Word Embeddings with Applications to Web Search and Advertising

Necati Bora Edizel

DOCTORAL THESIS UPF / 2019

Directors of the thesis:

Prof. Dr. Ricardo Baeza-Yates, Departament of Information and Communication Technologies

Dr. Amin Mantrach, Criteo Labs

Anneme
To my mom

Acknowledgements

It was lovely... I'm so happy that I made this journey which changed my life. I learnt, enjoyed and loved a lot.

Tülay, you are my hero, you are my light, you are my teacher, you are my mom. I learnt my values from you. Resilience, craziness, living every moment with intensity and passion, and maybe the most important one: to love and to be loved. My lovely sister *Özlem*, you always take care of younger brother, you are always unconditionally helpful and a great example. You always gave me so much inspiration and vision.

After the most beautiful women, worth to mention some guys from family as well. *Dad*, we lost you when I was one year old. I don't know you but I feel that you were a cool guy. I feel that we are good friends and will meet one day. And my grandfather *Necati*, I learnt the figure of a man from you. Humble, gracious, positive, helpful and lovely. Your loss is the last one that I cried for. Your loss changed the notion of sorrow. Grandfather *Emcet*, we never got to close but we always had a special relationship. In these lines, it is worth to mention that you provided my university expenses. One day, they said we lost you and you left some money to me. I used it for higher education. Without you, I couldn't be here.

My academic journey started at Yahoo Labs in 2013 thanks to dear *Ricardo*. After the master, I started my extra-ordinary PhD journey with *Ricardo* and *Francesco*. There were many ups and downs but we managed to reach here. If you have funding, I would love to do another PhD with you guys :) It was beautiful. Moreover, I would like to thank *Amin* and *Fabrizio*. I learnt a lot from you. I enjoyed all discussions and brainstorming sessions. I hope that we will work all together in a near future.

I also need to mention my beautiful, lovely, amazing colleagues *Ioanna* and *Çiğdem*. I shared many interesting feelings with you; mainly about Yahoo Labs and our brilliant academic family. Talking, sharing, enjoying life was very beautiful with you. *Çiğdem*, you will be always very special for me. Your deep soul inspired me so much. I'm missing you. *Ioanita*, my Greek beauty, I guess we did it very well. It was lovely to sit in between

of you and *Francesco*. All those lovely moments....

And music... I shared many amazing memories with my musician friends during 5 years. It is hard to express verbally all those feelings. *Diego*, *Emilio*, *Nick*, *Ugniş*, *PierPa*, *GGM*, *Urita*, *Pans*, *Ali*, *Alan*, ... thank you very much people! I learnt a lot from you. *Diego Saez Sexy Trumper*, even though you offend me frequently, I always felt your support. You are my Joker. When I need help, I will knock your door. Thanks a lot! *Emilio*, we are wild, dirty and real. I’m missing you... And *Nick*, Barcelona was much more meaningful with you. Music was really more fun with you (also with *Mireia*).

Fırat, Batu, Miraç and *Olcay*... You guys never leave me alone. Whenever I fall, one of you was around. I’m so lucky to be your friend. You are so beautiful, real and horny. *Batu*, you are my sun, flatmate, best friend. We laughed everyday. *Fırat*, you are my brother. Our friendship started 20 years ago. When I think about life, I remember you as a fundamental part of it. *Miraç* you are a lover and definitely have *the* taste. You know what I mean. And lastly dear *Olcay*, we had really good moments together especially in İstanbul. Fatih is much more beautiful with you. There is a lot to say for each of you...

Most importantly, I felt in love several times. Argentina, Uruguay, todo muy guay. Thanks life! Thanks Barcelona! Hope to stay always in between *Sex, Drugs & Rock’n Roll* and *Peace, Wisdom & Unity*.

Abstract

Word embeddings are a building block of many practical applications across NLP and related disciplines. In this thesis, we present theoretical analysis and algorithms to learn word embeddings. Moreover, we present applications of word embeddings that concern Web Search and Advertising.

We start by presenting theoretical insights for one the most popular algorithm to learn word embeddings *word2vec*. We also model *word2vec* in Reinforcement Learning framework and showed that it's an off-policy learner with a fixed behavior policy. Then we present an off-policy learning algorithm $word2vec_\pi$ that uses *word2vec* as a behavior policy.

Then, we present a method to learn word embeddings that are resilient to misspellings. Existing word embeddings have limited applicability to malformed texts, which contain a non-negligible amount of out-of-vocabulary words. We propose a method combining FastText with subwords and a supervised task of learning misspelling patterns. In our method, misspellings of each word are embedded close to their correct variants.

Lastly, we propose two novel approaches (one working at the character level and the other working at word level) that use deep convolutional neural networks for a central task in NLP, semantic matching. We experimentally showed the effectiveness of our approach using click-through rate prediction task for Sponsored Search.

Özet

Günümüzde birçok Doğal Dil İşleme ve ilgili alanlarda, kelimeleri çok düzlemli uzayda temsil eden vektörler temel yapı taşı olarak kullanılmaktadırlar. Bu tezde, bu vektörleri öğrenebilen algoritmalar ve onların teorik analizini sunacağız. Bunun yanında, Web Search ve Web Reklamcılığında alanlarını gözeterek, bu vektörlerin çeşitli uygulamalarını sunacağız.

Öncelikle, bu alandaki en popüler olan algoritma olan $word2vec$'in teorik analizini sunacağız. Dahası, $word2vec$'i Reinforcement Learning ekosistemine taşıyacak, onun bir off-policy learning methodu olduğunu ve sabit bir behaviour policye sahip olduğunu göstereceğiz. Akabinde, $word2vec$'i behaviour policy olarak kullanan $word2vec_\pi$'ı sunacağız.

Var olan kelime vektörü üreten methodlar, yazım hatalı kelimeler için efektif sonuçlar üretememektedirler. Kullanıcaların Web'de ürettikleri birçok yazının yazım hatası içerdiğini göz önüne alırsak, bunun ne kadar önemli bir problem olduğu görülecektir. Bu nedenle, yazım hatalı kelimeleri tolere edebilecek bir kelime vektörü öğrenme metodu sunucağız. Bu metod, FastText metodunu temel alırken, aynı zamanda yazım hataları patternlerini öğrenmeye çalışmaktadır.

Son olarak, anlamsal eşleme problemini hedef alan, 2 önemli çözüm sunacağız. Bunlardan bir tanesi karakter seviyesinde, diğeri ise kelime seviyesinde çalışan derin sinir ağları olacak. Bu çözümlerin var olan diğer çözümlerden daha iyi sonuçlar verdiğini click-through rate tahmini problemini gözeterek, deneysel bir biçimde göstereceğiz.

Resum

Dins del món del Processament del Llenguatge Natural (NLP) i d'altres camps relacionats amb aquest àmbit, les representaciones latents de paraules (word embeddings) s'han convertit en una tecnologia fonamental per a desenvolupar aplicacions pràctiques. En aquesta tesi es presenta un anàlisi teòric d'aquests word embeddings així com alguns algoritmes per a entrenar-los. A més a més, com a aplicació pràctica d'aquesta recerca també es presenten aplicacions per a cerques a la web i màrqueting. Primer, s'introdueixen alguns aspectes teòrics d'un dels algoritmes més populars per a aprendre word embeddings, el word2vec. També es presenta el word2vec en un context de Reinforcement Learning demostrant que modela les normes no explícites (off-policy) en presència d'un conjunt de normes (policies) de comportament fixes. A continuació, presentem un nou algoritme de d'aprenentatge de normes no explícites (off-policy), $word2vec_\pi$, com a modelador de normes de comportament. La validació experimental corrobora la superioritat d'aquest nou algorithme respecte *word2vec*.

Segon, es presenta un mètode per a aprendre word embeddings que són resistents a errors d'escriptura. La majoria de word embeddings tenen una aplicació limitada quan s'enfronten a textos amb errors o paraules fora del vocabulari. Nosaltres proposem un mètode combinant FastText amb sub-paraules i una tasca supervisada per a aprendre patrons amb errors. Els resultats proven com les paraules mal escrites estan pròximes a les correctes quan les comparem dins de l'embedding. Finalment, aquesta tesi proposa dues tècniques noves (una a nivell de caràcter i l'altra a nivell de paraula) que empren xarxes neuronals (DNNs) per a la tasca de similaritat semàntica. Es demostra experimentalment que aquests mètodes són eficaços per a la predicció de l'eficàcia (click-through rate) dins del context de cerces patrocinades.

Resumen

Los *Word Embeddings* son piezas fundamentales de muchas aplicaciones prácticas de Procesamiento del Lenguaje Natural y disciplinas afines. En esta tesis presentamos un análisis teórico y algorítmos para aprender *Word Embeddings*. Adicionalmente, mostramos aplicaciones de los *Word Embeddings* en el campo de los motores de busqueda y publicidad en internet.

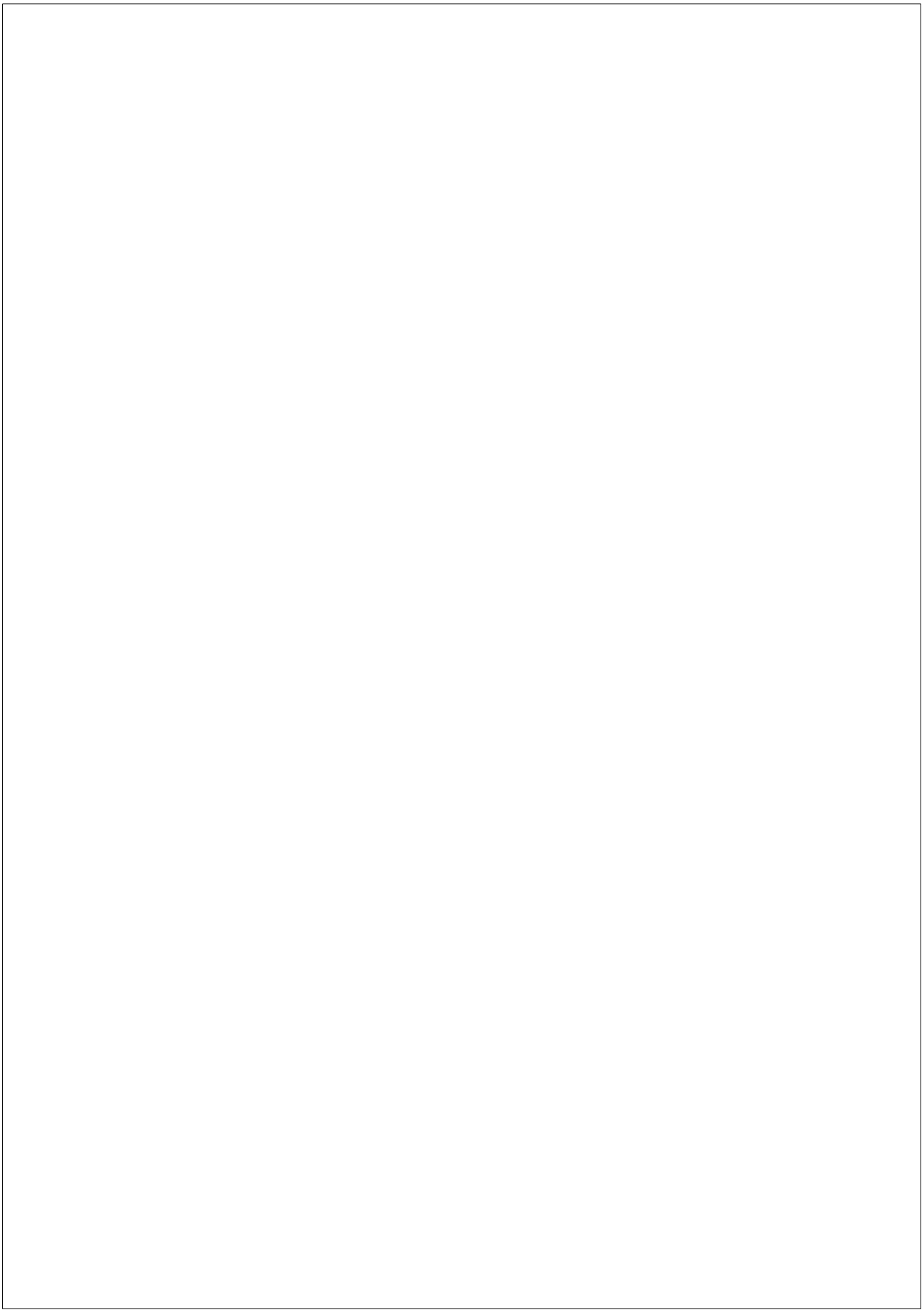
Comenzamos mostrando alguno de los aspectos teóricos de uno de los algorítmos más populares para aprender *Word Embeddings*: *word2vec*. También modelamos *word2vec* en un marco de Aprendizaje por Reforzamiento (*Reinforcement Learning*), mostrando.

Luego, presetanmos un metodo para aprender *Word Embeddings* que es resiliente a errores en la escritura. Los actuales *Word Embeddings* tienen limitaciones para su aplicación en textos malforados, que contienen un cantidad no menor the palabras fuera del vocabulario. Para lidiar con este problema, propoenes un método que combina *FastText* con sub-palabras (*subwords*) y una tarea de aprendizaje supervisado de patrones de errores en la escritúra. En nuestor método, los errores en cada palabra son embebidos cerca su versión correcta.

Finalmente, proponemos dos aproximaciones novedosas (una que trabaja a nivel de caracteres y otro a nivel de palabras) que usan redes neuronales profundas de convolución (*deep convolutional neural networks*) para un de las tareas centrales del procesamiento de lenguaje natural: las relaciones semánticas. Mostramos experimentalmente la efectividad de nuestra aproximación prediciendo el ratio de clicks (*click-through rate*) en el contexto de busquedas patrocinadas.

CONTENTS

LIST OF FIGURES



LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Motivation

Throughout the ages, knowledge has been mostly transferred by words. They are fundamental entities to understand and transfer knowledge. In order to process/understand/organize/summarize knowledge in huge databases such as the World Wide Web, we need computers. And computers need compact and powerful representations for words.

Before dense word embeddings, words were represented using sparse models like bag-of-words and n-grams. Although they were successful for many tasks, they suffer from the curse of dimensionality and scalability issues [6]. Observing that, researchers proposed dense, distributed representations of words [33, 14, 6].

Recently, an algorithm to learn neural distributed word embeddings *word2vec* [52, 53] has gained lots of attention both, from industry and the research community. *word2vec* has been used in several domains such as natural language processing [43], information retrieval [27], biology [2] and social networks [55].

There are different problems to study in the word embeddings research. Despite its popularity, theoretical background of *word2vec* algorithms are not well studied. Beyond loss function and optimization method, there

is not so much known about it, *i.e*, the sampling mechanism of context words.

Moreover, another important issue of word embeddings is that they are often not able to deal with malformed words, *i.e.* misspellings. Based on research [13], it is shown that human generated data has significant amount of malformed. On the other hand, *word2vec* cannot provide embeddings for words that have not been observed at training time such as misspelled words. Although, FastText [8], a subword variant of *word2vec* can generate representations for misspellings, it does not provide a satisfactory result.

One important application of word embeddings is semantic matching which is one of the central tasks in web search and advertising. The research community proposed models that use word level embeddings or models that require a lot of engineering efforts to define, compute, and select the appropriate features [35, 64, 70]. Hence, it is very appealing to apply deep character level models since they won't suffer from malformed text and also they are able to exploit a richer character level representation.

1.2 Goals and Contributions

The purpose of this thesis is to understand, improve and apply word embeddings at word, sub-word and character level. Our main contributions are:

- **Learning Word Vectors with Non-Fixed Policy** (Chapter 3). *word2vec* algorithm iterates over text word by word. For each word, it samples other words around it as context words based on a fixed probability distribution. We derive a closed-form formulation of the context words conditional probability distribution and show experimentally that it improves over a uniform distribution.

  Then, we give new insights about *word2vec* algorithm, by describing it as off-policy reinforcement learning algorithm with fixed behaviour policy. Moreover, We introduce an off-policy learning mechanism that uses *word2vec* as behavior policy and show on

state-of-the-art tasks, and languages used in the literature that the off-policy embeddings outperform the *word2vec* embeddings.

- **Misspelling Oblivious Word Embeddings** (Chapter 4). In this part of the thesis, we present a novel problem and a non-trivial solution to build word embeddings resistant to misspellings. Moreover, a novel evaluation method specifically suitable for evaluating the effectiveness of Misspelling Oblivious Word Embeddings **MOE** is presented. Lastly, a dataset on which such embeddings can be evaluated is released for the research community.

  Our work on misspelling oblivious word embeddings is accepted at North American Conference of the Association for Computational Linguistics in 2019, under the title "Misspelling Oblivious Word Embeddings" [23].

- **Character Level Embeddings and its Applications to Web Advertising** (Chapter 5). In this part of the thesis, we present a deep neural model to learn textual relationships. To the best of our knowledge, we are first to learn meaningful textual similarity between two pieces of text (*i.e*, query and ad) from scratch, *i.e*, at character level. Moreover, we are first to directly predict the click-through rate (CTR) in the context of sponsored search with little feature engineering (*i.e.*, page position as the only feature in addition to text).

  Our work on character level embeddings and its applications to web advertising was published in Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval in 2016 [22].

The semantic structure of the thesis and its contributions is summarized in Figure 1.1.

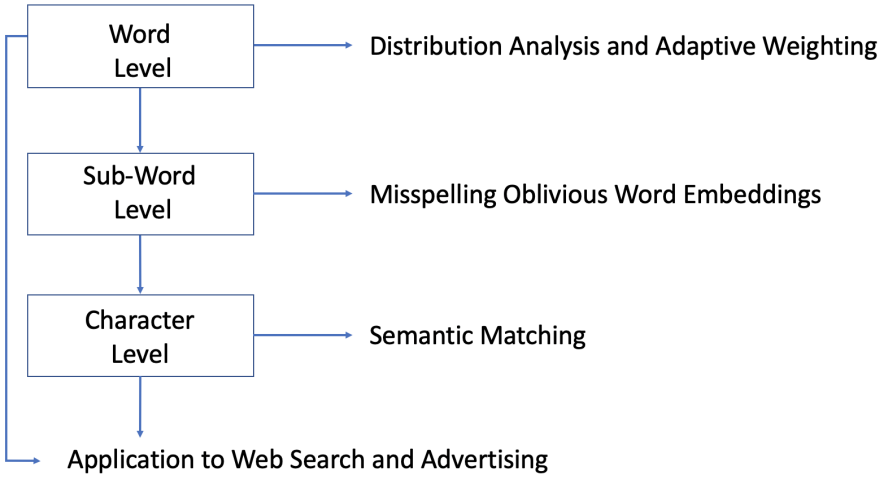


Figure 1.1: Diagram that illustrates the flow in the thesis.

1.3 Organization

This thesis is organized as follows. In Chapter 2, we present the state-of-the-art. In Chapter 3, we demystify sampling mechanism of *word2vec* algorithm. Then, we show that *word2vec* is an off-policy reinforcement learning method with fixed behavior policy, and introduce and off-policy with *word2vec* as behavior policy. In Chapter 4, we move from word level to sub-word level and study the problem of generating embeddings for misspelled words. In Chapter 5, we study the problem of learning textual relationships between 2 pieces of text where we compare word level models with character level models using a CTR prediction task. Lastly, in Chapter 6, we present conclusions of this thesis. Moreover, we discuss some possible future directions for word embeddings research.

CHAPTER 2

STATE OF THE ART

In this chapter, we present the state-of-the-art related to word embeddings. They are organized under three sections; (1) Word, (2) Sub-Word and (3) Character Level Word Embeddings.

2.1 Word Level Word Embeddings

One of the first works to introduce the concept of distributed representation for symbolic data was [33]. Later on, the Information Retrieval (IR) community proposed techniques of embedding words into a vector space. Latent Semantic Indexing (LSI) [14] was one of the most influential works in this area.

The first neural language model which jointly learns word embeddings was [6]. Although such a language model was outperforming the baselines, it was not practical because of long training time requirements. Collobert et al. [11] proposed new neural architectures for word embeddings and showed that pre-trained word embeddings can be very valuable for some downstream NLP tasks. Later on, when *word2vec* [53, 52] became very popular, both, because of its effectiveness and its ability to train a model on a very large text corpus efficiently, Levy et al. [47] showed that *word2vec*'s skip-gram with negative sampling model (SGNS) is implicitly equivalent

to word co-occurrence matrix factorization. Besides neural approaches, Pennington et al. [54] proposed an SVD based architecture which gained a lot of attention because it allows to effectively consider the popularity of each word in the model definition. Next, we would like to dive into the details of [53, 52]. *word2vec* embeddings can be learned using two different models: *skip-gram* and *Continuous Bag of Words (CBOW)*. Formally, let V be a vocabulary of words, and let $T = w_1, w_2, \ldots, w_n$ be a text represented as a sequence of words from V; given a word w_i in the text, we define the context of length l as $C_i = \{w_{i-l}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+l}\}$. In the skip-gram model the task is to predict each word of context C_i given a word w_i, *i.e.*, $\mathcal{P}(c_i|w_i;\theta)$, and the overall objective of the optimization problem associated with the task is that of maximizing $\sum_{i=1}^{n} \sum_{w_c \in C_i} \log \mathcal{P}(w_c|w_i;\theta)$. In the CBOW model the task, instead, is to predict a word w_i given its context C_i, *i.e.*, $\mathcal{P}(w_i|\, C_i;\theta)$, and the overall objective of the optimization problem is analogous to that of skip-gram. The probability measure $\mathcal{P}$ is usually parametrized as a softmax on each word w_c of the context C_i,

$$\mathcal{P}(w_c|w_i;\theta) = \frac{1}{\mathcal{Z}_i} e^{s(w_i,w_c)} \tag{2.1}$$

where $s(w_i, w_c)$ is a scoring function measuring how "similar" the words are and $\mathcal{Z}_i = \sum_{j \in V} e^{s(w_i,w_j)}$ is the normalization term. Finally, θ is the set of parameters of the model corresponding to the union of the set of input embedding vectors $\mathbf{v}$, and the set of output embedding vectors $\mathbf{u}$. We parametrize the scoring function s with the dot product $\mathbf{u}_c^T \mathbf{v}_i$, where $\mathbf{u}_c$ is an output vector associated with the word w_c and $\mathbf{v}_i$ is an input vector associated with the word w_i. Therefore, $s(w_i, w_c) = \mathbf{u}_c^T \mathbf{v}_i$.

As it is well known, computing the normalization term $\mathcal{Z}_i$ is computationally expensive. Several methods have been proposed to avoid computing it directly. The approach adopted in *word2vec* is known as *Negative Sampling*. Negative sampling replaces the original multi-class classification task with binary classification where the model uses k negative samples for each positive training pair (w_c, w_i). We encourage the reader to consult [20] for more details about Negative Sampling. The

skip-gram with negative sampling is therefore defined as follows:

$$L_{W2V} := \sum_{i=1}^{n} \sum_{w_c \in C_i} [\ell(s(w_i, w_c)) + \sum_{w_n \in N_{i,c}} \ell(-s(w_i, w_n))] \tag{2.2}$$

where ℓ denotes the logistic loss function $\ell(x) = \log(1 + e^{-x})$, $w_n \in N_{i,c}$ represents negative samples, number of negative samples for each positive example that is $k = |N_{i,c}|$.

2.2 Sub-Word Level Word Embeddings

Besides word level embedding models like *word2vec*, sub-word level embedding models have become popular, such as FastText [8]. Indeed, the major innovation of FastText is the introduction of subword level features to the *word2vec* framework. It uses the same loss function L_{W2V} as *word2vec* but it extends the way words are represented. In *word2vec*'s skip-gram model, a word w_i is represented by a single input vector $\mathbf{v}_i$. In FastText we additionally embed subwords of a word and make use of the subwords representations to represent w_i. We will refer to subwords as character n-grams. Formally, given an integer n with $m \leq n \leq M$, where M (resp. m) is the maximum (resp. minimum) length of an n-gram, the FastText model embeds all possible character n-grams of the word. For example, if $m = 3$, $M = 5$ and the word is *banana*, the set of n-grams is "ban, ana, nan, bana, anan, nana, banan, anana". Let $\mathcal{G}_{w_i}$ denote the set of all subwords of a word w_i plus the word itself (*e.g.* for the word banana $\mathcal{G}_{\text{banana}}$ is the set defined in the example above plus the word "banana" itself). Given $\mathcal{G}_{w_i}$, FastText's scoring function for word w_i and context w_c is defined as follows:

$$s(w_i, w_c) = \sum_{\mathbf{v}_g, g \in \mathcal{G}_{w_i}} {\mathbf{v}_g}^T \mathbf{u}_c \tag{2.3}$$

Therefore, the representation of w_i is simply the sum of the representations of each of the n-grams derived from w_i plus the representation of w_i itself. As like *word2vec*, FastText also uses a Negative Sampling technique. With extensive experiments, FastText showed clear improvements over the original *word2vec* skip-gram model [8]. We present a loss function of FastText L_{FT} as follows:

$$L_{FT} := \sum_{i=1}^{n} \sum_{w_c \in C_i} [\ell(\sum_{\mathbf{v}_g, g \in \mathcal{G}_{w_i}} {\mathbf{v}_g}^T \mathbf{u}_c) + \sum_{w_n \in N_{i,c}} \ell(-\sum_{\mathbf{v}_g, g \in \mathcal{G}_{w_i}} {\mathbf{v}_g}^T \mathbf{u}_n)] \quad (2.4)$$

An alternative to FastText is MIMICK [57]. MIMICK's goal is that of representing pre-trained word embeddings by means of character-based embeddings that learn to minimize the distance between embeddings produced by a char-based approach and the pre-trained embeddings. The rationale is that MIMICK is a generalization of FastText that should work also on out of vocab words.

2.3 Character Level Word Embeddings

There are a number of works learning at character level for different natural language processing (NLP) tasks in recent years. Nogueira dos Santos *et al.* [18] are among the first to use character-level information for part-of-speech tagging. They propose to jointly use character-level representation and the more traditional word embedding in a deep neural network for this. Later on, they propose to use a similar deep neural network with character-level and word-level representations to perform name entity recognition [61].

Several following works [5, 12, 39, 71] demonstrate the power of character-level information alone in NLP tasks. Ballesteros *et al.* [5] discuss the benefits of replacing word-level representation by character-level representation in long short-term memory (LSTM) recurrent neural

networks to improve transition-based parsing. Kim *et al.* [39] show in their work that character inputs are sufficient for modeling most of the languages, and their LSTM recurrent neural network language model processing character inputs are as good as the state-of-the-art models using word-level or morpheme-level inputs for English. Zhang *et al.* [71] explore the use of character-level convolutional networks for text classification and show that character-level convolutional networks achieve competitive results against traditional models and deep models such as word-based ConvNets [44]. Conneau *et al.* [12] further show that when using very deep networks of up to 29 convolutional layers, a model that operates directly at character level achieves significant improvements over the state-of-the-art on several public text classification tasks. Interestingly, in the case of big datasets, they report good results using shallower neural networks. Bojanowski *et al.* [8] extends the skip-gram model by learning representations for character n-grams. Words are then represented as a bag of character n-grams. The model shows state-of-the-art performance on word similarity and analogy tasks, especially for morphologically rich languages.

Chapters 3 (Learning Word Vectors with an Adaptive Policy), 4 (Misspelling Oblivious Word Embeddings) and 5 (Character Level Embedding and its Application to Web Advertising) have been removed at the author's request

CHAPTER 6

CONCLUSIONS

6.1 Summary

Word embeddings and its applications are becoming more popular and fundamental components of many real world applications. In this thesis, we attempt to understand, improve and apply word embeddings.

Even though there are many works on the topic of word embeddings, there are still many knowledge gaps. Even though famous approach *word2vec* is widely used, its sampling mechanism was not clear from theoretical point of view. In Chapter 3, we present an analysis of sampling mechanism of famous approach *word2vec*. We experimentally demonstrate that context words conditional probability distribution improve over uniform distribution. Later on, we approach to sampling mechanism from a different angle. We formulate *word2vec* algorithm, by describing it as off-policy reinforcement learning algorithm where behavior policy is fixed. Also, We develop an off-policy learner where behavior policy uses $word2vec$ policy. On state-of-the-art tasks and languages, we show that proposed off-policy embeddings outperform the *word2vec* embeddings. In this work, we consider *words* as an atomic unit.

In Chapter 4, we present a novel problem, generating embeddings for malformed text *i.e.* misspellings. While working with misspellings,

using *words* as atomic unit is not practical. That's why, we move into *sub-word* atomic level instead of *word*. We extend the original FastText loss function by adding a supervised loss in order to learn misspelled words. Experimental results show that proposed supervised loss is successfully mapping misspellings to its corrected versions. Moreover, a novel evaluation method suitable for evaluating the embeddings of misspelled words is presented. For the sake of reproducibility of study, we release a dataset collected from a social network. We hope that, released dataset will increase the number works about misspellings which is a clear problem for real life applications.

In Chapter 5, we work on a very central task in Natural Language Processing: Semantic Matching. In this chapter, we use *characters* as atomic units and present we present a deep neural model to learn textual relationships. To the best of our knowledge, we are first to learn meaningful textual similarity between two pieces of text (*i.e.*, query and ad) from scratch, *i.e.*, at character level. Moreover, we are first to directly predict the click-through rate in the context of sponsored search with little feature engineering.

6.2 Future Directions

In Chapter 3, we propose an off-policy learner to learn word embeddings. Since we formulate the problem in Reinforcement Learning setting, there can be many different approaches to check such as introducing a meaningful reward function. Another possible extension can be learning a value function. From word embeddings point of view, we can apply our off-policy learner for FastText.

Misspelling Oblivious Embeddings presented in Chapter 4 is the first work to deal with embedding of words that are resistant to misspellings. For this reason, there is plenty of open problems to address. First, we are also planning to test different ways of training embeddings for misspellings including the extension of the same technique to multi-lingual embeddings. Moreover, We are going to test deep architectures to combine the n-grams

in misspellings to better capture various interdependencies of n-grams and correct versions of words. Finally, we will assess the robustness of both character-based [40] and context-dependent embeddings [16], [56] with respect to misspellings.

In Chapter 5, a character level deep neural network to predict CTR of a query-ad pair is predicted. The Proposed model can be applied to different NLP problems where matching two pieces of text is needed. Moreover, it would be interesting to make runtime analysis of proposed models to see whether it can be used in a demanding, time-critical industrial settings.

BIBLIOGRAPHY

[1] L. M. Aiello, I. Arapakis, R. A. Baeza-Yates, X. Bai, N. Barbieri, A. Mantrach, and F. Silvestri. The role of relevance in sponsored search. In *Proceedings of the 25th ACM CIKM*, pages 185–194, 2016.

[2] E. Asgari and M. R. Mofrad. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PloS one*, 10(11):e0141287, 2015.

[3] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Pearson Addison Wesley, Harlow, England, 2 edition, 2011.

[4] X. Bai, E. Ordentlich, Y. Zhang, A. Feng, A. Ratnaparkhi, R. Somvanshi, and A. Tjahjadi. Scalable query n-gram embedding for improving matching and relevance in sponsored search. In *Proceedings of the 24th SIGKDD*, pages 52–61, 2018.

[5] M. Ballesteros, C. Dyer, and N. A. Smith. Improved transition-based parsing by modeling characters instead of words with lstms. In *Proceedings of EMNLP*, pages 349–359, 2015.

[6] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[7] G. Berardi, A. Esuli, and D. Marcheggiani. Word embeddings go to italy: A comparison of models and training datasets. In *IIR*, 2015.

[8] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[9] Y. Chen and T. W. Yan. Position-normalized click prediction in search advertising. In *Proceedings of the 18th ACM SIGKDD*, pages 795–803, 2012.

[10] H. Cheng and E. Cantú-Paz. Personalized click prediction in sponsored search. In *Proceedings of the 3rd ACM WSDM*, pages 351–360. ACM, 2010.

[11] R. Collobert and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[12] I. Conneau, H. Schwenk, L. Barrault, and Y. LeCun. Very deep convolutional networks for natural language processing. *CoRR*, abs/1606.01781, 2016.

[13] S. Cucerzan and E. Brill. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Proceedings of EMNLP 2004*, pages 293–300, 2004.

[14] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[15] T. Degris, M. White, and R. S. Sutton. Off-policy actor-critic. *CoRR*, abs/1205.4839, 2012.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[17] P. Domingos. A unified bias-variance decomposition. In *Proceedings of 17th International Conference on Machine Learning*, pages 231–238, 2000.

[18] C. N. dos Santos and B. Zadrozny. Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st ICML*, pages 1818–1826, 2014.

[19] S. Dreiseitl and L. Ohno-Machado. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, 35(5):352–359, 2002.

[20] C. Dyer. Notes on noise contrastive estimation and negative sampling. *arXiv preprint arXiv:1410.8251*, 2014.

[21] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. Technical Report 11765, National Bureau of Economic Research, November 2005.

[22] B. Edizel, A. Mantrach, and X. Bai. Deep character-level click-through rate prediction for sponsored search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 305–314, New York, NY, USA, 2017. ACM.

[23] B. Edizel, A. Piktus, P. Bojanowski, R. Ferreira, E. Grave, and F. Silvestri. Misspelling oblivious word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

[24] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited.

In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM, 2001.

[25] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *Proceedings of the 14th AISTAT*, volume 15, pages 315–323, 2011.

[26] T. Graepel, J. Q. Candela, T. Borchert, and R. Herbrich. Web-scale bayesian click-through rate prediction for sponsored search advertising in Microsoft's Bing search engine. In *Proceedings of the 27th ICML*, pages 13–20, 2010.

[27] M. Grbovic, N. Djuric, V. Radosavljevic, F. Silvestri, R. Baeza-Yates, A. Feng, E. Ordentlich, L. Yang, and G. Owens. Scalable semantic matching of queries to ads in sponsored search advertising. In *Proceedings of the 39th ACM SIGIR*, pages 375–384, 2016.

[28] I. Gurevych. Using the structure of a conceptual network in computing semantic relatedness. In *Proceedings of the Second International Joint Conference on Natural Language Processing*, IJCNLP'05, pages 767–778, Berlin, Heidelberg, 2005. Springer-Verlag.

[29] S. Hassan and R. Mihalcea. Cross-lingual semantic relatedness using encyclopedic knowledge. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, pages 1192–1201, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[30] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of IEEE ICCV*, 2015.

[31] X. He, J. Pan, O. Jin, T. Xu, B. Liu, T. Xu, Y. Shi, A. Atallah, R. Herbrich, S. Bowers, et al. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the 8th International Workshop on Data Mining for Online Advertising*, pages 1–9, 2014.

[32] D. Hillard, E. Manavoglu, H. Raghavan, C. Leggetter, E. Cantú-Paz, and R. Iyer. The sum of its parts: Reducing sparsity in click estimation with query segments. *Information Retrieval*, 14(3):315–336, June 2011.

[33] G. E. Hinton. Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, volume 1, page 12. Amherst, MA, 1986.

[34] B. Hu, Z. Lu, H. Li, and Q. Chen. Convolutional neural network architectures for matching natural language sentences. In *Proceedings of the 27th NIPS*, pages 2042–2050, 2014.

[35] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM CIKM*, pages 2333–2338, 2013.

[36] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd ICML*, pages 448–456, 2015.

[37] Z. Jiang, S. Gao, and W. Dai. Research on ctr prediction for contextual advertising based on deep architecture model. *Journal of Control Engineering and Applied Informatics*, 18(1):11–19, 2016.

[38] Y. Juan, Y. Zhuang, W.-S. Chin, and C.-J. Lin. Field-aware factorization machines for ctr prediction. In *Proceedings of the 10th ACM RecSys*, pages 43–50, 2016.

[39] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush. Character-aware neural language models. In *Proceedings of the 13th AAAI*, pages 2741–2749, 2016.

[40] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush. Character-aware neural language models. In *AAAI*, pages 2741–2749, 2016.

[41] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd ICLR*, 2014.

[42] M. Köper, C. Scheible, and S. Schulte im Walde. Multilingual reliability and "semantic" structure of continuous word spaces. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 40–45. Association for Computational Linguistics, 2015.

[43] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196, 2014.

[44] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[45] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710, 1966.

[46] I. Leviant and R. Reichart. Judgment language matters: Multilingual vector space models for judgment language aware lexical semantics. *CoRR*, abs/1508.00106, 2015.

[47] O. Levy and Y. Goldberg. Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185, 2014.

[48] T. Luong, R. Socher, and C. Manning. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113, 2013.

[49] X. Ma and E. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*, 2016.

[50] H. B. McMahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, D. Golovin, et al. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD*, pages 1222–1230, 2013.

[51] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized on-line matching. In *Proceedings of the 46th IEEE FOCS*, pages 264–273, 2005.

[52] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[53] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[54] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, pages 1532–1543, 2014.

[55] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[56] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proc. of NAACL*, 2018.

[57] Y. Pinter, R. Guthrie, and J. Eisenstein. Mimicking word embeddings using subword rnns. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 102–112, 2017.

[58] T. Qin and T. Liu. Introducing LETOR 4.0 datasets. *CoRR*, abs/1306.2597, 2013.

[59] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th WWW*, pages 521–530, 2007.

[60] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[61] C. N. d. Santos and V. Guimaraes. Boosting named entity recognition with neural character embeddings. *CoRR*, abs/1505.05008, 2015.

[62] M. Saveski and A. Mantrach. Item cold-start recommendations: learning local collective embeddings. In *Proceedings of the 8th ACM RecSys*, pages 89–96, 2014.

[63] B. Shaparenko, O. Çetin, and R. Iyer. Data-driven text features for sponsored search click prediction. In *Proceedings of the 3rd International Workshop on Data Mining and Audience Intelligence for Advertising*, pages 46–54, 2009.

[64] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM CIKM*, pages 101–110, 2014.

[65] C. Spearman. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101, 1904.

[66] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[67] E. M. Voorhees et al. The trec-8 question answering track report. In *Trec*, volume 99, pages 77–82, 1999.

[68] T. Wang, J. Bian, S. Liu, Y. Zhang, and T.-Y. Liu. Psychological advertising: Exploring user psychology for click prediction in sponsored search. In *Proceedings of the 19th ACM SIGKDD*, pages 563–571, 2013.

[69] T. Zesch and I. Gurevych. Automatically creating datasets for measures of semantic relatedness. In *Proceedings of the Workshop on Linguistic Distances*, pages 16–24. Association for Computational Linguistics, 2006.

[70] S. Zhai, K. Chang, R. Zhang, and Z. M. Zhang. Deepintent: Learning attentions for online advertising with recurrent neural networks. In *Proceedings of the 22nd ACM SIGKDD*, pages 1295–1304, 2016.

[71] X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. In *Proceedings of the 28th NIPS*, pages 649–657. 2015.

[72] Y. Zhang, H. Dai, C. Xu, J. Feng, T. Wang, J. Bian, B. Wang, and T.-Y. Liu. Sequential click prediction for sponsored search with recurrent neural networks. In *Proceedings of the 11th AAAI Conference*, pages 1369–1375, 2016.