2016 2nd International Conference on Next Generation Computing Technologies (NGCT-2016)
Dehradun, India 14-16 October 2016

Clickbait Detection using Deep Learning

Amol Agrawal
Citrix R&D India Pvt. Ltd.
Bangalore, India 560042
Email: pfrcks@gmail.com

***Abstract*—Clickbaits, in social media, are exaggerated headlines whose main motive is to mislead the reader to "click" on them. They create a nuisance in the online experience by creating a lure towards poor content. Online content creators are utilizing more of them to get increased page views and thereby more ad revenue without providing the backing content. This paper proposes a model for detection of clickbait by utilizing convolutional neural networks and presents a compiled clickbait corpus. We create a corpus using multiple social media platforms and utilize deep learning for learning features rather than undergoing the long and complex process of feature engineering. Our model achieves high performance in identification of clickbaits.**

***Index Terms*—Clickbait, convolutional neural networks, deep learning.**

I. Introduction

A popular trend in the online content today is the prevalence of clickbait, which is nothing but online content of misleading nature, with the sole aim of attracting the viewers' attention and luring them to their web page. Clickbaits are characterized by poor content with little value and the agencies deploying them are heavily dependent on revenue by ad streams. Therefore, they create eye catching titles which lure users into clicking them, thereby generating revenue. Often promising a worthwhile experience or an indispensable revelation, these articles feed on human psychology and create a frustrating experience for the user, as he or she does not usually get the quality of content they were expecting.

Fig. 1 shows some examples of clickbait that might appear during an average online media surfing session. As is visible, the titles seem to be promising some very unusual or expository content; however upon clicking and visiting the page we get only little valuable content. The amount of clickbait in social media has risen significantly in the last few years and has now reached such levels that some news publishers are also utilizing these techniques.

Even though research in the field of clickbait detection is still in an early phase, a lot of attention has come towards clickbait. Because of the increasing pervasiveness of clickbait in online media and news, significant backlash has started to happen against social media platforms where such content appears. Facebook decided to take action against clickbait as told by El-Arini and Tang [1], however it still continues to be flooded with clickbait articles. To combat this, a number of Twitter handles have sprung up and gained huge followings, whose sole purpose is to identify clickbait. Handles like @SavedYouAClick[1] and @HuffPoSpoilers[2] are consistently updating their feeds with clickbait posts to create awareness about them. However the method of their detection is manual; the users running those twitter accounts themselves read and classify the tweet as clickbait or not for the benefit of others.

The reasoning behind why clickbait has become so widespread can be derived from a number of sources. Reis et al. [2] studied around 69, 000 headlines from four international media houses in 2014. They analyzed the polarity of sentiments for these headlines and found extremities in sentiments resulted in increased popularity. Headlines provide the first impression and can affect how the news articles are perceived by users, as found out by Digirolamo and Hintzman [3]. By drawing attention to certain details or facts, a headline can affect which existing knowledge is activated in one's brain. By its choice of phrasing, a headline can influence one's mindset so that readers later recall details that coincide with what they were expecting, leading individuals to perceive the same content differently according to the headline as shown by Dooling and Lachman [4]. Another prominent explanation is the frequently cited Loewenstein' information gap theory [5]. In simple words, the theory holds that whenever we perceive a gap between what we know and what we want to know, that gap has emotional consequences. Such information gaps produce the feeling of deprivation labeled curiosity, he wrote. The curious individual is motivated to obtain the missing information to reduce or eliminate the feeling of deprivation. In other words, not knowing makes us uncomfortable [6].

Some research that has been done on this topic employs hand-created features and supervised learning. However, our approach focuses on deep learning for the detection. Deep Learning for Natural Language Processing(NLP) has received attention in the recent times. Although initially designed for research on Computer Vision [7] and Speech Recognition [8], deep learning has now been proved to be also quite successful for sentence classification purposes. Convolutional Neural Networks (CNN), a form of deep learning technique, has been utilized to achieve good and even state of the art on sentence classification as published by Kim [9].

In this present work, we obtain data by parsing various social platforms to get a corpus of clickbait and non-clickbait headlines. We then train a CNN on the dataset. Without going through the non-trivial process of feature creation, we

[1]https://twitter.com/SavedYouAClick
[2]https://twitter.com/HuffPoSpoilers

978-1-5090-3257-0/16/$31.00 ©2016 IEEE

This Cat Was Trying To Get The Attention Of Locals, But What Someone Finds With Her? Unbelievable

Little Girl Tries To Wake Her Brother Up. What He Did Was Truly Unbelievable!

Wait! Don't Put Butter on That Grilled Cheese Sandwich. Do THIS Instead! (You Can Thank Us Later.)

Fig. 1: Some Examples of clickbait

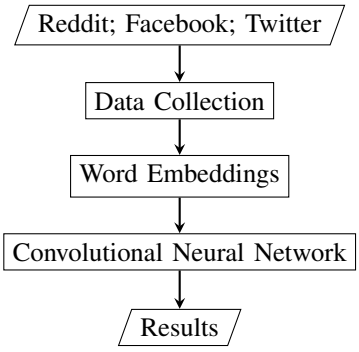


Fig. 2: The Proposed Model

achieve a high measure of accuracy with this model, even without doing any significant hyperparameters tuning. Our contributions are twofold: (1) We make publicly available a clickbait corpus that has been taken from different social media sources. Currently no such corpus is available, and (2) we utilize and evaluate the first deep learning model for clickbait detection which achieves a high accuracy along with precision and recall. By utilizing a corpus derived from different social media sources, our model is able to learn generalized features and not features that are platform-specific. We also contribute by strengthening the support towards the evidence that pre-training of word vectors using unsupervised learning makes an important addition to deep learning methodologies for NLP .

We discuss related work in Section II, our data collection and deep learning model in Section III, and in Section IV details of the results obtained.

II. Related Work

Clickbait has been a subject of research twice by linguists and twice by computer science research teams. Amongst linguists, Bram Vijgen [10] studied ”listicles” which are articles containing a list of things. Listicles are one of the major types of clickbaits. The titles like ”16 Cancer Causing Foods You Probably Eat Every Day” or ”38 Celebrities You Didnt Know Passed” are some examples which are taken from our own compiled dataset. The authors studied around 700 listicles by Buzzfeed. They found that the titles share a very homogeneous structure: 85% of them starting with a cardinal number– the number of items in the list–while all articles contain the number in some place or the other. Second was Blom and Hansen [11], who studied usage of forward reference in 2000 random headlines from a Danish news website. Forward reference is utilized to create a information gap by giving only a teasing headline and luring the user to click the title. Some examples include ”This shocking news will blow your mind” or ”What He did next shocked everyone.” In the aforementioned examples ’This’ and ’He’ are the forward references to some entities which are not disclosed, enticing the user to click to find out. They found that these forward references are mostly made up of definite articles, adverbs and personal and demonstrative pronouns.

Amongst computer scientists, Potthast, Kopsel, Stein and Hagen [12] collected around 3000 tweets from the top 20 publishers on Twitter. They created a model based on handcrafted features from three fields, the teaser message or title, the linked web page and the meta information. The teaser message had some basic text and dictionary features. The linked web page features included text and readability features, while the meta information had features related to the tweets themselves. These features were fed into a supervised classification mechanism which achieved 0.79 ROC-AUC at 0.76 precision and 0.76 recall. They found that features from category one alone outperformed all other categories, with character n-gram and word 1-gram features contributing the most as they are known to capture writing styles.

Biyani, Tsioutsiouliklis and Blackmer [13] defined eight types of clickbait. They used these definitions to gather 1349 clickbait and 2724 non-clickbait webpages from the Yahoo homepage. They handcrafted features in three major categories. Content-based features are ones which are derived from the text of the title and body of the pages. Presence of quotes, exclamations, questions, etc. were taken as features along with traditional features such as unigrams and bigrams. Similarity-based features calculated similarity between the title and first five lines of the body of the webpage individually. Informality features calculated formality and quality of the pages while Forward Reference features utilized features created after the four types of forward references given by Blom and Hansen [11]. For the clickbait class they achieved precision and recall of 0.712 and 0.548 respectively. They found that informality features were amongst the most important features followed by content features.

In contrast, our method gives up the time consuming and painstaking task of hand-created features. Instead we utilize deep learning to find features. We get very good results utilizing this technique, which might be because deep learning learns features on its own and it might be coming up with new and unthought of features. Also, the dataset collected by us comes from various sources which helps us in developing a more generalized model that is not constrained by the type of social media platform.

III. Proposed Model

The proposed model can be divided into three sections as shown in Fig. 2. First, we create the data corpus by collecting clickbait and non-clickbait headlines. Then these textual headlines are converted into word embeddings which

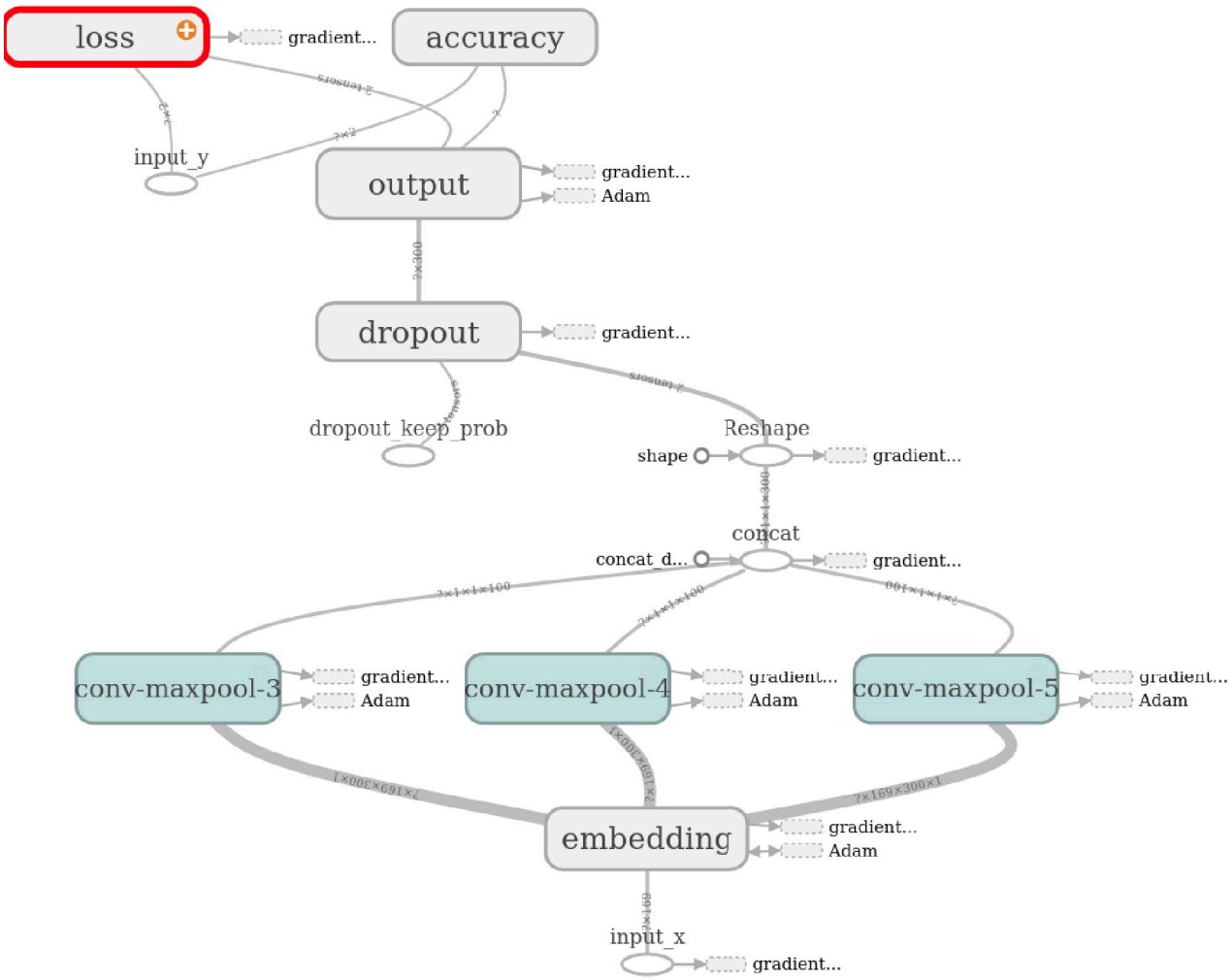


Fig. 3: The CNN Model Used

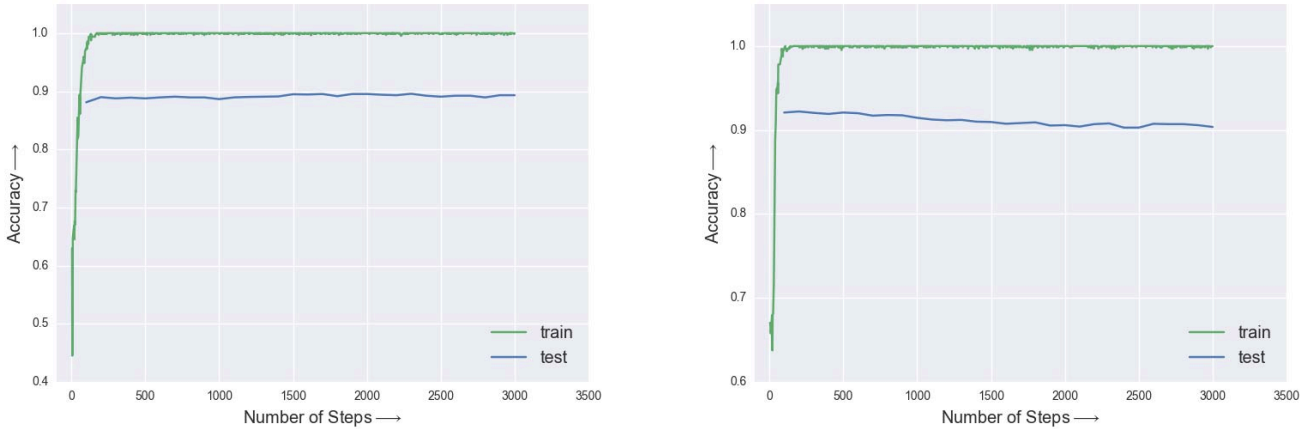
finally serves as input to our deep learning models, which in our case is a CNN.

A. Data Collection

Since there is an unavailability of any corpus related to clickbait, we create a corpus ourselves. Unlike [12] and [13] who only utilized data from a single source, we collected data from three sources, viz., Reddit, Facebook and Twitter, all three of them being popular social media platforms. We utilized this approach to ensure that the features learnt by our deep learning model are not social media platform-dependent. Each social media platform has it's own limitation, eg. Twitter allows a maximum of 140 characters per tweet. Hence we used multiple sources of data to train our deep learning model for clickbait detection.

To have good samples in both clickbait and non-clickbait categories, we utilized subreddits, pages and twitter handles who have a history of publishing clickbait or non-clickbait headlines. For collecting non-clickbait headlines, we utilized the Reddit subreddits /r/news[3] and /r/worldnews[4]. These subreddits are heavily moderated and do not allow any kind of clickbait, spam or ad to creep in. For collecting clickbait headlines, we utilized the Reddit subreddit /r/SavedYouAClick[5] which posts only clickbait headlines with the aim to educate people and make them avoid such links. We also used Twitter handle @HuffPoSpoilers and the Facebook page StopClickbait[6], both of which have the same motive as /r/SavedYouAClick. However, to maintain correctness of the data, data collected in both classes was assessed by three independent assessors. For both classes of headlines, we

[3]https://www.reddit.com/r/news
[4]https://www.reddit.com/r/worldnews
[5]https://www.reddit.com/r/SavedYouAClick
[6]https://www.facebook.com/StopClickBaitOfficial



(a) Click-Scratch (b) Click-Word2vec

Fig. 4: Accuracy vs Number of Steps for both models

achieved an "almost perfect" inter-assessor agreement, with the clickbait headlines having a Fleiss' $\varkappa$ of 0.85 while the non-clickbait headlines having a Fleiss' $\varkappa$ score of 0.83. The reason for such a high value of inter-assessor agreement is due to the fact that these headlines are already in their well defined categories owing to the nature of data collection. We collected a total of 814 clickbait samples and 1574 non-clickbait samples, which have been made available online[7], by taking majority vote as the ground truth. While getting the samples, only the headlines were taken into consideration as these are the ones which create a hook in the reader or viewer. We did not take into consideration the actual web pages behind the headlines, as the headlines are the ones which first come to the attention of the reader and lure the readers to "click" them.

B. Deep Learning Models

Convoluted Neural Networks (CNN) have been utilized for various deep learning tasks. Here, in our present work, we use a simple CNN having one layer of convolution. Fig. 3 shows a graphical representation of the complete model utilized. The CNN we utilize is based on the CNN architecture of Kim [9]. The first layer of the CNN is used for embedding the words into vectors of low-dimensions. For word embeddings we utilize two variants (1) word embeddings which are learnt from scratch, and (2) word embeddings which are learnt from an unsupervised neural language model which keep evolving as training occurs. This technique of initializing word vectors from an unsupervised neural language model has been shown to improve performance [14] [15]. We utilize the word vectors trained by Mikolov, Chen, Corrado and Dean [16] on 100 billion words of Google News. These vectors are publicly available as word2vec.[8].

[7]https://github.com/pfrcks/clickbait-detection
[8]https://code.google.com/p/word2vec/

TABLE I: Accuracy and ROC-AUC scores

Model	Accuracy	ROC-AUC
Click-Scratch	0.89	0.87
Click-Word2vec	0.90	0.90

In the next layer, filters of multiple sizes(3, 4, 5) are utilized to create convolutions over word vectors. Each such operation produces a new feature. All the new features thus generated are put into a feature map. Then, a max-over-time pooling operation [14] is applied over the feature map and the feature with the highest value is taken as the feature for that particular feature map. The next layer, which is the penultimate layer, is formed by such generated features from the filters. These features are then passed to a fully connected softmax layer, having the probability distribution over labels as output.

For measuring loss, we use cross-entropy loss which is a standard for such categorization problems. Our aim is to minimize this loss, which represents the error in our networks. We utilize Adam [17], which is a method for stochastic optimization, for optimization of the loss function of our network.

1) Hyperparameters: Most of the hyperparameters were left untouched like the size of the windows(3, 4, 5), the dropout rate(0.5), embedding dimension of 300, feature map size of 100, etc. and are kept the same as in [9]; however number of epochs was increased to 200 and the batch size for training was increased to 128 for smoothing out the graphs and better evaluation of the test data.

IV. Results

We utilize 5-fold cross validation over the dataset. Our models took around 50 minutes for each run of the cross validation fold when run on an Amazon c3.xlarge EC2 instance. Granted that the training period of a CNN model is significant than that compared to a supervised learning model, the overall time

TABLE II: Performance metrics for both models.

Model	Class	Precision	Recall	F1-Score
Click-Scratch	Clickbait	0.88	0.80	0.84
	Non-clickbait	0.90	0.94	0.92
Click-Word2vec	Clickbait	0.85	0.88	0.86
	Non-clickbait	0.94	0.92	0.93

saved by not undergoing the complex procedure of feature creation is substantial. We compare both (1) scratch and (2) non-static word2vec models, hereby referred to as *click-scratch* and *click-word2vec*. We used accuracy, precision, recall, f1-score and ROC-AUC metrics to evaluate the performance of the model.

For calculating accuracy on the test set, the model was evaluated after every 100 steps to see how the CNN is evolving. Fig. 4a and 4b graph the accuracy over the number of steps for click-scratch and click-word2vec respectively. Table I shows the accuracy and ROC-AUC scores for both the models. Table II show the precision, recall and f1-scores for both classes, clickbait and non-clickbait, for both the models. All the metrics were obtained after averaging the individual metrics over the 5-fold run of the models.

As is evident from the graphs both the models became stagnant after around 2000 steps. This indicates that the model has nothing new to learn and we could probably reduce the number of epochs. This also indicates the need for more data. We also find that the click-scratch model's testing accuracy remains more or less constant while the accuracy of the click-word2vec model keeps on evolving. This can be attributed to the pretrained word vectors utilized for this model and also to the fact that click-word2vec continues to learn and changes the embeddings over the period of time wherein we train our data.

As given by Table I, the *click-word2vec* model outperforms the *click-scratch* model. The F1-Scores for click-word2vec also beat the ones by click-scratch. Hence we see that by taking a basic CNN we are able to achieve a very good model for clickbait detection, without having to go through the time consuming process of feature creation. We also add to the evidence that utilizing pretrained vectors in NLP is a significant step since click-word2vec outperformed click-scratch.

V. Conclusion

The nuisance of clickbait keeps on increasing in online media. To curb that, we collected data from multiple sources and created a new corpus for clickbait and non-clickbait headlines. We then developed a deep learning model based on CNN that performs strongly on the classification of headlines into clickbait and non-clickbait categories. We were able to receive an accuracy of 0.90 along with a precision of 0.85 and a recall of 0.88 on the clickbait class. We aim to make available this model and the corpus for further usage. Future scope includes (1) finding the features that the model has learnt and finding the most important ones, (2) gathering more data for developing better models and (3) coming up with a server-backed web browser plugin which can harness the power of this model and can alert the user about the clickbaits on the page.

References

[1] K. El-Arini and J. Tang, "News feed fyi: Click-baiting," 2014. [Online]. Available: http://newsroom.fb.com/news/2014/08/news-feed-fyi-click-baiting/

[2] J. C. dos Reis, F. Benevenuto, P. O. S. V. de Melo, R. O. Prates, H. Kwak, and J. An, "Breaking the news: First impressions matter on online news," in *Proceedings of ICWSM 2015*.

[3] G. J. Digirolamo and D. L. Hintzman, "First impressions are lasting impressions: A primacy effect in memory for repetitions," *Psychonomic Bulletin & Review*, vol. 4, no. 1, pp. 121–124, 1997. [Online]. Available: http://dx.doi.org/10.3758/BF03210784

[4] D. J. Dooling and R. Lachman, "Effects of comprehension on retention of prose." *Journal of Experimental Psychology*, vol. 88, no. 2, pp. 216–222, 1971.

[5] G. Loewenstein, "The psychology of curiosity: A review and reinterpretation," *Psychological Bulletin*, vol. 116, no. 1, pp. 75–98, July 1994.

[6] B. Gardiner, "Youll be outraged at how easy it was to get you to click on this headline," 2015. [Online]. Available: http://www.wired.com/2015/12/psychology-of-clickbait/

[7] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional networks," in *Proceedings of NIPS 2012*.

[8] A. Graves, A. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of ICASSP 2013*.

[9] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of EMNLP 2014*.

[10] B. Vijgen, "The listicle: An exploring research on an interesting shareable new media phenomenon," *Studia Universitatis Babes-Bolyai-Ephemerides*, vol. 59, no. 1, pp. 103–122, June 2014.

[11] J. Nygaard Blom and K. Hansen, "Click bait: Forward-reference as lure in online news headlines," *Journal of pragmatics : an interdisciplinary journal of language studies*, vol. 74, pp. 87–100, 2015.

[12] M. Potthast, S. Köpsel, B. Stein, and M. Hagen, "Clickbait Detection," in *Advances in Information Retrieval. 38th European Conference on IR Resarch (ECIR 16)*, ser. Lecture Notes in Computer Science, vol. 9626. Berlin Heidelberg New York: Springer, March 2016, pp. 810–817.

[13] P. Biyani, K. Tsioutsiouliklis, and J. Blackmer, ""8 amazing secrets for getting more clicks": Detecting clickbaits in news streams using article informality," 2016. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11807

[14] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, pp. 2493–2597, 2011.

[15] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 151–161. [Online]. Available: http://dl.acm.org/citation.cfm?id=2145432.2145450

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of ICLR 2013*.

[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980