Assessing the Trustworthiness of Healthcare Advertisements

HEART: How to Evaluate an Advertisement's Reliability and Trustworthiness

Ankit Saxena
Master of Behavioral and Decision Sciences, University of Pennsylvania
asaxena7@sas.upenn.edu

Capstone Project Summary

Customers and regulatory agencies are increasingly becoming aware of tactics used by advertisers to increase sales. Organizations like the FDA actively monitor ads in the pharmaceutical industry to ensure the firms follow laws like the Fair Balance law. Moreover, a smarter customer base trusts companies that are transparent in their communication. Hence, the best strategy in this landscape is to be transparent to build trust.

There are clear competitive advantages to being transparent in one's business practices to position oneself as trustworthy. Additionally, in industries like healthcare, being transparent is an ethical concern since lives are at stake.

The HEART framework will allow promotion review teams and regulatory organizations to evaluate and score the trustworthiness of pharmaceutical marketing content.

Acknowledgment

Special thanks to Gregory Szwartz, Ellie Chessen, Anna Dunietz, and Sarah Godby for guiding me throughout the summer in our weekly meetings to help me complete this project. Also, a shout out to Jennifer Machiaverna , Michael Greene, and Mark Miller for helping me with specific aspects of the project.

This was an excellent learning experience and I hope to apply these ideas in future projects to build amazing products and design creative solutions.

– Ankit Saxena

Contents

Importance of Gaining Customer Trust

Assessing the Trustworthiness of Advertisements

Details of the HEART Framework

Expanding the HEART Framework

Importance of Gaining Customer Trust

Importance of Gaining Customer Trust

What consumers think about the brands they trust

Highlights of survey results on brand trust conducted by a marketing consultancy firm across 8 markets, namely, Brazil, China, France, Germany, India, Japan, the UK, and the US on 16,000 participants in 2019.

81%

Buy from brands that they trust to do what's right

67%

Will stop buying from a brand if they lose trust

69%

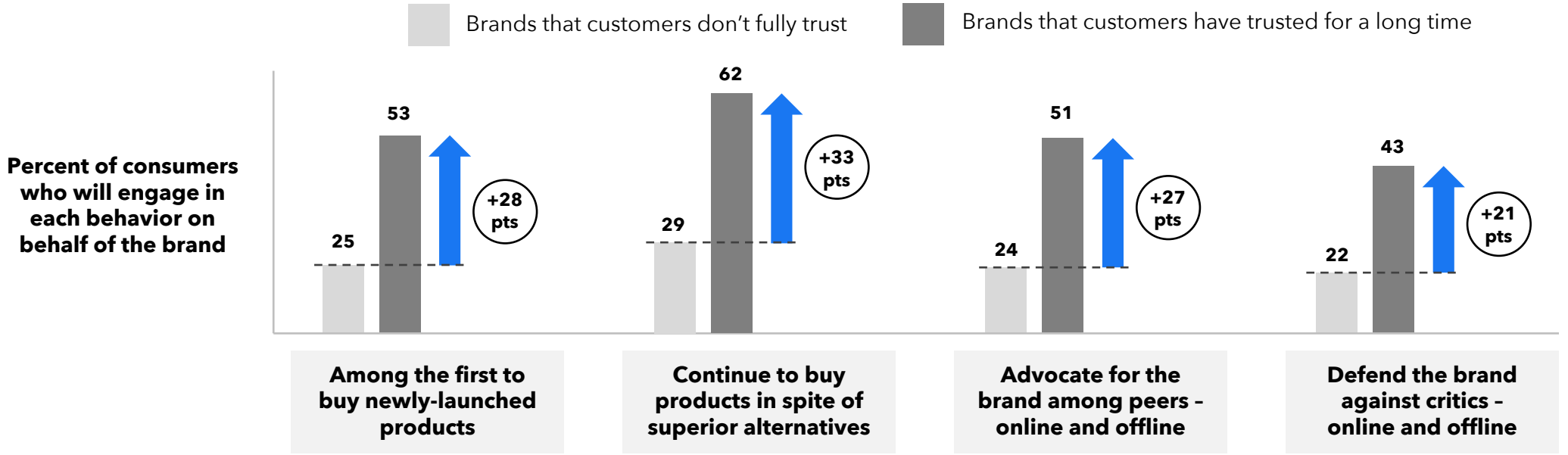
Concerned about misleading advertisements

Source: Edelman | Trust Barometer Special Report | 2019

Importance of Gaining Customer Trust

How consumers behave towards the brands they trust

Highlights of survey results on brand trust conducted by a marketing consultancy firm across 8 markets, namely, Brazil, China, France, Germany, India, Japan, the UK, and the US on 16,000 participants in 2019.



Source: Edelman | Trust Barometer Special Report | 2019

Importance of Gaining Customer Trust

How building trust helps agents involved in pharma advertising

Gaining customer trust affects the following goals of agents and decision-makers involved in the pharmaceutical advertisement process.

FDA

- Enforce the Fair Balance law which states that the ad should balance risks and benefits in the text, design, color, etc.
- Enforce the HIPAA privacy rule meant to protect the personal data of patients
- Block misleading and untruthful claims
- Protect the well-being of consumers

PRC

- Ensure all negative effects are disclosed
- Prevent legal violations of intellectual property, fraud, kickback regulations, etc.
- Guarantee that the data are up-to-date
- Verify that the marketing claims are backed by reference materials
- Ensure the clinical trial results are mentioned

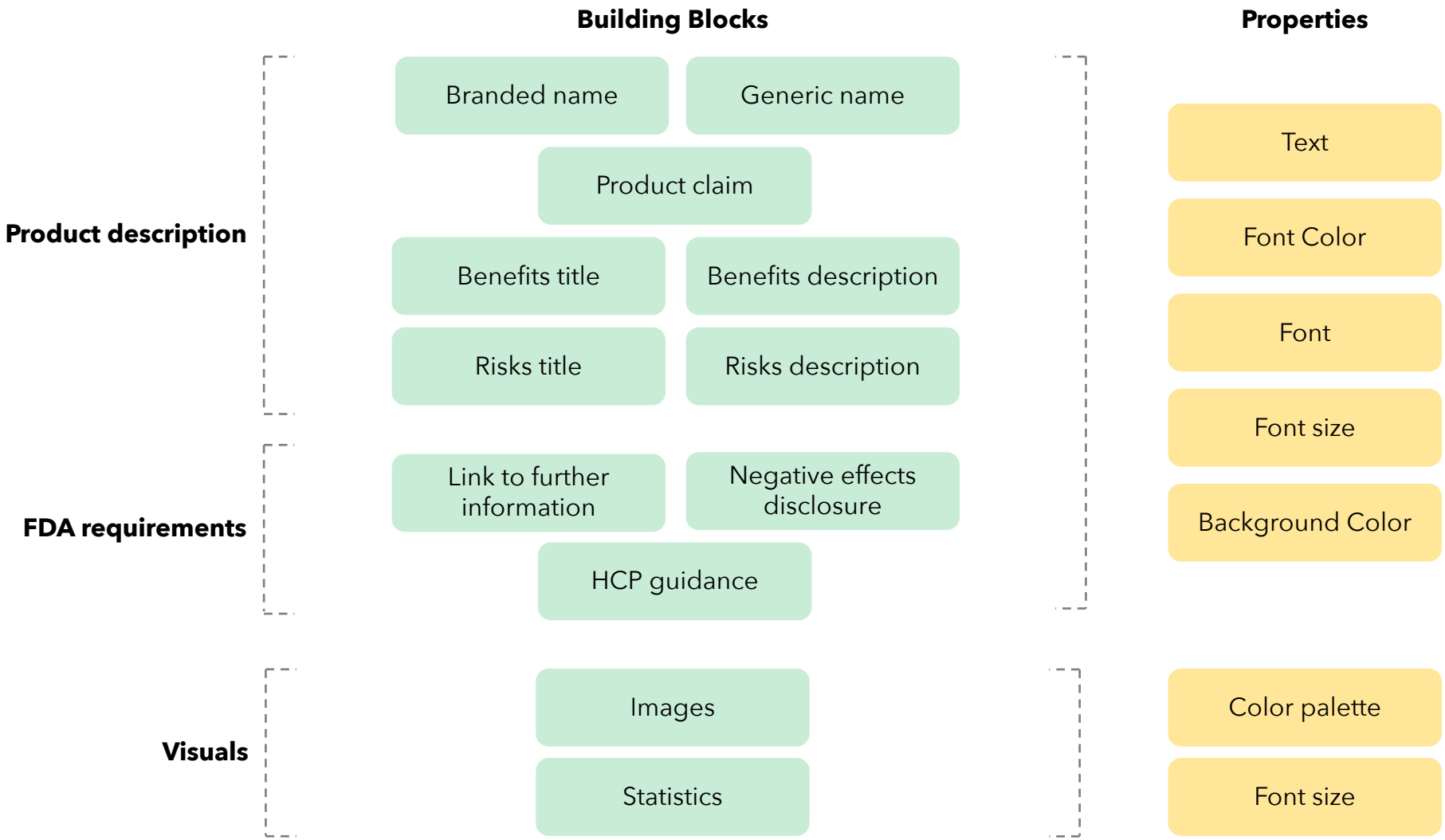
Marketing Team

- Strengthen long-term customer relationships by building trust and loyalty
- Increase demand and generate leads for the sales team in the B2B2C channel
- Conduct market research
- Collaborate with ad agencies for best practices

Assessing the Trustworthiness of Advertisements

Assessing the Trustworthiness of Advertisements

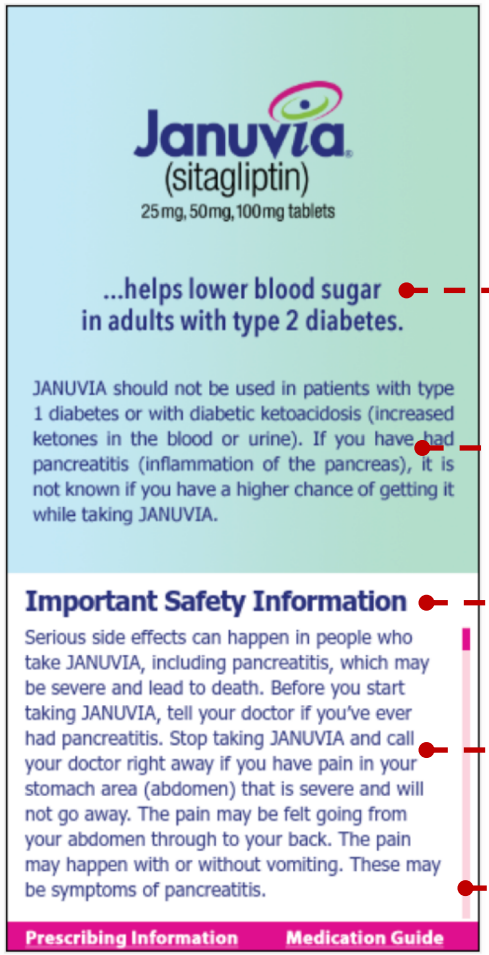
The building blocks of healthcare advertisements



Assessing the Trustworthiness of Advertisements

Examples of some indicators of trustworthiness

Januvia Vaccine Advertisement



- Increases trustworthiness
- Decreases trustworthiness

Benefits description: Uses a large, bold font which attracts attention

Silvennoinen, et al. (2016). Appraisals of salient visual elements in web page design.

Risks description: Long sentences are difficult to comprehend

Flesch reading-ease test | Grammarly (2019). How To Use Readability Scores in Your Writing.

Negative effects disclosure: High color contrast attracts attention

Cugelman et al. (2019). Color Psychology. AlterSpark.

Negative effects disclosure: High evidentiality increases perceived credibility

Su et al. (2010). Evidentiality for text trustworthiness detection.

Negative effects disclosure: Scroll feature is a barrier to salience

Silvennoinen, et al. (2016). Appraisals of salient visual elements in web page design.

Assessing the Trustworthiness of Advertisements

List of measurable components of trustworthiness

Trustworthiness components	Short name	Measurement strategy
Readability of the content should be consistent throughout the ad to make all sections equally comprehensible	Readability gap	Flesch and fog scales
Tone should be consistently neutral throughout to make all sections equally comprehensible	Tone gap	VADER and LIWC
Color contrast of risks and benefits should be consistent across the document	Color salience gap	Google Vision API
Large quantity of the text can cause information overload at important sections	Text quantity gap	Words, syllables, sentence length, etc.
Extra steps to access important information (like scrolling and pop-ups) makes it less salient	Content visibility gap	Scroll or pop-ups
Present bias: The temporal description of risks and benefits should be similar	Present bias gap	Dictionary of tenses
Evidentiality of scientific claims should be consistent throughout the risks and benefits sections	Evidentiality gap	Dictionary of evidentiality
High color contrast of images can distract consumers from important information	Image salience	Google Vision API
Social proof given by images of people or content about people using the product can be misleading	Social proof	Amazon Rekognition, Social words dictionary
Font size and color can determine how salient a section is	Font salience	Identify font properties

Sources: Appendix II

Assessing the Trustworthiness of Advertisements

Classifying the components by elements in the Health Belief Model

The Health Belief Model is a framework that describes the behavioral drivers for consumers that lead them to take a health-related action. Advertisers often apply elements of the HBM in their ads to persuade consumers to take a desired action like purchasing a product.

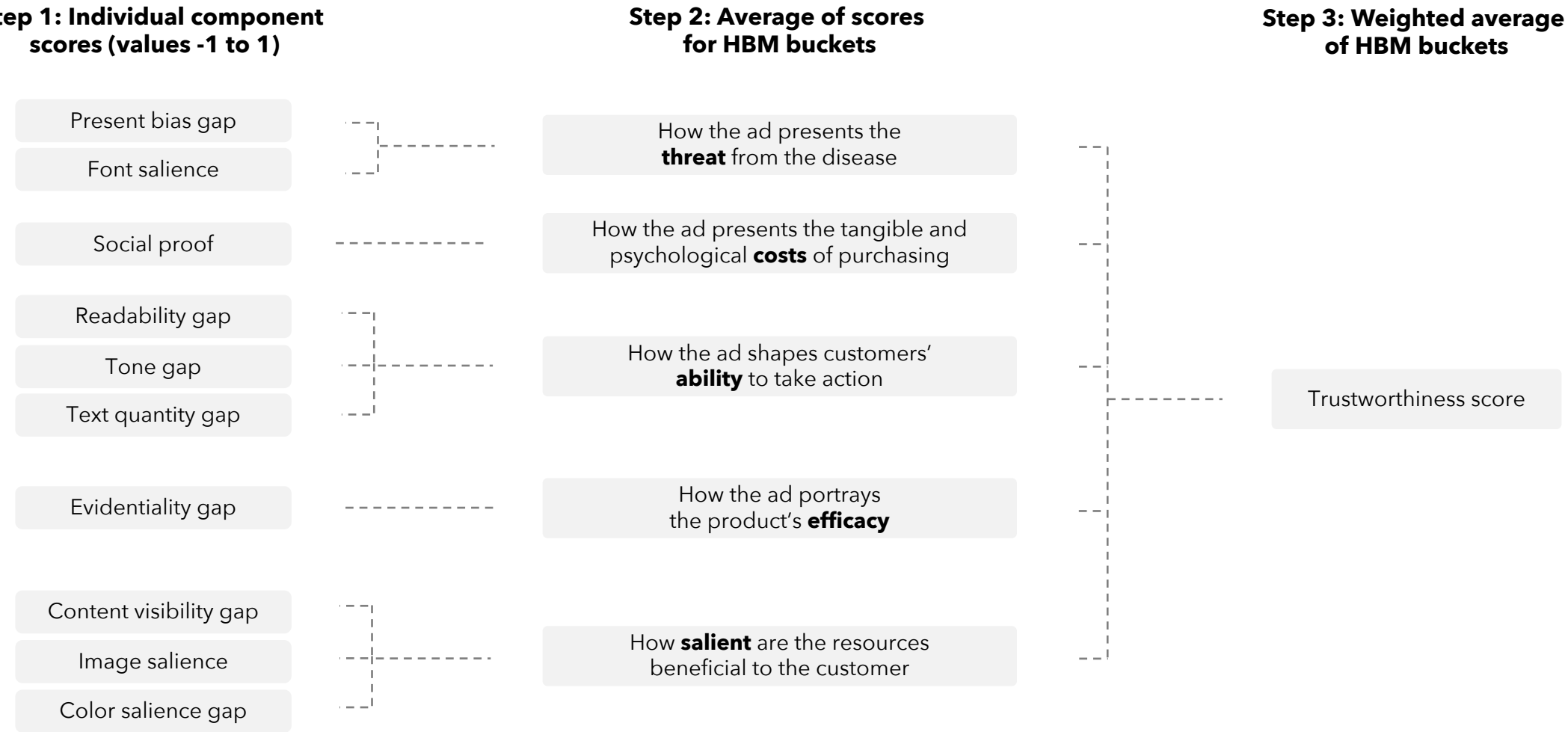
Behavioral drivers of action in the HBM	Elements of the HBM applied in advertising
Perceived **threat** of getting the disease	Highlight the severity and likelihood of the **threat** from the disease
Belief in the **efficacy** of the healthcare product	Highlight the benefits and **efficacy** of the healthcare product
Perceived **costs** of purchasing the product	Reduce the perceived tangible and psychological **costs** of purchasing the product
Confidence in one's **ability** to take action	Simplify steps to increase the **ability** of customers to take action
Cues to take desired **action**	Increase salience of the **call to action** in the advertisement

→ **Action taken to purchase a healthcare product**

Source (see Appendix III for alternative taxonomies): Champion and Skinner | The health belief model | 2008

Assessing the Trustworthiness of Advertisements

Trustworthiness score calculation methodology



Details of the HEART Framework

Details of the HEART Framework

Calculating the readability gap of risks and benefits

The readability of the text in the risks and benefits sections should be similar. A higher readability score of risks section compared to the benefits section would improve the overall trustworthiness score.

Input variables	Programming Tools	Calculations	Output
1. Text from the 'risks' section 2. Text from the 'benefits' section	1. Import *textstat* Python library 2. Calculate the following scores separately for risks and benefits: • Flesch Reading ease to measure readability • Fog Scale to measure grade-level of text	1. Calculate the difference of scores between benefits and risks for both the metrics 2. Standardize each score on a scale of -1 to 1 3. Calculate the mean of both scores	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Yeung et al. | Cognitive load and learner expertise | 1998

Details of the HEART Framework

Calculating the difference in the tone of risks and benefits

Ads with positive framing are more persuasive at encouraging customers to purchase healthcare products. A trustworthy advertisement would balance its tone across the benefits and risks sections.

Input variables	Programming Tools	Calculations	Output
1. Text from the 'risks' section 2. Text from the 'benefits' section	1. Import *vaderSentiment* 2. Calculate the following sentiment scores separately for risks and benefits • Positive tone score • Negative tone score • Neutral tone score	1. Calculate the average tone score 2. Calculate the difference of scores between benefits and risks 3. Standardize each score on a scale of -1 to 1	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Veer and Pervan | How the tone and wording of advertisements interact | International Journal of Advertising | 2008

Details of the HEART Framework

Calculating the color salience gap of risks and benefits

Sections of an ad that have a high color contrast between the font and background attract more attention. Important information in a healthcare ad should have higher contrast values to draw attention.

Input variables	Programming Tools	Calculations	Output
1. RGB value of the 'risks' section text 2. RGB value of the 'benefits' section text 3. RGB value of 'risks' section background 4. RGB value of 'benefits' section background	1. Import *vision* from *google.cloud* 2. Calculate the fractional RGB color distributions for each of the four input items	1. Calculate the average RGB value of each input item 2. Calculate the luminance of each input item 3. Calculate difference of contrast ratio of text and background for benefits and risks sections 4. Standardize on -1 to 1 scale	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source (see Appendix IV for details): Bakar et al. | Attributes for image content that attract consumers' attention to advertisements | Social and Behavioral Sciences | 2015

Details of the HEART Framework

Calculating the text quantity gap of risks and benefits

The amount of text is proportional to the information overload experienced by the reader. This overload reduces comprehension and retention of information resulting in more impulsive decisions.

Input variables	Programming Tools	Calculations	Output
1. Text from the 'risks' section 2. Text from the 'benefits' section	1. Calculate the following scores separately for risks and benefits: • Total characters • Total words • Total sentences	1. Calculate the difference of scores between benefits and risks for both the metrics 2. Standardize each score on a scale of -1 to 1 3. Calculate the mean of all scores	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Hadar et al. | When knowledge is demotivating | 2014

Details of the HEART Framework

Calculating the content visibility gap of risks and benefits

Information not immediately visible to the user because of extra steps like pop ups and scrolling can deter people from accessing useful information. Risks should be more accessible than benefits in an ad.

Input variables	Programming Tools	Calculations	Output
1. Ratio of visible to hidden content from the 'risks' section 2. Ratio of visible to hidden content from the 'benefits' section	1. Identify the visibility ratios for both risks and benefits by comparing the visible and hidden areas	1. Calculate the difference of scores between benefits and risks for both the metrics 2. Standardize the score on a scale of -1 to 1	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Pascualet al. | Impact of accessibility barriers on the mood of blind, low-vision and sighted users. Procedia Computer Science | 2014

Details of the HEART Framework

Calculating the present bias gap of risks and benefits

People overweigh immediate outcomes more than future ones. Trustworthy ads balance the tense used in risks and benefits to avoid misleading customers by marketing immediate rewards and delayed risks.

Input variables	Programming Tools	Calculations	Output
1. Text from the 'risks' section 2. Text from the 'benefits' section	1. Import *nltk* library 2. Determine the score for each tense in the text 3. Get total score for each input using the formula: • Past * 1 + Present * 0 + Future * -1	1. Calculate the difference of tense scores between benefits and risks for both the metrics 2. Standardize the score on a scale of -1 to 1	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Hardisty et al. | Good or bad, we want it now | 2013

Details of the HEART Framework

Calculating the evidentiality gap of risks and benefits

The evidentiality of a statement, i.e., the degree of evidence provided, can convince people of its validity. Statements with high evidentiality are more persuasive and should be used consistently across the ad.

Input variables

1. Text from the 'risks' section
2. Text from the 'benefits' section

Programming Tools

1. A dictionary of words scored on varying levels of evidentiality complied by Su et al.

Calculations

1. Score the inputs using the evidentiality dictionary
2. Standardize each score on a scale of -1 to 1

Output

1. Values (-1 to 1)
   - -1 (least trustworthy)
   - 1 (most trustworthy)

Source (see Appendix V for details): Su et al. | Evidentiality for text trustworthiness detection. Workshop on NLP and Linguistics | 2010

Details of the HEART Framework

Calculating the salience of images relative to text

Studies using eye-tracking software have shown that a high color contrast and brightness of images in an advertisement can distract users from the actual content which may be useful to them.

Input variables	Programming Tools	Calculations	Output
1. Image used in the advertisement 2. Image of the textual part of the ad	1. Import *vision* from *google.cloud* 2. Calculate the fractional RGB color distributions for each of the two input items	1. Calculate the average RGB value of the inputs 2. Calculate the luminance of the inputs 3. Find the difference of the luminance 4. Standardize the score on a scale of -1 to 1	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Beymer et al. | An eye tracking study of how pictures influence online reading. In IFIP Conference on Human-Computer Interaction | 2007

Details of the HEART Framework

Calculating the social proof score

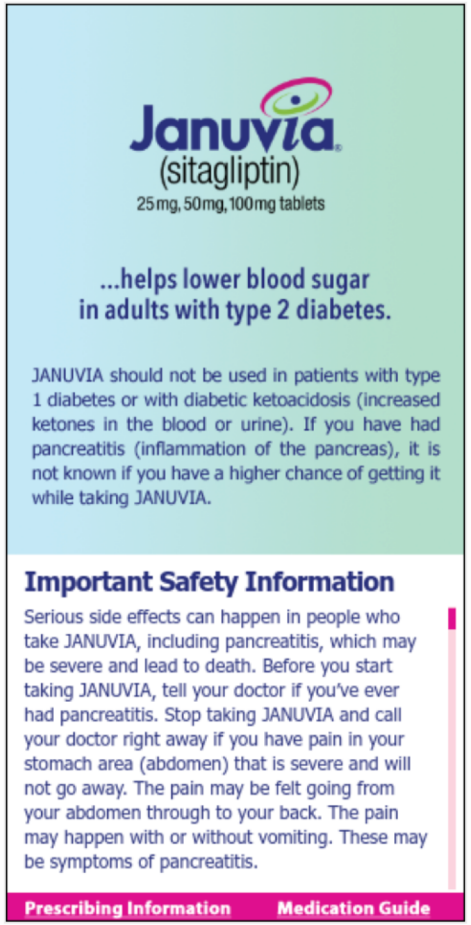
Providing social proof in the form of images of people or in the text is a tactic used by advertisers to persuade customers of the quality of a product but does not equate to actual quality of the product.

Input variables	Programming Tools	Calculations	Output
1. Image used in the advertisement	1. Import *vision* from *google.cloud* 2. Identify the presence of a person or people in the image	1. Give a score of 1 if the image of a person is absent and -1 if it's present	1. Values (-1 to 1) • -1 (least trustworthy) • 1 (most trustworthy)

Source: Das et al. | Increasing security sensitivity with social proof: A large-scale experimental confirmation | 2014

Details of the HEART Framework

Demo: Merck's Januvia advertisement



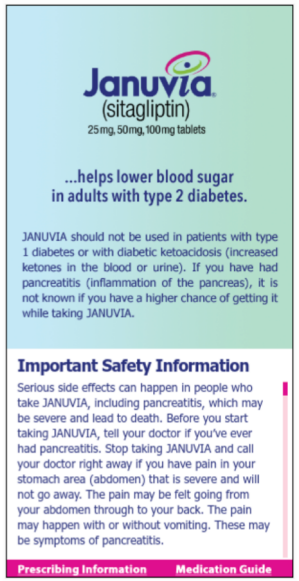
Building Block	Value
Benefits Text	Helps lower blood sugar in adults with type 2 diabetes.
Risks Text	JANUVIA should not be used in patients with type 1 diabetes or with diabetic ketoacidosis (increased ketones in the blood urine). If you have had pancreatic (inflammation of the pancreas), it is not known if you have a higher chance of getting it while taking JANUVIA.
Benefits Text RGB	rgb(53, 61, 126)
Risks Text RGB	rgb(53, 61, 126)
Benefits Background RGB	rgb(193, 225, 223)
Risks Background RGB	rgb(255, 255, 255)
Benefits Visibility Ratio	100%
Risks Visibility Ratio	15%
Image RGB	False
Rest RGB	rgb(194, 211, 219)
Image Person Exists	False

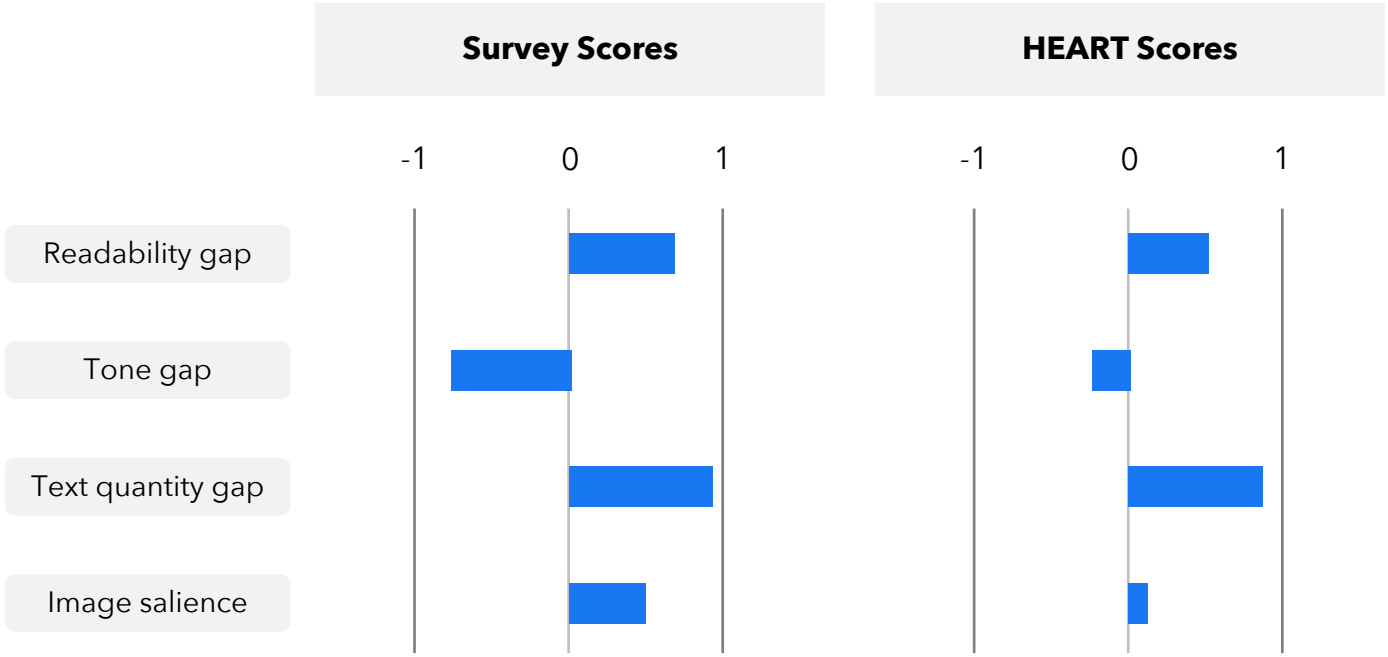
Python program screenshots in Appendix VI

Details of the HEART Framework

Testing the internal validity of the framework

Measure the components of the framework through a survey by showing advertisements to participants and then match those survey scores with the corresponding values from the framework.





Details of the HEART Framework

Testing the external validity of the framework

Identify specific actions taken by PRC members on pharmaceutical advertisements. Find a relationship between PRC's action and scores calculated by the HEART framework.

	PRC Action	HEART Scores
	Approved	70%
	Rejected	35%
	Approved	65%

Expanding the HEART Framework

Expanding the HEART Framework

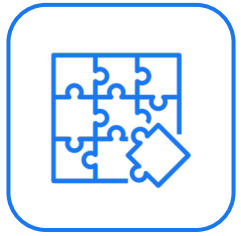
Additional dimensions to the framework

Analyze Effect of Typography

Font style, size, color, weight, etc. affect readers' retention and comprehension.

Dressler (2019)

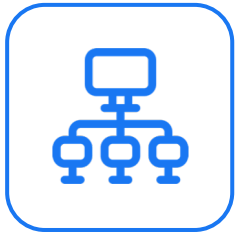
Contextualize Marketing



Trust is built through a combination of factors in the customer journey. Contextualize advertising with these factors.

Brenner (2017)

Create Alternative Taxonomies



Categorize the components using a different system like the transtheoretical model of behavior change top suit the needs of different users.

Prochaska et al. (2009)

Expanding the HEART Framework

Applications in the healthcare industry

Reach Minority Groups

Understand what types of marketing content minority groups are more receptive to and score ads based on that.

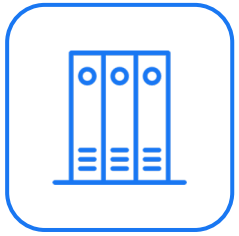
Moore et al. (1996)

Build Marketing Profiles

Categorize healthcare firms based on patterns in marketing practices.

Tomlinson (2006)

Automate Administrative Tasks



Use text analytics to automate other administrative tasks of regulatory agencies.

Lexalytics (2019)

Expanding the HEART Framework

Applications in other industries

Assess Financial Reports

Assess financial reports based on how balanced and transparent they are.

Lewis et al. (2019)

Analyze Media Content

Find patterns in video ads (subtitle data and video) and see how they affect responses of audiences.

Itamar et al. (2008)

Assess Legal Documents

Analyze legal transcripts to see if there are patterns in how judicial proceedings affect legal outcomes.

Dale (2018)

Expanding the HEART Framework

What's next?

Build a product for regulatory agencies like the FDA and PRC teams which they can use to provide an 'advertising report card' for companies selling products like the COVID-19 vaccine. Pharmaceutical firms can show customers that they are trustworthy and use that to position themselves as trustworthy.

1. **Add components**: Identify other components using behavior science, design principles, linguistics, etc.
2. **Find other taxonomies**: Find different ways to bucket the components relevant for different users.
3. **Test**: Test for internal and external validity.
4. **Build**: Develop a minimal viable product.

Appendix

Appendix I

FDA requirements

What must product claim ads tell you?

- At least one approved use for the drug
- The generic name of the drug
- All the risks of using the drug
  - Under certain circumstances, ads can give only the most important risks
  - For more detail, see brief summary and adequate provision

Does the law say anything about the design of ads for prescription drugs?

- Yes. The layout of an ad — the way information is presented — can affect whether an ad meets the fair balance requirement. For example, ads must present side effect information in a manner similar to that used for the benefit information. Various ways of presenting information that can affect fair balance include type size, bulleting, amount of white space, and headlines.

What are ads not required to tell you?

- Cost
- If there is a generic version of the drug (a drug with the same active ingredient that might be cheaper)
- If there is a similar drug with fewer or different risks that can treat the condition
- If changes in your behavior could help your condition (such as diet and exercise)
  - Sometimes this information is required. It depends on the prescribing information for the particular drug
- How many people have the condition the drug treats
- How the drug works (its "mechanism of action")
- How quickly the drug works
  - However, if the ad claims that the drug works quickly, the ad must explain what "quickly" means
- How many people who take the drug will be helped by it

Appendix II

Sources for components of the HEART framework

Trustworthiness components	Short name	Measurement strategy	
Readability of the content should be consistent throughout the ad to make all sections equally comprehensible	Readability gap	Flesch and fog scales	Yeung et al. (1998)
Tone should be consistently neutral throughout to make all sections equally comprehensible	Tone gap	VADER and LIWC	Donovan et al. (2000)
Color contrast of risks and benefits should be consistent across the document	Color salience gap	Google Vision API	Smith (2016)
Large quantity of the text can cause information overload at important sections	Text quantity gap	Words, syllables, sentence length, etc.	Hadar et al. (2014)
Extra steps to access important information (like scrolling and pop-ups) makes it less salient	Content visibility gap	Scroll or pop-ups	Sullivan et al. (2000)
Present bias: The temporal description of risks and benefits should be similar	Present bias gap	Dictionary of tenses	Hardisty et al. (2013)
Evidentiality of scientific claims should be consistent throughout the risks and benefits sections	Evidentiality gap	Dictionary of evidentiality	Su et al. (2010)
High color contrast of images can distract consumers from important information	Image salience	Google Vision API	Smith (2016)
Social proof given by images of people or content about people using the product can be misleading	Social proof	Amazon Rekognition, Social words dictionary	Samuel (2020)
Font size and color can determine how salient a section is	Font salience	Identify font properties	Hoffmeister (2016)

Appendix III

Alternative taxonomies

The components can be classified into easy-to-understand buckets and aggregate trustworthiness scores can be calculated for each bucket. Some possible categorizations can be as follows:

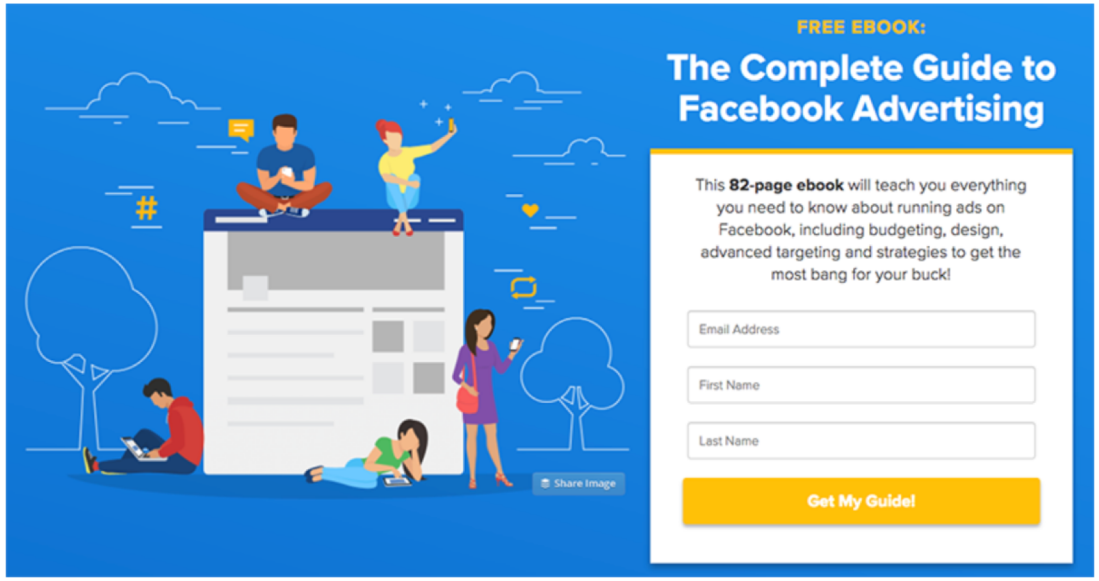
Which cognitive function does the ad appeal to	Which legal requirements are satisfied	Design element of the advertisement	Behavioral effect on the consumer	Building block of the advertisement
Emotional appeal (System 1)	Adheres to the Fair Balance law	Aesthetics	Information overload	Benefits sections
Rational appeal (System 2)	Contains the FDA-mandated sections	Text	Salience	Risks sections
Other	Other	Numbers	Tone	Other
			Present bias	

Appendix IV

Color contrast

Source: Smith, C. (2016). The conversion code: Capture internet leads, Create quality appointments, close more sales.

Choosing colors across from each other attracts the eye to your call-to-action buttons, headline, images, benefit list, etc.



Notice that the accent color (orange) effectively draws the eye to the call to action.

Appendix V

Evidentiality

Source: Su et al. (2010). Evidentiality for text trustworthiness detection. Workshop on NLP and Linguistics.

	Absolute	High	Moderate	Low
Attributive/modal adverb	*certainly, sure, of course, definitely, absolutely, undoubtedly*	*clearly, obviously, apparently, really, always*	*Seemingly, probably*	*maybe, personally, perhaps, possibly, presumably*
Lexical verb	*report, certain*	*believe, see*	*seem, think, sound*	*doubt, wish, wonder, infer, assume, forecast, fell, heard*
Auxiliary verb		*must*	*ought, should, would, could, can*	*may, might*
Epistemic adjective	*definite*		*possible, likely, unlikely, probable, positive, potential*	*not sure, doubtful*

Different types of evidentialities and their levels

Appendix VI

Python source code (I/V)

```
# ---------------------------------------------------------------------- #
# ---------------- ADVERTISEMENT PROPERTIES INPUT ---------------- #
# ---------------------------------------------------------------------- #

# Text
vBenefitsText = "Helps lower blood sugar in adults with type 2 diabetes."
vRisksText = "JANUVIA should not be used in patients with type 1 " \
             "diabetes or with diabetic ketoacidosis (increased ketones " \
             "in the blood urine). If you have had pancreatic (inflammation " \
             "of the pancreas), it is not known if you have a higher chance " \
             "of getting it while taking JANUVIA."

# Colors
vRGBBenefitsText = (53, 61, 126)
vRGBRisksText = (53, 61, 126)
vRGBBenefitsBackground = (193, 225, 223)
vRGBRisksBackground = (255, 255, 255)

# Visibility
vBenefitsVisibilityRatio = 1
vRisksVisibilityRatio = 0.15

# Average colors of image used and the remaining part from google.vision applet
vRGBImage = False
vRGBRest = (194, 211, 219)

# Social proof from images
vImagePersonExists = False
```

Appendix VI

Python source code (II/V)

```
# ------------------------------------------------------------ #
# ------------------- BACK-END FUNCTIONS --------------------- #
# ------------------------------------------------------------ #

# Import libraries
import textstat  # Readability score
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer  # Tone
import re  # Regex
from nltk import word_tokenize, pos_tag, punkt # tenses

# Calculate final heart score
def CalculateHEARTScore(pRisksText,
                        pBenefitsText,
                        pRGBBenefitsText,
                        pRGBRisksText,
                        pRGBBenefitsBackground,
                        pRGBRisksBackground,
                        pBenefitsVisibilityRatio,
                        pRisksVisibilityRatio,
                        pRGBImage,
                        pRGBRest,
                        pImagePersonExists):

    print("\n")
    # Component I
    vReadabilityGap = MeasureReadabilityGap(pBenefitsText, pRisksText)
    print("Readability gap score (-1 to 1): " + str(vReadabilityGap))

    # Component II
    vColorSalienceGap = MeasureColorSalienceGap(pRGBBenefitsText, pRGBRisksText, pRGBBenefitsBackground, pRGBRisksBackground)
    print("Color salience gap score (-1 to 1): " + str(vColorSalienceGap))

    # Component III
    vToneGap = MeasureToneGap(pBenefitsText, pRisksText)
    print("Tone gap score (-1 to 1): " + str(vToneGap))

    # Component IV
    vTextQuantityGap = MeasureTextQuantityGap(pBenefitsText, pRisksText)
    print("Text quantity gap score (-1 to 1): " + str(vTextQuantityGap))
```

```
    # Component V
    vContentVisibilityGap = MeasureContentVisibilityGap(pBenefitsVisibilityRatio, pRisksVisibilityRatio)
    print("Content visibility gap score (-1 to 1): " + str(vContentVisibilityGap))

    # Component VI
    vImageSalienceGap = MeasureImageSalienceGap(pRGBImage, pRGBRest)
    print("Image salience gap score (-1 to 1): " + str(vImageSalienceGap))

    # Component VII
    vImageSocialProof = MeasureImageSocialProof(pImagePersonExists)
    print("Image social proof score (-1 to 1): " + str(vImageSocialProof))

    # Component VII
    vPresentBiasGap = MeasurePresentBiasGap(pBenefitsText, pRisksText)
    print("Present bias gap score (-1 to 1): " + str(vPresentBiasGap))

    # Final score
    vHEARTScore = round((vReadabilityGap +
                         vColorSalienceGap +
                         vToneGap +
                         vTextQuantityGap +
                         vContentVisibilityGap +
                         vImageSalienceGap +
                         vImageSocialProof +
                         vPresentBiasGap) /8, 3)
    print("\n")
    print("The HEART score (-1 to 1): " + str(vHEARTScore))

    print("The HEART score (percent): " + str((vHEARTScore + 1/0.02)) + "%")
```

Appendix VI

Python source code (III/V)

```
# Measure the readability gap
def MeasureReadabilityGap(pBenefitsText, pRisksText):

    # Calculating the Flesch Reading-ease score gap
    vDifferenceFlesch = textstat.flesch_reading_ease(pBenefitsText) - textstat.flesch_reading_ease(pRisksText)

    # Standardize
    vDifferenceFleschStandard = vDifferenceFlesch/10

    # Calculating the Fog Scale score
    vDifferenceFog = (textstat.gunning_fog(pBenefitsText) if
                      textstat.gunning_fog(pBenefitsText) <= 12 else 12) - \
                     (textstat.gunning_fog(pRisksText) if textstat.gunning_fog(pRisksText) <= 12 else 12)

    # Standardize
    vDifferenceFogStandard = vDifferenceFog/12

    return round((vDifferenceFleschStandard + vDifferenceFogStandard)/2, 3)

# Measure the tone gap
def MeasureToneGap(pBenefitsText, pRisksText):

    # Sentiment analyzer tools
    vVaderToneAnalyser = SentimentIntensityAnalyzer()

    def SentimentAnalyzerScores(pDialogue):
        vScore = vVaderToneAnalyser.polarity_scores(pDialogue)
        return dict(vScore)

    vWeights = [1, 0, 1, 0]

    def GetAggregateTone(pText, pWeights):
        vAggregateTone = 0
        for x, y in zip(SentimentAnalyzerScores(pText).values(), pWeights):
            vAggregateTone = vAggregateTone + (x * y)
        return vAggregateTone

    return round(GetAggregateTone(pBenefitsText, vWeights) - GetAggregateTone(pRisksText, vWeights), 3)
```

```
# Measure the color salience gap
def MeasureColorSalienceGap(pRGBBenefitsText, pRGBRisksText, pRGBBenefitsBackground, pRGBRisksBackground):

    # Relative luminance of benefits
    if (CalculateLuminance(pRGBBenefitsText) <= CalculateLuminance(pRGBBenefitsBackground)):
        L1 = CalculateLuminance(pRGBBenefitsText)
        L2 = CalculateLuminance(pRGBBenefitsBackground)
    else:
        L1 = CalculateLuminance(pRGBBenefitsBackground)
        L2 = CalculateLuminance(pRGBBenefitsText)

    vRelLuminanceBenefits = (L1 + 0.05)/(L2 + 0.05)

    # Relative luminance of risks
    if (CalculateLuminance(pRGBRisksText) <= CalculateLuminance(pRGBRisksBackground)):
        L1 = CalculateLuminance(pRGBRisksText)
        L2 = CalculateLuminance(pRGBRisksBackground)
    else:
        L1 = CalculateLuminance(pRGBRisksBackground)
        L2 = CalculateLuminance(pRGBRisksText)

    vRelLuminanceRisks = (L1 + 0.05)/(L2 + 0.05)

    return round((vRelLuminanceBenefits - vRelLuminanceRisks)/40, 3)

# Measure the text quantity gap
def MeasureTextQuantityGap(pBenefitsText, pRisksText):

    # Functions
    def TextLength(pText):
        return len(pText)
    def TextWords(pText):
        return len(pText.split())
    def TextSentences(pText):
        return len(re.split(r'[.!?]+', pText))
    def StandardDifference(Function, pText1, pText2):
        return (Function(pText1) - Function(pText2))/max(Function(pText1), Function(pText2))

    return round((StandardDifference(TextLength, pBenefitsText, pRisksText) +
                  StandardDifference(TextWords, pBenefitsText, pRisksText) +
                  StandardDifference(TextSentences, pBenefitsText, pRisksText))/3, 3)
```

Appendix VI

Python source code (IV/V)

```python
# Measure the visibility ratio gap
def MeasureContentVisibilityGap(pBenefitsVisibilityRatio, pRisksVisibilityRatio):
    return round(pBenefitsVisibilityRatio - pRisksVisibilityRatio, 3)

# Measure the color salience gap
def MeasureImageSalienceGap(pRGBImage, pRGBRest):

    if pRGBImage == False:
        return 1
    elif pRGBRest == False:
        return -1
    else:
        # Luminance of image
        L1 = CalculateLuminance(pRGBImage)

        # Luminance of the rest
        L2 = CalculateLuminance(pRGBRest)

        return round((L1 - L2)/255, 2)

# Function to measure social proof
def MeasureImageSocialProof(pImagePersonExists):
    if(pImagePersonExists == True):
        return -1
    else:
        return 1
```

```python
# Function to measure present bias
def MeasurePresentBiasGap(pBenefitsText, pRisksText):

    # Tokenized text
    vTokenizedBenefitsText = word_tokenize(pBenefitsText)
    vTokenizedRisksText = word_tokenize(pRisksText)

    # Tagged text
    vTaggedBenefitsText = pos_tag(vTokenizedBenefitsText)
    vTaggedRisksText = pos_tag(vTokenizedRisksText)

    # Tenses for benefits
    vTenseBenefitsText = {}
    vTenseBenefitsText["Future"] = len([vWord for vWord in vTaggedBenefitsText if vWord[1] == "MD"])
    vTenseBenefitsText["Present"] = len([vWord for vWord in vTaggedBenefitsText if vWord[1] in ["VBP", "VBZ","VBG"]])
    vTenseBenefitsText["Past"] = len([vWord for vWord in vTaggedBenefitsText if vWord[1] in ["VBD", "VBN"]])
    vTenseBenefitsText["Total"] = len(pBenefitsText.split())
    vTaggedBenefitsTextScore = (int(vTenseBenefitsText["Past"]) * -1 +
                                int(vTenseBenefitsText["Present"]) * 0 +
                                int(vTenseBenefitsText["Future"]) * 1)/int(vTenseBenefitsText["Total"])

    # Tenses for risks
    vTenseRisksText = {}
    vTenseRisksText["Future"] = len([vWord for vWord in vTaggedRisksText if vWord[1] == "MD"])
    vTenseRisksText["Present"] = len([vWord for vWord in vTaggedRisksText if vWord[1] in ["VBP", "VBZ","VBG"]])
    vTenseRisksText["Past"] = len([vWord for vWord in vTaggedRisksText if vWord[1] in ["VBD", "VBN"]])
    vTenseRisksText["Total"] = len(pRisksText.split())
    vDominantTenseRisksTextScore = (int(vTenseRisksText["Past"]) * 1 +
                                    int(vTenseRisksText["Present"]) * 0 +
                                    int(vTenseRisksText["Future"]) * -1)/int(vTenseRisksText["Total"])

    vPresentBiasScore = round((vTaggedBenefitsTextScore + vDominantTenseRisksTextScore)/2, 2)

    return vPresentBiasScore
```

Appendix VI

Python source code (V/V)

```
# Function to calculate luminescence
def CalculateLuminance(pRGBTuple):
    R, G, B = pRGBTuple
    return 0.2126 * R + 0.7152 * G + 0.0722 * B

# ------------------------------------------------------------ #
# ------------------------ HEART SCORE ----------------------- #
# ------------------------------------------------------------ #

# Calculating the HEART score
CalculateHEARTScore(vBenefitsText,
                    vRisksText,
                    vRGBBenefitsText,
                    vRGBRisksText,
                    vRGBBenefitsBackground,
                    vRGBRisksBackground,
                    vBenefitsVisibilityRatio,
                    vRisksVisibilityRatio,
                    vRGBImage,
                    vRGBRest,
                    vImagePersonExists)
```

Appendix VI

Python program output

```
Readability gap score (-1 to 1): -0.916
Color salience gap score (-1 to 1): 0.001
Tone gap score (-1 to 1): -0.362
Text quantity gap score (-1 to 1): 0.637
Content visibility gap score (-1 to 1): 0.85
Image salience gap score (-1 to 1): 1
Image social proof score (-1 to 1): 1
Present bias gap score (-1 to 1): -0.03


The HEART score (-1 to 1): 0.273
The HEART score (percent): 50.273%
```