Documentation for the Headline Scoring

Table of Contents

1. Introduction

The project's objective is to create an AI-powered algorithm that can assess these headlines according to their effectiveness in communicating the main idea, clarity, and alignment with the target keywords. A scoring system ranging from 1 to 10 will be used for the evaluation procedure.

2. Project Overview

The goal of this project is to use a specially created grading algorithm to assess the quality of Google Ad Headlines produced for the website ronspotflexwork.com. The purpose of the headlines was to advertise parking automation technologies, particularly for office parking. A list of pre-established keywords about automated parking, secure parking, and parking management forms the basis of the generated headlines.

The objective of this test is to design an algorithm that evaluates these headlines on a scale of 1-10, considering their relevance, clarity, potential impact, and alignment with the target keywords. The algorithm should be able to automatically score each headline based on these criteria.

3. Code Implementation Details

- Extracting Keywords

This code tokenises headlines and keywords to preprocess and analyse text using NLTK (Natural Language Toolkit). To make the text more consistent, it is first changed to lowercase. Next, non-alphabetic words and common stopwords (such as "the" or "and") are eliminated. Words are lemmatized—that is, reduced to their root form—after tokenisation (e.g., "parking" becomes "park"). The code uses `Counter` from Python's `collections` module to count the frequency of each word across both datasets after processing the ad **headlines** and a list of **keywords**. Lastly, the most prevalent terms found in the

headlines and keywords are shown, offering a glimpse into the terms that appear most frequently.

- Scoring Algorithm

This method is intended to give an automated and data-driven means of assessing the efficacy of ad headlines, guaranteeing that they are compelling, impactful, relevant, and clear.

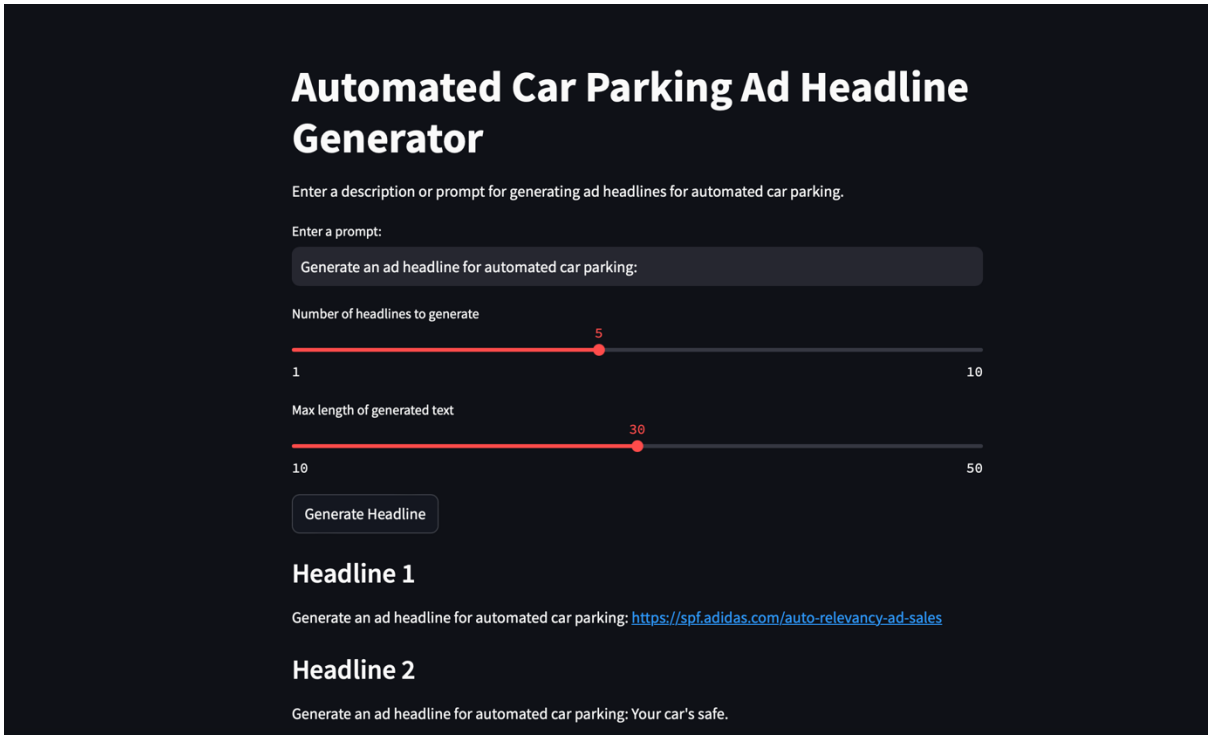
• Relevance: Uses cosine similarity to gauge how well the headline aligns with the desired keywords.
• Clarity: Uses headline length to evaluate clarity, penalising extremely brief headlines.
• Conciseness: Promotes compelling headlines that are shorter, penalising lengthy ones.
• Persuasiveness: Use action words to gauge how compelling and action-oriented the headline is.

- Headline generation

The GPT-2 model, a language model trained for natural language production tasks, is used in this code to generate text using the `transformers` package. The "text-generation" task is loaded using the `pipeline` function, which enables the model to produce text in response to a prompt. The prompt here is "Generate an ad headline for automated car parking." Ten different ad headline variations with a maximum length of 30 characters each are then produced by the `generator` function. Lastly, a loop of the created headlines is printed, showing each result individually.

4. Streamlit App for Generating Headline

The app features a title and a brief description that explains its purpose, which is to help users generate ad headlines. To direct the creation of the headline, users can insert a custom prompt in the text input area. Users can regulate how many headlines are generated and how long each headline can be with the help of movable sliders. The application uses the GPT-2 model to create headlines based on the prompt once the user selects the "Generate Headline" button. The generated headlines, each labelled consecutively, are then shown beneath the button. Although the functionality and appearance of the app can be further altered, this configuration offers an intuitive user experience for interactively crafting ad headlines.



5. Conclusion

Overall this code helps to score the headlines and generate believable training data for the headlines and keywords.