**How Subword Information Using Character n-gram Vectors Facilitate Word Representation for Morphologically Rich Languages**: **An Observational Report**

Supervised by: **Prof. Dr. Achim Rettinger**

By: **Abdullah Shareef**

Matriculation Number: **1717959**

**Table of Contents**

**Abstract**

1. **Introduction**
2. **Fasttext embeddings: general introduction**
3. **The character n-gram approach via the skipgram model**
4. **Methods**

**1. Treatment of words as character n-grams using the skipgram model**

**2. A closer look at character n-gram treatment and the consideration of subword information**

1. **Previous models and related research**
2. **General background**
3. **Character n-gram model used in CWI processes**
4. **Results**
5. **Word analogy tasks**
6. **Morphological representations**
7. **The size of training data**
8. **The size of n-grams**
9. **The potential of the proposed model and future work**
10. **Conclusion**

**Abstract**

Bojanowski et al. employ the skipgram model to develop a new approach that takes into consideration subword information to adequately represent words in morphologically rich language. The model treats words as bags of character n-grams and divides words into sequences of n-grams per word. The model demonstrates superior syntactic performance and outperforms other models and techniques in terms of various baselines and different languages. The model's predictive ability for rare words remains robust as it can predict rare words by making use of word analogy tasks. The model is open source and available for the public to facilitate comparing it with other previous models and future ones.

1. **Introduction**

Word processors consist of various databases of different languages that they have been trained on. They are capable of performing numerous tasks during the typing process. These include predicting, correcting and modifying; more specifically: word representing. As societies develop, so do the languages used by the people living in these societies, and as the need for communication grows, technologies are constantly being updated to accommodate the needs of these societies to realize their communication demands and to connect effectively, as well as quickly, in various languages. For these reasons, word processors are continuously being updated and improved.

According to Zhang et al., “words are commonly represented using vectors, where distances among the vectors are related to the similarity of the words.” (2020, 1). This method assigns a vector for each word, and has been convenient for many years; however, the greater the amount of words found in a language, the bigger the size of training data will have to be. For morphologically rich languages, a word can have dozens of variations, and the notion of assigning a vector for each variation of said word is either illogical or unreasonable; therefore, integrating subword information into word vectors can assist in reducing the size of word datasets and facilitate the process of representing languages which are morphologically rich.

1. **Fast-text embeddings: general introduction**

Mikolov et al. (2013a) mention that “simple models trained on huge amounts of data outperform complex system trained on less data,” however, “simple techniques are at their limits in many tasks.” An example of these limitations is that in machine translation or word representation processors, “the existing corpora for many languages contain only a few billions of words or less. Thus, there are situations where simple scaling up of the basic techniques will not result in any significant progress, and we have to focus on more advanced techniques.” (1). This is the exact case of languages with rich morphology, where it is impossible or unreasonable to assign vectors to every single word variation. Thus, the researchers proposed two new architectures to estimate vector representations efficiently. This task is done by training new model architectures. One has been titled ‘cbow,’ or ‘continuous bag of words.’ This architecture is trained to predict “the current word based on the context,” meanwhile, the other is titled ‘skipgram,’ it “predicts surrounding words given the current word.” (2013a, 5).

1. **The character n-gram approach via the skipgram model**

The two architectural models, skipgram and ‘cbow,’ can be considered as the foundation of the character n-gram approach, which is renewed by Bojanowski et al. to consider subword information relying on the skipgram model. Bojanowski et al. wish to treat the morphology and other additional elements of words, which can change depending on their position in the sentence, as bags of n-gram. The word is thus comprised of sequences, as demonstrated in the following example:

Consider the word banana, and the n-gram is 2-gram.

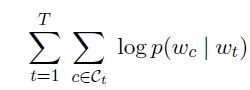
Ba – an – na – an – na

Assigning random, for clarification use, values for each bi-gram in a vector,

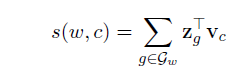
Ba = [0.1, 0,2], an = [0.3, 0.4], na= [0.5, 0.6]

Banana = [0.1, 0,2] + [0.3, 0.4] + [0.5, 0.6] + [0.3, 0.4] + [0.5, 0.6] =[1.7, 2.2 ]

In other words, the skipgram model is designed to achieve the maximum likelihood of the following:



“where the context Ct is the set of indices of words surrounding word wt.” (Bojanowski et al., 2017, 2). Thus subword model, on the other hand, appears as follows:



This model, according to the authors, “allows sharing the representations across words, thus allowing to learn reliable representation for rare words.” (3). Rare words, in this case, can stand for the many variations of common words in languages which are morphologically rich. The subword model facilitates word processing by providing subword information as forms of n-gram vectors.

In brief, the new approach aims to feed word vectors with subword information in order to improve word representation of morphologically rich languages. The character n-gram approach improves the process of predicting rare words and other word variations in a specific language relying on similar words which can be connected together through the training provided by the same skipgram algorithm. The difference here is that instead of concentrating on semantics, the focus is placed on syntax, where the model is trained on two or more words with similar morphemes, thus allowing it to predict similar words which follow the same pattern. For example, the model can be trained on the words ‘humanism’ and ‘feminism,’ in addition to other occurring common words, if possible, like ‘humanist’ and ‘feminist.’ In the process, if the word ‘accelerationist’ appears in the training data, the model will successfully register or predict the noun of this adjective, which is ‘accelerationism,’ etc.

In this report, the character n-gram technique implemented via the skipgram model by Bojanowski et al. will be examined and observed methodically, in addition to the value it adds to previous word representation models and the potential it promises as it progresses.

1. **Methods**

To obtain information regarding the topic of this report, a paper by Bojanowski et al. (2017) was examined and analyzed methodologically, where the contents were investigated and compared with the content and information of other research papers. It is clear that normal or generic language processing models are quite effective when it comes to specific languages. However, some languages are extremely rich morphologically, as their vocabulary shifts and changes depending on various elements. These elements include pronouns, subject-verb agreement, tenses and other factors that preserve the roots of words but integrate them with affixes, suffixes and prefixes that radically make the words seem as though they do not belong to these languages. This is where the method of turning words into bags of character n-grams, using the skipgram model, is mostly effective. According to Bojanowski et al. (2017), their method lies in executing the following: to "model morphology by considering subword units, and representing words by a sum of its character n-grams." (2).

1. **Treatment of words as character n-grams using the skipgram model**

Given that the approach of word representation using the bag-of-words model does not efficiently or sufficiently regulate texts semantically, and the fact that the skipgram model ignores the subword information (since it represents words instead of the characters of words), Bojanowski et al. devised a method where the internal structure of words is not ignored; this was done by making each word ѡ stand as a bag of character n-gram on its own, where the symbols < and > are placed before and after each word, allowing for the ability to specify affixes for each word. This not only trains processors to be familiar with various words with different morphologies in a specific language, it also allows them to predict these words and predict rare words which belong to this language. This process is also done by representing words as character n-grams, where each sequence is taken separately as a standalone affix or inflection of each word in the morphologically rich language.

1. **A closer look at character n-gram treatment and the consideration of subword information**

What distinguishes the model made by Bojanowski et al. from others is that it divides words into sequences of 3 grams. Unlike other models which divide them into sequences of 2 or 6. An interesting example of this is found in the paper by Popovic (2018), where the author used character n-grams for complex word identification, where it was found that "combination of 2-grams and 4-grams is the best option for the standard setting," whereas "individual 3-grams, 4-grams and 5-grams outperforms the combinations when a larger English corpus is used." (344). Thus, it is observed that Popovic's approach relies heavily on the data found within the corpus, while the approach employed by Bojanowski et al. "seems to quickly saturate and adding more data does not always lead to improved results." (7). Popovic experimented with the bag of n-gram lengths, from 2 to 6, and found that lengths over 6 did not exhibit improvements. Bojanowski et al. mainly utilized a length of 3, which appears ideal for many languages, given that most of one of these languages’ vocabulary, such as German and Finnish, consist of inflected forms with various morphemes. What Bojanowski et al. resort to solely relies on subword information and they report finding that the "arbitrary choice of 3-6 was a reasonable decision, as it provides satisfactory performance across languages. The optimal choice of length ranges depends on the considered task and language and should be tuned appropriately," since "taking a large range such as 3 – 6 provides a reasonable amount of subword information." (7). Again, the numbers stand for the character n-grams or lengths of these n-grams.

1. **Previous models and related research**
2. **General background**

Given that Bojanowski et al. have relied on the skipgram model to complete their research, it is proper to investigate the model more closely. Zhang et al. (2020) have a research paper titled "An Analysis on the Learning Rules of the Skip-Gram Model," where they meticulously examine the model and conduct various experiments explaining the processes the model undertakes when representing words using vectors. The research paper contributed enormously to understanding the approach which Bojanowski et al. have utilized, as it explained the learning rules of the skipgram models, as well as demonstrating the limitations and difficulties of using the model, in addition to the suggested improvements which can be applied to develop it.

A method of implementing the skipgram model is called word2vec, or Word-to-Vector, which is a word representation method where words are represented as vectors. Furthermore, the process of implementing this method relies on two separate layers of vector representations: the input layer and the output layer, or the input vector and the output vector, respectively. The input layer is fed or trained using various articles and textual data from Wikipedia and the internet, and by using and optimizing a specific word2vec code, the processor's capability of predicting the correct context words is improved. According to Zhang et al., the word2vec implementation is "widely used due to its computational efficiency and its ability to capture interesting analogue relationships," moreover, "systems built on word2vec representations often lead to significant performance improvements." (2020, 1). This is relevant to the model developed by Bojanowski et al., as their work was syntactically ideal; however, it required more investigation regarding its semantic word representation tasks.

1. **Character n-gram model used in CWI processes**

The character n-gram model has been used and experimented on previously, specifically in CWI processes, or Complex Word Identification, as Popovic (2018) points. The paper by said author investigates the use of character n-grams in processes of complex word identification. The author points to how this model has been successful in “machine translation evaluation metrics in recent years,” and how “these metrics correlate very well with human judgments for all analysed target languages, which indicates that character sequences carry some important information.” (341). Thus, the use of character n-gram in translation has proved that it has been correlative with human judgement; therefore, it is vital and has great potential in word processing task and word representation tasks. The model is proven to be syntactically more developed and outperforms other models; Bojanowski et al. have applied a method named ‘sisg’ or ‘Subword Information Skip Gram,’ where subword information, affixes and morphological elements are treated as forms of character n-grams. Their method proved that “morphological information significantly improves the syntactic tasks; our approach outperforms the baselines. In contrast, it does not help for semantic questions.” (2018, 5). Thus, the treatment of words as character n-grams is only semantically superior in machine translation tasks, but not in word representation and processing tasks.

Popovic lastly states that the system they proposed ranked in “middle-range position for all tracks except for the cross-lingual track where it was ranked very,” which is due to the fact that “frequencies of character sequences in words are intuitively rather language-dependent.” (2018, 347). This information provides vital insight that paves the way to understanding how character n-gram models and subword information function in word processing tasks. Although Popovic’s approach was mainly focused on word identification tasks, it nevertheless is tied directly with word representation tasks, and can help researchers understand the models and methods further.

1. **Results**
2. **Word analogy tasks**

Many factors come into play in the model proposed by Bojanowski et al. As mentioned previously, the "sisg" method utilizes character n-grams as subword information to improve word representation. It was noted that this method "outperforms the baselines on all datasets except the English WS353 dataset. Moreover, computing vectors for out-of-vocabulary words (sisg) is always at least as good as not doing so (sisg-). This proves the advantage of using subword information in the form of character n-grams." (2017, 5). Furthermore, the accuracy of tasks involving word analogy also outperform other models syntactically, as show in the table below taken from the paper:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Language | Analogy | Sg | cbow | Sisg |
| Czech | |  | | --- | | Semantic | | Syntactic | | |  | | --- | | 25.7 | | 52.8 | | |  | | --- | | 27.6 | | 55.0 | | |  | | --- | | 27.5 | | 77.8 | |
| German | |  | | --- | | Semantic | | Syntactic | | |  | | --- | | 66.5 | | 44.5 | | |  | | --- | | 66.8 | | 45.0 | | |  | | --- | | 62.3 | | 56.4 | |
| English | |  | | --- | | Semantic | | Syntactic | | |  | | --- | | 78.5 | | 70.1 | | |  | | --- | | 78.2 | | 69.9 | | |  | | --- | | 77.8 | | 74.9 | |
| Italian | |  | | --- | | Semantic | | Syntactic | | |  | | --- | | 52.3 | | 51.5 | | |  | | --- | | 54.7 | | 51.8 | | |  | | --- | | 52.3 | | 62.7 | |

Table 1 Accuracy of Bojanowski et al's model and baselines on word analogy tasks for listed languages. Results for semantic and ssyntactic analogies are reported separately. (Bojanowski et al., 2017, 5).

1. **Morphological representations**

The authors further indicate that despite the degradation of performance semantically, the improvement in syntax is of greater value for morphologically rich languages. This is indeed the case with these results; given that improvements can be observed in the syntactic context, the semantic context can be developed with the integration of other models that outperform others semantically. What the researchers did initially was building on a previous model, and this can be done in future works in other contexts to enhance word representation for morphologically rich languages.

1. **The size of training data**

The size of the training data was additionally evaluated to study any chance of improving word representation. The researchers trained their model “portions of Wikipedia of increasing size,” but found that increasing the size of data “does not always lead to improved results.” However, they observed that their approach “provides very good word vectors even when using very small training datasets.” (Bojanowski et al. 2017, 7). This is demonstrated in the following table:

|  |  |  |
| --- | --- | --- |
|  | Sisg | Cbow |
| German GUR350 dataset | |  | | --- | | Percentage: 5% | | Performance: 66 | | |  | | --- | | Percentage: full dataset | | Performance: 62 | |
| English RW dataset | |  | | --- | | Percentage: 1% | | Performance: 45 | | |  | | --- | | Percentage: full dataset | | Performance: 43 | |

Table 2 Performance of the sisg model compared with the cbow baseline depending on the percentage of the datasets the two were trained on.

This implies that the vectors are able to represent words which have not been seen before; thus assisting greatly in identifying many forms of words in languages which are morphologically rich. Furthermore, the process of training this model is significantly shorter, whether trained on full datasets or otherwise. As mentioned by Mikolov et al, “subsampling of frequent words during training results in a significant speedup (around 2x – 10x), and improves accuracy of the representations of less frequent words.” (2013b, 2). The goal of Bojanowski et al., to begin with, is to improve representing rare words.

1. **The size of n-grams**

As mentioned in the methods section, Bojanowski et al. chose a length of 3 to 6 characters for each n-gram, and found that it was a reasonable choice. In word analogy tasks, they also observed that “­­­­ using larger n-grams helps for semantic analogies. However, results are always improved by taking n ≥ 3 rather than n ≥ 2, which shows that character 2-grams are not informative for that task.” (7). Again, 2-grams are not sufficient for words with long roots, morphemes or affixes, and 3-grams provide better performance than lesser lengths.

1. **The potential of the model and future work**

The model proposed by Bojanowski et al. builds on previous models, such as the skipgram and the ‘cbow’ architectures. It aims towards representing words by considering subword information. The model, according to the authors, is "capable of building word vectors for words that do not appear in the training set." (2017, 9). This advantage is a significant step in word representation for languages which are morphologically rich, and the model promises to encompass more languages as machine training progresses. Furthermore, in the same manner that the skipgram and cbow models were used as the foundation for the proposed model, it is possible to utilize various others which can improve the model's function semantically, and make it surpass previous word representing models. Future work should focus on improving and advancing the model in terms of semantics, while at the same time maintaining its syntactic performance. The model is open-source and is available for the public on the following link: <https://github.com/facebookresearch/fastText>

1. **Conclusion**

Relying on the skipgram model, it is possible to incorporate sub-word information as character n-grams to improve the word representation tasks for morphologically rich languages. The model developed by Bojanowski et al. can train fast on significantly small amounts of data. Furthermore, it demonstrates superior performance over models that overlook sub-word details or information. The model is open-source and aims to encourage and facilitate future comparisons and advancements in sub-word representation learning. Moreover, it is an important breakthrough for word processors and word representation tasks, specifically for people who use those and whose languages are rife with morphemes, affixes and elements that change according to the words’ positions and types.

**References**

Bojanowski, P., Grave, E., Joulin, A., Mikolov, T. (2017). *Enriching Word Vectors with*

*Subword Information*. <http://arxiv/abs/1607.04606v2>

Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). *Efficient Estimation of Word*

*Representations in Vector Space*. <http://arxiv.org/abs/1301.3781v3>

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J. (2013). *Distributed*

*Representations of Words and Phrases and their Compositionality.* <http://arxiv.org/abs/1310.4546v1>

Popovic, M. (2018). Complex Word Identification Using Character n-grams. *Proceedings of*

*the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 341–348.

Zhang, C., Liu, X., Bis, D. (2020). An Analysis on the Learning Rules of the Skip-Gram

Model. *International Joint Conference on Neural Networks,* 1-8.