RooAd: A Computationally Creative Online Advertisement Generator

Mika Hämäläinen and Khalid Alnajjar
Rootroo Ltd
Helsinki, Finland
firstname@rootroo.com

Abstract

Automated generation of textual advertisements for specific products is a natural language generation problem that has not received too wide a research interest in the past. In this paper, we present a genetic algorithm based approach that models the key components of advertising: *creativity*, *ability to draw attention*, *memorability*, *clarity*, *informativeness* and *distinctiveness*. Our results suggest that our method outperforms the current state of the art in *readability* and *informativeness* but not in *attractiveness*.

1 Introduction

Generation of a variety of different kinds of creative text has received quite a lot of attention in the past years ranging from story generation (Concepción et al., 2016; Fan et al., 2018) to poem generation (Loller-Andersen and Gambäck, 2018; Hämäläinen and Alnajjar, 2019a) and humor generation (Weller et al., 2020; Alnajjar and Hämäläinen, 2021). However, one task of creative text generation that has eluded an extensive research is advertisement generation.

Advertisements need to be appealing, informative, catchy and novel. Novelty is a trait that is very important in advertising as reusing another company's advertisement for your product might in fact work in the favor of the competing company. As we will see in the related work section, the few existing approaches to advertisement generation fail to take the novelty into account.

We present a genetic algorithm based approach for computationally creative advertisement generation. We focus on short textual advertisements for products as our advertisement generator is designed to be a part of a larger online product recommendation system. For a given product recommendation, our method generates a short advertisement message.

We model our system in such a fashion that it aims to maximize the features commonly associated with computational creativity in its output. These features are *novelty*, *value* and *typicality* as identified by (Ritchie, 2007). Another way of defining computational creativity is a creative tripod framework by (Colton, 2008). According to this view, a creative system should exhibit *skill*, *imagination* and *appreciation*.

While there is a clear overlap between novelty and imagination in the two definitions, the creative tripod brings an interesting point of view to how value should be modeled. For appreciation refers to the computational system's own capacity of estimating how good or valuable its own output is based on several different parameters. Therefore, for our system it is not enough that people can see value in its output, but the system itself should also be able to evaluate its own output. The notion of typicality can be contrasted to the notion of skill; a system that has the skill of generating advertisements, must make the advertisements typical enough for them to be recognized as such.

2 Related Work

A very early approach to advertisement generation was presented by (Somers et al., 1997). They generated e-mail ads for open job positions by using a schema based approach. They store information related to the job position offered in a rule-based schema. Their system takes in ads written by people and parses them into a schema that is stored into a job ad database. When a user is looking for a job in the system, their system generate job ads based on the database and a set of grammar rules and templates. The authors do not present any evaluation of their approach.

A more recent approach that is slightly related to ad generation, is the generation of advertising plots based on a human-conducted analysis of ad videos (Ono et al., 2019). Rather than doing direct advertisement generation, the authors approach the problem from the point of view of narrative generation as the goal of their system is to generate plots for ad videos. The narratives are generated by using three principal building blocks: events, their relations and the state of the narrative.

The most recent work on advertisement generation uses two neural networks for the task; one is used for generation and the other for selection (Chan et al., 2020). They use a multi-agent communication framework in the generative neural network. They present a human evaluation of their approach, which we will use to also evaluate our system.

3 Data

Given the scarceness of publicly available ad corpora, we construct a our corpus by downloading good example ads crafted by well-known brands on social media platforms (such as Facebook, Twitter and Instagram) from AdEspresso's Academy[1]. AdEspresso provides such ads as an inspiration for beginner advertisers; however, the ads are provided as screenshots in a PDF format. To tackle this issue, we manually transcribed the textual descriptions in them. In total, the corpus contains around 1400 ad descriptions.

Following the work described by (Alnajjar and Toivonen, 2020), we build a repository of ad skeletons where ad descriptions are syntactically-parsed using spaCy (Honnibal and Montani, 2017) and, then, any content words are replaced with a placeholder. Skeletons act as an initial block for the method to build on by filling and continuously altering placeholders with words while satisfying grammatical constraints defined by the syntactical relations and optimizing multiple criteria.

We utilize a dataset of 12 million English grammatical relations (Alnajjar, 2018). A grammatical relation consists of a token, its head-token, the parts-of-speech of both tokens and the type of relation such as *nsubj* and *advmod*. In the following section we describe how these resources are harnessed in our approach.

[1]https://adespresso.com/

4 Generating Ads

In this section, we describe our genetic algorithm based approach for advertisement generation. Before doing so, it is important to define what the meaningful attributes are for advertising in general.

(Dahl, 2011) identifies six important attributes for advertisements: *creativity* (novelty), *ability to draw attention*, *memorability*, *clarity*, *informativeness* and *distinctiveness*. These are the features that we model computationally in our generative system. For the sake of simplicity, we treat creativity and distinctiveness as one attribute as they are near synonyms; both are referring to a degree of novelty in an ad. These are related to the computational creativity notions of novelty and imagination.

The remaining of the attributes are assimilated with the notions of appreciation and value in computational creativity. It is therefore important that the system is capable of assessing them individually instead of producing a single confidence score representing all of them.

The skeletons extracted in the previous section contribute to the typicality and skill of the system. When the generated ads follow an ad-like pattern and are grammatical, they are perceived more easily as ads. It is important that the output remains very ad-like as a familiar structure will make the generated ads be perceived more positively by the audience (c.f. (Veale, 2016)).

4.1 Genetic Algorithm

We opt for a genetic algorithm approach following the implementation presented in (Alnajjar et al., 2018; Alnajjar and Hämäläinen, 2018) on the DEAP tool (Fortin et al., 2012). Our implementation of the genetic algorithm takes in a random ad skeleton from the ad skeleton corpus and uses it to produce an initial population of 100 individuals. These individuals produce an offspring of another 100 individuals that go through mutation and crossover as a part of the genetic process. At the end of each generation, the individuals (ads) are scored according to the fitness functions defined

	Readability	Informativeness	Attractiveness	Rationality
Our approach	**3.672**	**3.528**	3.411	**3.373**
Chan et al., 2020	3.645	3.395	**3.500**	-

Table 1: Averages of the human evaluations in comparison with the current state-of-the-art

later in the following subsection. The 100 fittest individuals are selected with NSGA-II algorithm (Deb et al., 2002) to survive to the next generation. The individuals are picked both from the offspring and the current population so that the quality of the generated ads cannot degrade from one generation to another. This process is done for 200 generations.

All individuals in the initial population are based on a randomly selected ad and the name, description and category of the product to be advertised. We populate each ad skeleton in the initial population once by retrieving, from the grammatical relations dataset, candidate substitutions that comply with the syntactical rules imposed by the ad description. The filling process starts with the *ROOT* relation in the skeleton and continues until all placeholders are replaced with content words. Additionally, proper nouns in the skeleton are replaced by the product name that is to be advertised. This process is applied to each individual in the population, resulting in different variations of ads for the same skeleton.

In the mutation step, a random content word is picked in the ad and it is replaced by a word related to the input product in terms of the category or the description, while ensuring that the grammaticality of the expression is intact by validating that the introduced change appears in the grammatical relations dataset at least 10 times.

In terms of the crossover, we employ a single-point crossover on a word-level where one point in both individuals is selected at random and word to the right of that point are swapped.

4.2 Fitness Functions as an Internal Metric of Value

As the genetic algorithm is in the process of execution, it has to have a way of ranking its advertisements so that it can move the fittest ones to the next population and discard the worst ones. Our system uses the following five methods to rank the individual attributes of advertisements as identified by (Dahl, 2011).

4.2.1 Creativity/Distinctiveness

Novelty is an important factor in advertising, and in creative text generation, it is a parameter that is often overlooked. The degree to which a machine learning model just reproduces its training data is hardly ever discussed in any contemporary creative text generation approach (c.f. (Hämäläinen, 2020)).

In order to maximize novelty, we compare a given ad to all the ads in our ad corpus. We do this by counting BLEU scores (Papineni et al., 2002) that indicate how similar a generated ad is to an existing one. This fitness function outputs the highest BLEU score with an existing ad, and our genetic algorithm tires to minimize this parameter.

4.2.2 Ability to draw attention

We see ads all the time, but a successful one requires us to pay attention to itself, for attention is what turns what is merely seen into something that is perceived by our conscious mind (c.f (Wolfe et al., 2006)). Our brains process our surroundings by forming hypotheses and focusing less on things that follow those hypotheses and more on things that do not quite fit in. In fact, it has been argued for a long time that there is a link between surprise and attention (Horstmann, 2015). When we see something surprising, our attention is more likely to be drawn towards the surprising element.

For measuring surprise (Bunescu and Uduehi, 2019) propose using a language model (named audience model) that is separate from the model (called composer model) that is used to generate text. With the same idea, we use an AWD LSTM based language model (Merity et al., 2018) trained on another corpus to measure surprise. The less probable a sentence is according to the model, the more surprising it is. This fitness function outputs the average probability of the sentences in the ad. The genetic algorithm minimizes this value.

4.2.3 Memorability

There are several ways of improving recall in the form of applying mnemonics. The most common way for advertisements of achieving this is ensuring catchiness in the message. One way of making a message catchy is by introducing rhyme.

Rhyming is also a method for increasing memorability (c.f. (Lindstromberg and Boers, 2008)).

This fitness function counts the number of words that have rhyming pairs in the ad and divides it by the number of words, in other words it returns the proportion of words that at least rhyme with one other word in the ad. We consider several different types of rhyme: consonance, assonance, alliteration and full rhyme. We model this with simple rules. Because in English it is difficult to know how well words rhyme together based on their written form, we use Espeak-ng[2] to produce IPA transcription for each word similarly to (Hämäläinen and Alnajjar, 2019b). As IPA is supposed to relatively closely model how words are pronounced, it makes it possible to detect rhyming more accurately. The genetic algorithm tries to maximize this fitness function.

4.2.4 Clarity

For clarity, we use a previously established metric for estimating how readable texts written in English are. Flesch Reading Ease is a metric that takes into account the number of words per sentence and the number of syllables per words, with the idea that longer words and sentences result in less readable text. The higher the score, the more readable the text is. We calculate the score for each ad and our genetic algorithm tries to maximize this fitness function.

4.2.5 Informativeness

An informative ad communicates effectively information about the product. In order to ensure the ad describes the product as well as possible, we compare the meaning of the content words to the keywords of the product from its description. The comparison is done by calculating the semantic similarity of each content word in the ad with each one of the keywords by using the English FastText model by (Grave et al., 2018). The maximum similarity is picked for each word and the fitness function returns their average as a result. The genetic algorithm maximizes this value.

5 Results and Evaluation

For evaluation, we follow the evaluation approach established by (Chan et al., 2020). They evaluated their state-of-the-art approach by producing 200 ads with their system and having 3 human evaluators go through them. The evaluators were asked to rate the ads based on *readability*, *informativeness*, *attractiveness* and *rationality* on a scale from 1 to 4 (from bad to good).

We replicate their evaluation method in order to able to make a comparison to the current state-of-the-art possible. Similarly to them, we use our system to produce 200 ads for different tech products (each ad is for a different product). We present the product together with its corresponding randomly sampled ad to 3 evaluators. The first three evaluation questions are the same[3] as the previous work: *Is the ad grammatically formed and smooth?* (readability), *Does the ad contain informative words*? (informativeness), *How attractive is the ad?* (attractiveness) and *Is the ad suitable for the product?* (rationality). The last evaluation question is different for us as our system does not do product recommendation[4].

The results of the evaluation can be seen in Table 1. Our approach outperforms the current state-of-the-art in readability and informativeness, but is worse on attractiveness. The results also suggest that our method is capable of producing ads that are suitable for the product being advertised.

Are you a gamer? Nintendo Switch gives
you all the great games, experiences
and skills you want. Enhance your
gaming into the extreme
with Nintendo Switch

Above is an example of an ad produced by the system for Nintendo Switch. The advertising messages the system produces are designed to be shown in an online store for products recommended by an external system.

6 Conclusions

We have proposed a new method for generating advertisements automatically. Our method can outperform the state-of-the-art in two out of three common evaluation metrics.

We have taken an approach that is based on the main important notions of advertisements, each of which has been modeled independently as a part of the genetic algorithm. These notions, how they have been implemented in the system and their relation with computational creativity has been discussed extensively.

[2]https://github.com/espeak-ng/espeak-ng

[3]The only difference is that we use the word *ad* instead of *copywriting*

[4]The previously used question was *Is the product selection reasonable?*

Our method achieves novelty in ads as it is not trained on any existing advertisements and it continuously minimizes the similarity of its output with existing ads. At the same time it exhibits what is required by the notion of appreciation as it has several methods for assessing its own output.

References

Alnajjar, Khalid and Mika Hämäläinen. 2018. A master-apprentice approach to automatic creation of culturally satirical movie titles. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 274–283.

Alnajjar, Khalid and Mika Hämäläinen. 2021. When a computer cracks a joke: Automated generation of humorous headlines. In *Proceedings of the 12th International Conference on Computational Creativity (ICCC 2021)*. Association for Computational Creativity.

Alnajjar, Khalid and Hannu Toivonen. 2020. Computational generation of slogans. *Natural Language Engineering*, page 1–33.

Alnajjar, Khalid, Hadaytullah Hadaytullah, and Hannu Toivonen. 2018. “Talent, Skill and Support.” A method for automatic creation of slogans. In *Proceedings of the 9th ICCC*, pages 88–95.

Alnajjar, Khalid. 2018. The 12 million most frequent English grammatical relations and their frequencies. https://doi.org/10.5281/zenodo.1255800, May.

Bunescu, Razvan C and Oseremen O Uduehi. 2019. Learning to surprise: A composer-audience architecture. In *ICCC*, pages 41–48.

Chan, Zhangming, Yuchi Zhang, Xiuying Chen, Shen Gao, Zhiqiang Zhang, Dongyan Zhao, and Rui Yan. 2020. Selection and generation: Learning towards multi-product advertisement post generation. In *Proceedings of EMNLP*, pages 3818–3829.

Colton, Simon. 2008. Creativity versus the perception of creativity in computational systems. In *AAAI spring symposium: creative intelligent systems*, volume 8.

Concepción, Eugenio, Pablo Gervás, and Gonzalo Méndez. 2016. Mining knowledge in storytelling systems for narrative generation. In *Proceedings of the INLG 2016 Workshop on Computational Creativity in Natural Language Generation*, pages 41–50.

Dahl, Gary. 2011. *Advertising for dummies*. John Wiley & Sons, Hoboken, New Jersey, United States.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans. Evol. Comp*, 6(2):182–197, April.

Fan, Angela, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th ACL (Volume 1: Long Papers)*, pages 889–898, July.

Fortin, Félix-Antoine, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. 2012. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul.

Grave, Edouard, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of LREC 2018*.

Hämäläinen, Mika and Khalid Alnajjar. 2019a. Generating modern poetry automatically in finnish. In *Proceedings of EMNLP-IJCNLP*, pages 6001–6006.

Hämäläinen, Mika and Khalid Alnajjar. 2019b. Modelling the socialization of creative agents in a master-apprentice setting: The case of movie title puns. In *Proceedings of the 10th International Conference on Computational Creativity*. Association for Computational Creativity.

Hämäläinen, Mika. 2020. *Generating Creative Language-Theories, Practice and Evaluation*. University of Helsinki.

Honnibal, Matthew and Ines Montani. 2017. spaCy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Horstmann, Gernot. 2015. The surprise-attention link: a review. *Annals of the New York Academy of Sciences*, 1339(1):106–115.

Lindstromberg, Seth and Frank Boers. 2008. The Mnemonic Effect of Noticing Alliteration in Lexical Chunks. *Applied Linguistics*, 29(2):200–222, 06.

Loller-Andersen, Malte and Björn Gambäck. 2018. Deep learning-based poetry generation given visual input. In *ICCC*, pages 240–247.

Merity, Stephen, Nitish Shirish Keskar, and Richard Socher. 2018. Regularizing and optimizing lstm language models. In *International Conference on Learning Representations*.

Ono, Juumpei, Atsushi Sasaki, and Takashi Ogata. 2019. Advertising plot generation system based on comprehensive narrative analysis of advertisement videos. In *Proceedings of the 4th Workshop on Computational Creativity in Language Generation*, pages 39–46.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th ACL*, pages 311–318.

Ritchie, Graeme. 2007. Some empirical criteria for attributing creativity to a computer program. *Minds and Machines*, 17(1):67–99.

Somers, Harold, Bill Black, Joakim Nivre, Torbjörn Lager, Annarosa Multari, Luca Gilardoni, Jeremy Ellman, and Alex Rogers. 1997. Multilingual generation and summarization of job adverts: the tree project. In *Fifth Conference on Applied Natural Language Processing*, pages 269–276.

Veale, Tony. 2016. The shape of tweets to come: automating language play in social networks. *Multiple Perspectives on Language Play. Mouton De Gruyter, Language Play and Creativity series*, pages 73–92.

Weller, Orion, Nancy Fulda, and Kevin Seppi. 2020. Can humor prediction datasets be used for humor generation? humorous headline generation via style transfer. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 186–191.

Wolfe, Jeremy M, Keith R Kluender, Dennis M Levi, Linda M Bartoshuk, Rachel S Herz, Roberta L Klatzky, Susan J Lederman, and DM Merfeld. 2006. *Sensation & perception*. Sinauer Sunderland, MA.