapt2x}}(v)) - \texttt{S}_{\texttt{adapt2x}}(v))/\texttt{Z}_2$;

  `# sampling word in` v `to replace word in` x

  forall *j = 0, 1, . . . , m* **do**

    $x' \leftarrow$ **forall** $w \in x$ **do**

      if $\textit{replace\_prob}(w) > \textit{rand\_prob}$ **then**

        $w' \leftarrow random(v \in D, \texttt{Sampling}_{\texttt{prob}})$;

    end

    $A \leftarrow A \cup x'$

  end

end

applications. Despite being simple, the introduced discriminative weights encourage the candidates $\{w_i'\}$ to be adaptable to the topic of each sentence and can avoid selecting the primary keyword for revision.

4.3.3 Diversity-oriented Labeling

To obtain high-quality machine labels, we propose the diversity-oriented labeling algorithm to propagate the labels iteratively through teach-student annealing. We consider the setting of pool-based active learning [38]. Specifically, the unlabeled samples are generated via Algorithm 1.

Diversity-oriented Selection Metric Intuitively, the unsupervised objective can benefit from exploiting samples that can be recognized to some extent by machine annotator or human but not labeled consistently [38]. Thus, with respect to the selection criterion, we propose a token-level diversity-oriented selection metric to selectively prioritize the high-value data $(\tilde{x}, g(\tilde{x}))$. The selection metric is defined as:

$$
\begin{aligned}
\mathcal{C}(A, \mathcal{M}) &= \sum_{x \in D, x' \in A} \epsilon(x, \mathcal{M}) \\
\underset{w_t \in x, \hat{w}_t \in x', x \sim \{x'\}}{\epsilon(x, \mathcal{M})} &= \sum_{t=1}^{|x|} \sum_{\ell=1}^{J} \mathrm{Var}(P(Y = \ell \mid w_t, \mathcal{M}), \\
&\qquad P(\hat{Y} = \ell \mid \hat{w}_{t_1}, \mathcal{M}), \\
&\qquad P(\hat{Y} = \ell \mid \hat{w}_{t_2}, \mathcal{M}), \\
&\qquad \ldots, \\
&\qquad P(\hat{Y} = \ell \mid \hat{w}_{t_m}, \mathcal{M}))
\end{aligned}
\tag{4.1}
$$

where J is the number of response classes; m is the number of augmented examples $\{x'\}$; $x \sim \{x'\}$ are the paired original sentence and augmented examples, respectively.

w_t is word in x and Y is its corresponding predicted label; while $\hat{w}_{t_m}$ is word in x' with label $\hat{Y}$. $\mathcal{M}$ is the model used for annotation;

Labeling Distillation Moreover, with respect to the labeling function $g(x')$ presented in Figure 4.2, a well-established neural network would be utilized to do annotation in previously proposed NLP augmentation tasks. However, in our research, we do not assume such resources are available for the under-represented applications. The limited amount of clean data is not sufficient to train a well-performing model solely for machine annotation. In this work, we employ the "born-again" distillation presented in Clark et al. [18]. In "born-again" networks, the student model has the same model architecture as the teacher model and is expected to outperform the teachers accuracy [18]. Formally, let's define both the teacher model and student model, which share the same architecture and prediction function $f(\cdot)$ as the main model in Figure 4.2. The meta model as annotation machine and its prediction function $g(\cdot)$ can be defined as follows:

$$g(x') = \lambda f(x', \theta_{teacher}) + (1 - \lambda) f(x', \theta_{student}), \tag{4.2}$$

where λ is a weight increasing from zero to one throughout training, and θ are the weights of the model.

Distill Label by Active Learning we demonstrate in Figure 4.3 how we consistently propagate the reliable labeling through active learning by iteratively training student model $f_{\theta_{student}}(\cdot)$ and updating meta model $g(\cdot)$. Concretely, the noisy unlabeled data U are the pool for active learning. The unlabeled data in pool are labeled by $\mathcal{M}$, which is set to $f_{\theta_{student}}(\cdot)$ in our design, and evaluated by the diversity-oriented selection metric (Equation 4.1). The scoring function $\mathcal{C}$ measures the inconsistency across all the perturbations. Then a batch of K prioritized high-value samples is selected into the clear data set B and removed from A. All the

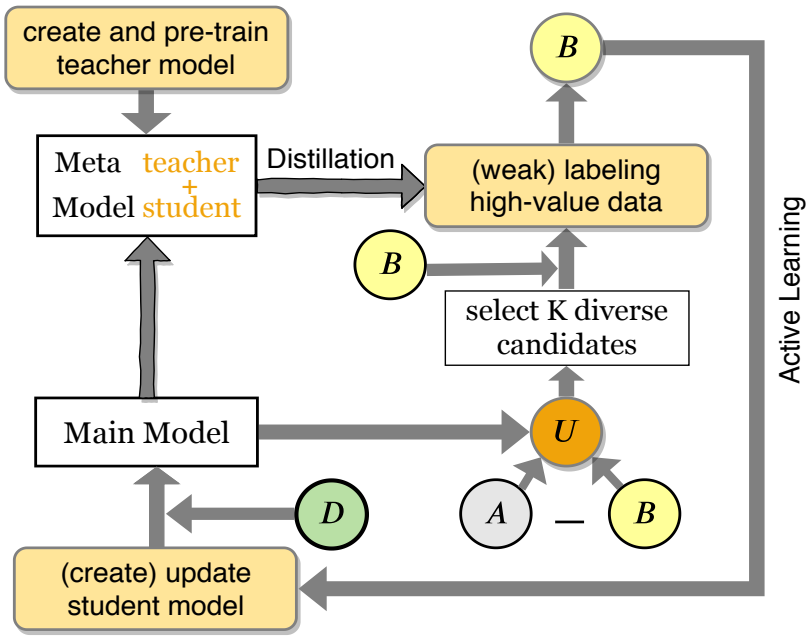

Figure 4.3 Diversity-oriented active labeling.

augmented instances in B are labeled again by labeling Equation (4.2). We update the main model until the model is fully trained, or no more candidates are available.

Early in training, the meta model $g(\cdot)$ is mostly relying on student model $f_{\theta_{student}}$ to make inconsistent labeling such that $f_{\theta_{student}}$ is largely learning to get as useful of a training signal as possible from inconsistent samples. The progressively infused diverse realistic noise by the pool-based active learning setting can enhance the learning of $f_{\theta_{student}}$. Towards the end of the training, the $g(\cdot)$ is mainly relying on the teacher model having more standard labels from $f_{\theta_{teacher}}$. The "born-again" teacher-student annealing not only improves the training of $f_{\theta_{student}}$ and $g(\cdot)$ with more reliable annotations but also enhances $f_{\theta_{student}}$ to outperform $f_{\theta_{teacher}}$ in return, thereby facilitating to train a more robust main model.

4.3.4 Confidence-based Annealing Masking

Our overall training objective, showing in Equation (4.4), is defined as consistency losses in SSL settings, which is to train the neural network in the combination of small

labeled data and extensive unlabelled data. Note that $\mathcal{L}_\ell(x, y, \mathcal{M})$ for the supervised loss can be cross-entropy or other forms of likelihood, depending on the task. In this research, we set it to the negative log-likelihood for the NER task.

Moreover, for an augmentation to be valid, it requires that any example $\tilde{x} \in q(\tilde{x}|x)$ drawn from the distribution shares the same ground-truth as x, i.e., $y(\tilde{w}_t) = y(w_t)$ given any $\tilde{w}_t \in \tilde{x}, w_t \in x$. To enforce such an objective, we minimized the Kullback-Leibler (KL) divergence between the predicted distribution of augmented instances and their original samples to make the $\{f(\tilde{w}_t), t \in [0, n]\}$ approximate $\{y_t, t \in [0, n]\}$. More specifically, we defined it as follows:

$$\begin{aligned} \underset{x \in D; \tilde{x} \in B}{\mathcal{L}_u(x, \tilde{x}, \mathcal{M})} =& \mathcal{L}_u(y, f(\tilde{x})) \\ =& \mathcal{D}_{KL}(p_{\tilde{\theta}}(f(x)|x) \| p_\theta(f(\tilde{x})|\tilde{x})), \end{aligned} \tag{4.3}$$

where $\|$ is the KL measure of two probability distributions, $\tilde{\theta}$ is a copy of the current parameters θ of model $f(\cdot)$, indicating that the gradient is not propagated through $\tilde{\theta}$, as presented in Xie et al. [105].

Furthermore, a good model should be invariant to any small perturbations that do not change the nature of a sample. Here, we employ the Training Signal Annealing (TSA) technique in Xie et al. [105] to encourage the model to focus on the confident representation but not disturbing by the less confident signals, especially the noisy signal from augmented examples. We integrate this technique into our architecture by referring to the authors' codebase [1]. And for the integrity of our work, we reiterate the formulation of three schedulers here:

$$\eta_t = \alpha_t * (1 - \frac{1}{J}) + \frac{1}{J} \begin{cases} \alpha_t = 1 - exp(-\frac{t}{T} * 5) & : log \\ \alpha_t = \frac{t}{T} & : linear \\ \alpha_t = exp((\frac{t}{T} - 1) * 5) & : exp \end{cases} \tag{4.9}$$

[1] https://github.com/google-research/uda (Accessed on Mar 31, 2020)

Algorithm 2: Train our Model in Active Learning

Data: Labeled data $\boldsymbol{D}$; Augmented unlabeled data pool $\boldsymbol{A}$;

Result: Main model $\boldsymbol{M}$;

Input: Active learning batch size $\boldsymbol{K}$;

Initialization:

$\mathrm{U}_0 = A$;

$\mathrm{L}_0 = \{(x, y) : (x, y) \in D\}$;

Process:

forall $t = 0, 1, \ldots, T-1$ **do**

(training main model - SSL)

$$\mathcal{M}_t \leftarrow \arg\min_{\mathcal{M}} \{ \frac{1}{|D|} \sum_{(x,y)\in D} \mathcal{L}_\ell(x, y, \mathcal{M}) + \frac{1}{|L_t|} \sum_{x \in L_t} \mathcal{L}_u(x, \mathcal{M}) \} \tag{4.4}$$

(selection of clear data)

$$B_t \leftarrow \arg\max\{\mathcal{C}(U_t, \mathcal{M}_t)\} \quad \text{s.t. } |B_t| = K, \mathcal{C} = \textit{Equation } (4.1) \tag{4.5}$$

(labeling for clear data - KL)

$$J_t \leftarrow \lambda f(\tilde{x}, M_t) + (1-\lambda) f(\tilde{x}, M_{t-1}) \quad \text{w.r.t. } (\tilde{x} \in B) \tag{4.6}$$

(labeled data update)

$$L_{t+1} \leftarrow L_t \cup \{B, J_t\} \tag{4.7}$$

(unlabeled pool update)

$$U_{t+1} \leftarrow U_t B_t \tag{4.8}$$

end

where J is the number of response classes, t is the current step, T is the global step. This policy works with minimizing the loss of the model (Equation 4.4) during the training procedure, which is summarized in Algorithm 2 of which we show detailed steps for the framework. Since it is costly to train LDA on-the-fly during training, we generate the augmented examples offline. Multiple augmented examples are generated for each sample. In rare cases, some augmented instances would be duplicated, and we remove them from the pool A.

4.4 Experiments and Results

We present the data, experiments and results in this section. Our ablation study teases apart the contribution of each component in Sections 4.3.2, 4.3.3 and 4.3.4.

4.4.1 Datasets

For our study, we selectively evaluate the effectiveness and generalization ability of our model on two previously published benchmarks in the biomedical domain including NCBI disease corpus [31] and Genomic Variant corpus tmVar [103], of which the language exhibits different levels of exotic linguistic heterogeneity from newswire articles. It could inspire other similar NLP tasks like tweets and machinery logging, which are less likely to have substantial domain-dependent resources compared with generic NLP tasks. The statistics of the data are reported in Table 4.2. We used the standard split of training/validation/test sets, and the hyperparameters were tuned based on the performance on the validation set.

4.4.2 Baseline NER Model

Recent high-performing neural architecture for NER tasks is BiLSTM-CRF [16, 57]. The latest improvements mainly stem from using new types of representations learned from character-level embeddings [64, 61] and contextualized embeddings derived from language models pre-trained on large unlabeled corpus like ELMO [78] and BERT

Table 4.2 Statistics of Different Datasets

Data	Size	Types and Counts	Entity Examples
NCBI	8,336 sentences	Disease (6,881)	MCF-7 tumours; sporadic T-PLL
tmVar	4,783 sentences	Protein Variant (653) DNA Variant (751) SNP (136)	p.Pro246HisfsX13; Ex2+860G>C; rs2234671

[29]. Unless otherwise indicated, we use the BiLSTM-CNN-CRF model proposed in [64] to initialize the teacher and student models in all experiments.

For the character-level CNN encoder, we use single-layer CNNs with 50 filters and kernel width three. For the LSTM word-level encoder, we use a single-layer model with 100 hidden units. Dropout rates are all set as 0.5. We used the 200 dimensional Word2vec [67] embedding trained by [85] on PubMed, PubMed Central (PMC) and Wikipedia text. We didn't fine-tune the word embedding in our experiments. Finally, we used RMSP [97] as the optimizer and uniformly set the step size as 0.001 and the batch size as 64.

Our goal is to evaluate the effectiveness of the proposed model and BCD principles with minimal resources. We didn't perform the cost extensive hyper-parameter tuning on the BiLSTM-CNN-CRF architecture. Noted without further exceptions, we uniformly use the same set of hyperparameters for all the teacher and student models in our experiments. We first trained the baseline NER model for 100 epochs with early stopping. The pretrained model will be regarded as the baseline in our experiments. Moreover, we also used the baseline model to define the teacher model described in Section 4.3.3.

4.4.3 Hyperparameters

While we sealed the hyperparameters for the baseline model, we still introduced a few hyperparameters in our model to guide optimization with unlabeled noisy data. We focused on exploring a few critical factors in our experiments and fixed some less critical ones.

More specifically, we set the number of hidden topics h for LDA model in our experiment to the number of tags for each task, with the number of unlabeled augmentations m as 20, and the batch size K for active learning in Algorithm 2 as $|A|/epochs$. In our experiments, we set the number of epochs to train the student model as half of the epochs to train the baseline model, which is $100/2 = 50$. We focused on investigating the effects of the augmentation temperature t in Algorithm 1, the distillation weight λ in Equation (4.2) and the choice of annealing scheduler in Equation (4.9). Moreover, while we use CRF to make the final prediction, we replace the CRF layer with a softmax function to produce the probability score for Equation (4.2) and 4.3.

We fix the seed in our experiments and report the F1 score for each method, which is standard for NER tasks. It is important to note that we evaluate the exact match in all the results. All the experiments are implemented in Keras 2.3.1 with TensorFlow backend 2.0 using Python 3.6.8. All the code will be publicly available after the double-blind review process.

4.4.4 Main Results

When training with unlabeled data, the noise can easily undermine training and performance. As the first step, we try to verify the fundamental idea of the model and BCD principles. Based on the NCBI-disease and tmVar, we compare the performance of our method with two recently proposed text augmentation approaches:

- AWR: our adaptive replacement method in Section 4.3.2;

- TFIDF: TFIDF replacement method in [105];
- Random: replace a token with a random token uniformly sampled from the vocabulary [101].

We also compared our training strategy with another recent annealing technique:

- TSA: the training signal annealing [105];
- CD: our diversity-oriented active labeling principle in combination with TSA.

We report the results in Table 4.3. The baseline is defined and described in Section 4.4.2, which is also the teacher model in our model. The other results are reported from the student model by searching the parameter space described in Section 4.4.3. As we can see from the results, our model can significantly outperform the baseline and other recently proposed strategies, when training with the noisy augmented data. It demonstrates that our framework can guide the training to learn informative representations from the augmented set, and in return, improve the performance of the baseline model.

More specifically, as we can see from Table 4.3, our AWR consistently outperforms the Random and TFIDF augmentation methods across two data sets in various settings. We argue that it is because our AWR can adaptively align the topic for each instance; therefore it can keep more informative signals. More encouragingly, when training under the guidance of our CD principles, all the augmentation strategies, including random sampling, can achieve marginal performance improvement while our AWR achieves the best record. And TSA can achieve marginal performance gains when working with our AWR. It failed to achieve consistent performance improvement across two data sets with random sampling and TFIDF. It indicates that our active labeling algorithm substantially improves the quality of weak labels, and in return, facilitates training a more robust model.

Table 4.3 Results and Comparisons

Data Set	Methods	Precision	Recall	F1 Score
NCBI-Disease	Baseline	**82.32**	74.45	78.18
	Random-TSA	77.8	77.07	77.43
	TFIDF-TSA	78.1	77.54	77.82
	AWR-TSA	78.85	78.45	78.65
	Random-CD	79.56	75.1	77.27
	TFIDF-CD	81.43	80.25	80.84
	AWR-CD	81.22	**81.39**	**81.31**
tmVar	Baseline	**79.22**	79.91	79.57
	Random-TSA	74.20	85.10	79.28
	TFIDF-TSA	76.15	86.17	80.85
	AWR-TSA	72.65	**88.34**	79.73
	Random-CD	75.87	84.23	79.83
	TFIDF-CD	76.76	84.88	80.61
	AWR-CD	79.18	82.94	**81.01**

4.4.5 Component Analysis

We also study the effect of each component in our design and try to understand to what extent they can improve the task.

Distillation Strategy First of all, we study the strategies to distill knowledge defined in Equation (4.2). In our experiments, we tried two approaches: (1) annealing, and (2) ensemble. For the annealing approach, the λ_t is set to $\frac{t}{T}$, where the t is the current time step, and T is the global step. For the ensemble approach, we set the λ_t to a fixed value 0.5 across all the time steps during training. The results are reported in Table 4.4. It turns out that the ensemble method works better than

the annealing distillation. One potential reason is that the teacher model in our experiments essentially is still a weak learner without tuning any hyperparameter, and cannot provide enough gold knowledge at the end of the training. Setting too much weight for the teacher model in the second half of the training stage may introduce some weak labels from the weak teacher model.

Table 4.4 Comparison of Different Distill Strategies

Data Set	Distillation	Augment	Precision	Recall	F1 Score
NCBI-Disease	annealing	Random	75.40	77.36	76.37
		TFIDF	81.43	80.25	80.84
		AWR	79.53	81.10	80.31
	ensemble	Random	79.56	75.1	77.27
		TFIDF	80.32	81.33	80.82
		AWR	81.22	81.39	**81.31**
tmVar	annealing	Random	76.08	83.80	79.75
		TFIDF	77.4	83.59	80.37
		AWR	76.76	84.88	80.61
	ensemble	Random	75.87	84.23	79.83
		TFIDF	76.76	84.88	80.61
		AWR	79.18	82.94	**81.01**

The TSA Scheduler Lastly, we investigated the effect of the training signal annealing scheduler in our experiments. As shown in Table 4.5, on NCBI-disease, both TFIDF and AWR prefer the linear scheduler, while on tmVar, they both favor the log scheduler. The difference between those schedulers is the speed of releasing supervised training signals. More specifically, the log scheduler will release the training signal rapidly at the beginning of the training, while the exp scheduler does

the opposite. The linear scheduler will release the signal progressively along with the training. The results in Table 4.4 indicate that, when the size of the corpus is severely restricted like tmVar, the training can benefit from the quickly released supervised training signal from the log scheduler. However, when the data is of medium size like NCBI-disease, the model can learn more from the progressively released training signals by linear scheduler where the amount of noisy data won't overwhelmingly flush into the training.

Table 4.5 Ablation Results on TSA Scheduler

Data Set	Scheduler	Augment	Precision	Recall	F1 Score
NCBI-Disease	exp	TFIDF	80.73	80.81	80.77
		AWR	80.85	79.42	80.13
	linear	TFIDF	81.43	80.25	<u>80.84</u>
		AWR	81.22	81.39	**81.31**
	log	TFIDF	80.33	81.33	80.82
		AWR	79.53	81.10	80.31
tmVar	exp	TFIDF	77.4	83.59	80.37
		AWR	72.08	88.12	79.30
	linear	TFIDF	76.62	84.23	80.25
		AWR	74.81	84.66	79.43
	log	TFIDF	76.76	84.88	<u>80.62</u>
		AWR	79.18	82.94	**81.01**

4.4.6 Retraining Cost

At the end of our experiments, we further investigated the retaining effect in the active learning setting. Traditional active learning schemes are expensive for deep learning because they require complete retraining after each round with newly annotated

samples. Since retraining from scratch in each round is not practical for deep learning, Shen et al. [88] proposed to carry out incremental training with each batch of new labels and update neural network weights for a small number of epochs before querying new labels. In our experiment, we tried two different retraining strategies: (1) retraining single epoch in each round; (2) retraining ten epochs in each round. As the results shown in Table 4.6, our model can achieve a significant performance increment even with the single retrain epoch. Further retraining with more epochs can only achieve marginal gains but with significantly higher retraining cost, as shown in the last column.

Table 4.6 Comparisons on Retrain-cost for Active Labeling

Data Set	Retain Epoch	Precision	Recall	F1 Score	AvgHour
NCBI-Disease	1	81.62	80.69	81.15	5.5
	10	81.22	81.39	81.31	17.5
tmVar	1	76.76	84.88	80.61	2.5
	10	79.18	82.94	81.01	4.5

To this end, we demonstrate the effectiveness and significance of our model.

4.5 Conclusion

In this chapter, we discussed a novel consistent-based unsupervised data augmentation approach with the "**B**urden-free, **C**onsistent-based, and **D**iversity-oriented" training principle. Our method can train a model using a combination of small-sized clean data and large-sized noisy data, which leads to consistent and significant performance improvement. Our extensive experiments demonstrated the superiority of our method and confirmed that enhanced data augmentation, with proper training guidance, could boost performance significantly.

More importantly, our method offers a competitive lightweight alternative for under-represented domains with limited resources. Our augmentation approach is burden-free and domain-independent. Our active labeling algorithm eliminates the dependency on well-established machine annotators, which may not always exist. We hope that our encouraging results can inspire more future research to investigate the challenges for NLP tasks in resource-limited environments.

CHAPTER 5

CONCLUSION

In this dissertation, we considered bridging the gap between deep learning and domain-specific text mining applications by utilizing different techniques of transfer learning. We explored the pretrain-finetune and knowledge distillation with other deep learning techniques for improving the performance of two specific domains: success prediction in crowdfunding and named entity recognition in biomedical literature.

In Chapter 2, we took the multimodal approach with pretrain-finetune efforts towards enhancing success prediction in crowdfunding projects. While other work in this field focused on utilizing rich post-launch dynamic information from crowdfunding and social media platforms, we wanted to make the prediction before the projects are launched to avoid predictable failure. We integrated knowledge from different modalities collaboratively, like text and images, while limited resources are available at the pre-launch stage. In particular, we acquired the feature representation by transferring the domain knowledge from the large benchmark, Wikipedia and ImageNet, for text and images, respectively. We implemented a multimodal deep learning framework to combine different modalities simultaneously to predict project success. We demonstrated the effectiveness of doing this by evaluating our approach on a large collection of project profiles.

In Chapter 3, we addressed the named entity recognition problem in a specific biomedical application, genomic variant identification, of which only restricted training corpus is available. This task also exposes highly heterogeneous linguistic patterns of entities. The traditional machine learning approaches heavily relied on hand-crafted features, while we wanted to build an end-to-end approach in fully automated construction. We explored several generic NER models and their ability

to identify variants. To further improve the performance, we built a DeepVar model that integrated the residual technique and pretrain-finetune principle to support the training of a deep model. We also explored and demonstrated the effectiveness of different word embedding benchmarks in low-resource settings.

In Chapter 4, we introduced the consistency-based unsupervised data augmentation model, which aimed to tackle the data insufficiency situation for low-resource applications in general, and also facilitate to address the overfitting problem when training the deep learning model in low-resource settings. It's the first proposed model which is designed with minimal resources in low-resource settings in this field. Compared to previous text data augmentation approaches, this model doesn't rely on any large external resources or assume any robust annotation machine exists. We defined three principles: *adaptive, active, annealing*, and described all the elements of those three modules. All the modules are jointly optimized together, utilizing active learning and knowledge distillation techniques. Through empirical experiments, we examined and demonstrated the effectiveness and efficiency of the model on two low-resource biomedical applications.

Altogether, we believe transfer learning is a promising technique to address the challenges in domain-specific applications. At the same time, we are still facing enormous challenges in many unique domain-specific problems. One key challenge is the limitation of the size of labeled data in those domains where labels are cost-prohibitive to obtain. Often this occurs due to the high cost of expert annotation or scarceness of the target. In the future, we will have to address the challenges for those domains, rather than just answering problems for large corpus, to bridge the gap between the generic deep learning community and the domain-specific tasks.

REFERENCES

[1] Ajay K Agrawal, Christian Catalini, and Avi Goldfarb. The geography of crowdfunding. Technical report, National Bureau of Economic Research, Cambridge, MA, 2011.

[2] Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. Flair: An easy-to-use framework for state-of-the-art nlp. In *The North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, 2019.

[3] Tim Althoff and Jure Leskovec. Donor retention in online crowdfunding communities: A case study of donorschoose. org. In *The 24th international conference on world wide web*, pages 34–44. International World Wide Web Conferences Steering Committee, 2015.

[4] Tim Althoff, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. How to ask for a favor: A case study on the success of altruistic requests. In *International Conference on Web and Social Media*, 2014.

[5] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. Active learning of regular expressions for entity extraction. *IEEE Transactions on Cybernetics*, 48(3):1067–1080, 2017.

[6] Paul Belleflamme, Thomas Lambert, and Armin Schwienbacher. Crowdfunding: An industrial organization perspective. In *The Workshop Digital Business Models: Understanding Strategies*, pages 25–26. Citeseer, 2010.

[7] Iz Beltagy, Arman Cohan, and Kyle Lo. Scibert: Pretrained contextualized embeddings for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.

[8] Michael Bennet, Scott Brown, Jeff Merkley, and Mary Landrieu. Crowdfund act(s. 2190), 2012.

[9] Adam L Berger, Vincent J Della Pietra, and Stephen A Della Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996.

[10] David M Blei and Michael I Jordan. Modeling annotated data. In *The 26th annual international ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 127–134, 2003.

[11] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

[12] J Gregory Caporaso, William A Baumgartner Jr, David A Randolph, K Bretonnel Cohen, and Lawrence Hunter. Mutationfinder: a high-performance system for extracting point mutation mentions from text. *Bioinformatics*, 23 (14):1862–1865, 2007.

[13] Juan Miguel Cejuela, Aleksandar Bojchevski, Carsten Uhlig, Rustem Bekmukhametov, Sanjeev Kumar Karn, Shpend Mahmuti, Ashish Baghudana, Ankit Dubey, Venkata P Satagopam, and Burkhard Rost. nala: text mining natural language mutation mentions. *Bioinformatics*, 33(12):1852–1858, 2017.

[14] Pengguang Chen. Gridmask data augmentation. *arXiv preprint arXiv:2001.04086*, 2020.

[15] Shuxiao Chen, Edgar Dobriban, and Jane H Lee. Invariance reduces variance: Understanding data augmentation in deep learning and beyond. *arXiv preprint arXiv:1907.10905*, 2019.

[16] Jason PC Chiu and Eric Nichols. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4: 357–370, 2016.

[17] Dan C Ciresan, Ueli Meier, Jonathan Masci, Luca Maria Gambardella, and Jürgen Schmidhuber. Flexible, high performance convolutional neural networks for image classification. In *International Joint Conference on Artificial Intelligence*, volume 22, page 1237. Barcelona, Spain, 2011.

[18] Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D Manning, and Quoc Le. Bam! born-again multi-task networks for natural language understanding. In *The 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, 2019.

[19] Robert A Cochran, Loris D'Antoni, Benjamin Livshits, David Molnar, and Margus Veanes. Program boosting: Program synthesis via crowd-sourcing. In *The 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 677–688, 2015.

[20] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.

[21] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, volume 1, pages 1–2. Prague, 2004.

[22] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

[23] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical data augmentation with no separate search. *arXiv preprint arXiv:1909.13719*, 2019.

[24] Jia Cui, Brian Kingsbury, Bhuvana Ramabhadran, George Saon, Tom Sercu, Kartik Audhkhasi, Abhinav Sethy, Markus Nussbaum-Thom, and Andrew Rosenberg. Knowledge distillation across ensembles of multilingual models for low-resource languages. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4825–4829. IEEE, 2017.

[25] Sun Daoyuan. *What is the recipe for success? An empirical analysis of crowdfunding project performance*. PhD thesis, 2016.

[26] Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Benjamin L King, Roy McMorran, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. The comparative toxicogenomics database: update 2017. *Nucleic Acids Research*, 45(D1):D972–D978, 2016.

[27] Franck Dernoncourt, Ji Young Lee, Ozlem Uzuner, and Peter Szolovits. De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association*, 24(3):596–606, 2017.

[28] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *The North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[30] Bhuwan Dhingra, Danish Pruthi, and Dheeraj Rajagopal. Simple and effective semi-supervised question answering. *arXiv preprint arXiv:1804.00720*, 2018.

[31] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10, 2014.

[32] Anca Dumitrache, Lora Aroyo, and Chris Welty. Crowdsourcing ground truth for medical relation extraction. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 8(2):11, 2018.

[33] Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. Understanding back-translation at scale. *arXiv preprint arXiv:1808.09381*, 2018.

[34] Vincent Etter, Matthias Grossglauser, and Patrick Thiran. Launch hard or go home!: predicting the success of kickstarter campaigns. In *The first ACM Conference on Online Social Networks*, pages 177–182. ACM, 2013.

[35] Marzieh Fadaee, Arianna Bisazza, and Christof Monz. Data augmentation for low-resource neural machine translation. *arXiv preprint arXiv:1705.00440*, 2017.

[36] Jiangfan Feng, Yuanyuan Liu, and Lin Wu. Bag of visual words model with deep spatial features for geographical scene classification. *Computational Intelligence and Neuroscience*, 2017, 2017.

[37] Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. Annotating named entities in twitter data with crowdsourcing. In *The NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88. Association for Computational Linguistics, 2010.

[38] Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan O Arik, Larry S Davis, and Tomas Pfister. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. *arXiv preprint arXiv:1910.07153*, 2019.

[39] Elizabeth M Gerber, Julie S Hui, and Pei-Yi Kuo. Crowdfunding: Why people are motivated to post and fund projects on crowdfunding platforms. In *The International Workshop on Design, Influence, and Social Technologies: Techniques, Impacts and Ethics*, volume 2. ACM New York, NY, 2012.

[40] Michael D Greenberg, Bryan Pardo, Karthic Hariharan, and Elizabeth Gerber. Crowdfunding support tools: predicting success & failure. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 1815–1820. ACM, 2013.

[41] Xifeng Guo, En Zhu, Xinwang Liu, and Jianping Yin. Deep embedded clustering with data augmentation. In *Asian conference on Machine Learning*, pages 550–565, 2018.

[42] Suchin Gururangan, Tam Dang, Dallas Card, and Noah A Smith. Variational pretraining for semi-supervised text classification. *arXiv preprint arXiv:1906.02242*, 2019.

[43] Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48, 2017.

[44] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[45] Alex Hernández-García and Peter König. Data augmentation instead of explicit regularization. *arXiv preprint arXiv:1806.03852*, 2018.

[46] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[47] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[48] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[49] Julie Hui, Michael Greenberg, and Elizabeth Gerber. Understanding crowdfunding work: implications for support tools. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 889–894. ACM, 2013.

[50] Qiao Jin, Bhuwan Dhingra, William W Cohen, and Xinghua Lu. Probing biomedical embeddings from language models. *arXiv preprint arXiv:1904.02181*, 2019.

[51] Douwe Kiela and Léon Bottou. Learning image embeddings using convolutional neural networks for improved multi-modal semantics. In *The 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 36–45, 2014.

[52] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[53] Sosuke Kobayashi. Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*, 2018.

[54] Wouter Kool, Herke Van Hoof, and Max Welling. Stochastic beams and where to find them: The gumbel-top-k trick for sampling sequences without replacement. *arXiv preprint arXiv:1903.06059*, 2019.

[55] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? *arXiv preprint arXiv:1805.08974*, 2018.

[56] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[57] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[58] Jeroen FJ Laros, André Blavier, Johan T den Dunnen, and Peter EM Taschner. A formalized description of the standard human variant nomenclature in extended backus-naur form. *BMC bioinformatics*, 12(4):S5, 2011.

[59] Yan Li, Vineeth Rakesh, and Chandan K Reddy. Project success prediction in crowdfunding environments. In *The Ninth ACM International Conference on Web Search and Data Mining*, pages 247–256. ACM, 2016.

[60] Hongfang Liu, Zhang-Zhi Hu, Jian Zhang, and Cathy Wu. Biothesaurus: a web-based thesaurus of protein and gene names. *Bioinformatics*, 22(1):103–105, 2005.

[61] Liyuan Liu, Jingbo Shang, Xiang Ren, Frank Fangzheng Xu, Huan Gui, Jian Peng, and Jiawei Han. Empower sequence labeling with task-aware neural language model. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[62] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[63] Lin Ma, Zhengdong Lu, and Hang Li. Learning to answer questions from image using convolutional neural network. In *AAAI*, volume 3, page 16, 2016.

[64] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *The 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, 2016.

[65] Valentin Malykh, Varvara Logacheva, and Taras Khakhulin. Robust word vectors: Context-informed embeddings for noisy texts. In *The 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 54–63, 2018.

[66] Joel Mathew, Shobeir Fakhraei, and José Luis Ambite. Biomedical named entity recognition via reference-set augmented bootstrapping. *arXiv preprint arXiv:1906.00282*, 2019.

[67] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[68] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

[69] Tanushree Mitra and Eric Gilbert. The language that gets people to give: Phrases that predict success on kickstarter. In *The 17th ACM conference on Computer Supported Cooperative Work & Social Computing*, pages 49–61. ACM, 2014.

[70] Sunil Mohan, Nicolas Fiorini, Sun Kim, and Zhiyong Lu. A fast deep learning model for textual relevance in biomedical information retrieval. In *The 2018 World Wide Web Conference*, pages 77–86. International World Wide Web Conferences Steering Committee, 2018.

[71] Ethan Mollick. The dynamics of crowdfunding: An exploratory study. *Journal of business venturing*, 29(1):1–16, 2014.

[72] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[73] Nona Naderi and René Witte. Automated extraction and semantic analysis of mutation impacts from the biomedical literature. In *BMC genomics*, volume 13, page S10. BioMed Central, 2012.

[74] Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. *arXiv preprint arXiv:1906.05474*, 2019.

[75] 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.

[76] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[77] Florent Perronnin and Diane Larlus. Fisher vectors meet neural networks: A hybrid classification architecture. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 3743–3752, 2015.

[78] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *the North American Chapter of the Association for Computational Linguistics*, 2018.

[79] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.

[80] Arnau Ramisa Ayats. Multimodal news article analysis. In *The Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 5136–5140, 2017.

[81] Nils Reimers and Iryna Gurevych. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*, 2017.

[82] Stephen Roller and Sabine Schulte Im Walde. A multimodal lda model integrating textual, cognitive and visual modalities. In *The 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1146–1157, 2013.

[83] Fabian Ruffy and Karanbir Chahal. The state of knowledge distillation for classification. *arXiv preprint arXiv:1912.10850*, 2019.

[84] Gerard Salton and Michael J McGill. Introduction to modern information retrieval. 1986.

[85] Hans Moen Tapio Salakoski Sampo Pyysalo, Filip Ginter and Sophia Ananiadou. Distributional semantics resources for biomedical text processing. *Proceedings of LBM*, pages 39–44, 2013.

[86] Jan Schlüter and Thomas Grill. Exploring data augmentation for improved singing voice detection with neural networks. In *International Society for Music Information Retrieval*, pages 121–126, 2015.

[87] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813, 2014.

[88] Yanyao Shen, Hyokun Yun, Zachary C Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*, 2017.

[89] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[90] Edwin Simpson, Jonas Pfeiffer, and Iryna Gurevych. Low resource sequence tagging with weak labels. *The AAAI Conference on Artificial Intelligence*, 2020.

[91] Jacob Solomon, Wenjuan Ma, and Rick Wash. Don't wait!: How timing affects coordination of crowdfunding donations. In *The 18th acm conference on Computer Supported Cooperative Work & Social Computing*, pages 547–556. ACM, 2015.

[92] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[93] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[94] Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. Fast and accurate entity recognition with iterated dilated convolutions. *arXiv preprint arXiv:1702.02098*, 2017.

[95] Fei Tan, Zhi Wei, Jun He, Xiang Wu, Bo Peng, Haoran Liu, and Zhenyu Yan. A blended deep learning approach for predicting user intended actions. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 487–496. IEEE, 2018.

[96] Philippe Thomas, Tim Rocktäschel, Jörg Hakenberg, Yvonne Lichtblau, and Ulf Leser. Seth detects and normalizes genetic variants in text. *Bioinformatics*, Jun 2016. doi: 10.1093/bioinformatics/btw234.

[97] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2):26–31, 2012.

[98] Katrin Tomanek and Udo Hahn. Semi-supervised active learning for sequence labeling. In *The Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1039–1047. Association for Computational Linguistics, 2009.

[99] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 5005–5013, 2016.

[100] William Yang Wang and Diyi Yang. Thats so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *The 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563, 2015.

[101] Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. Switchout: an efficient data augmentation algorithm for neural machine translation. *arXiv preprint arXiv:1808.07512*, 2018.

[102] Xuan Wang, Yu Zhang, Xiang Ren, Yuhao Zhang, Marinka Zitnik, Jingbo Shang, Curtis Langlotz, and Jiawei Han. Cross-type biomedical named entity recognition with deep multi-task learning. *Bioinformatics*, 35(10):1745–1752, 2018.

[103] Chih-Hsuan Wei, Bethany R Harris, Hung-Yu Kao, and Zhiyong Lu. tmvar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*, 29(11):1433–1439, 2013.

[104] Sebastien C Wong, Adam Gatt, Victor Stamatescu, and Mark D McDonnell. Understanding data augmentation for classification: when to warp? In *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–6. IEEE, 2016.

[105] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation. *arXiv preprint arXiv:1904.12848*, 2019.

[106] Ziang Xie, Sida I Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y Ng. Data noising as smoothing in neural network language models. *arXiv preprint arXiv:1703.02573*, 2017.

[107] Guohai Xu, Chengyu Wang, and Xiaofeng He. Improving clinical named entity recognition with global neural attention. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, pages 264–279. Springer, 2018.

[108] Jun Xu, Yaoyun Zhang, Hua Xu, et al. Clinical abbreviation disambiguation using neural word embeddings. *Proceedings of BioNLP 15*, pages 171–176, 2015.

[109] Weidi Xu, Haoze Sun, Chao Deng, and Ying Tan. Variational autoencoder for semi-supervised text classification. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[110] Jie Yang, Shuailong Liang, and Yue Zhang. Design challenges and misconceptions in neural sequence labeling. *arXiv preprint arXiv:1806.04470*, 2018.

[111] Wonjin Yoon, Chan Ho So, Jinhyuk Lee, and Jaewoo Kang. Collabonet: collaboration of deep neural networks for biomedical named entity recognition. *BMC bioinformatics*, 20(10):249, 2019.

[112] Hui Yuan, Raymond YK Lau, and Wei Xu. The determinants of crowdfunding success: A semantic text analytics approach. *Decision Support Systems*, 91: 67–76, 2016.

[113] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Robert Fergus. Deconvolutional networks. In *The Conference on Computer Vision and Pattern Recognition*, volume 10, page 7, 2010.

[114] Hao Zhang, Gunhee Kim, and Eric P Xing. Dynamic topic modeling for monitoring market competition from online text and image data. In *The 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1425–1434. ACM, 2015.

[115] Shanshan Zhang, Lihong He, Slobodan Vucetic, and Eduard Dragut. Regular expression guided entity mention mining from noisy web data. In *The 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1991–2000, 2018.

[116] Shanshan Zhang, Lihong He, Eduard Dragut, and Slobodan Vucetic. How to invest my time: Lessons from human-in-the-loop entity extraction. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2305–2313, 2019.

[117] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, pages 649–657, 2015.

[118] Xiao Zhang, Yong Jiang, Hao Peng, Kewei Tu, and Dan Goldwasser. Semi-supervised structured prediction with neural crf autoencoder. In *The 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1701–1711, 2017.

[119] Yi Zhang, Tao Ge, Furu Wei, Ming Zhou, and Xu Sun. Sequence-to-sequence pre-training with data augmentation for sentence rewriting. *arXiv preprint arXiv:1909.06002*, 2019.

[120] Hongke Zhao, Hefu Zhang, Yong Ge, Qi Liu, Enhong Chen, Huayu Li, and Le Wu. Tracking the dynamics in crowdfunding. In *The 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 625–634. ACM, 2017.

[121] Haichao Zheng, Dahui Li, Jing Wu, and Yun Xu. The role of multidimensional social capital in crowdfunding: A comparative study in china and us. *Information & Management*, 51(4):488–496, 2014.

[122] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.