AMERICAN UNIVERSITY OF BEIRUT

SUPPORTING AFFORDANCE DETECTION AND EXTRACTION IN ONLINE PRODUCT REVIEWS

by
REMI MAHER NASSAR

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Science in Business Analytics
of the Olayan School of business
at the American University of Beirut

Beirut, Lebanon
February 2020

AMERICAN UNIVERSITY OF BEIRUT

SUPPORTING AFFORDANCE DETECTION AND EXTRACTION FROM ONLINE PRODUCT REVIEWS

by
REMI MAHER NASSAR

Approved by:

Dr.Fouad Zablith, Assistant professor Olayan School of Business	Advisor
Dr.Bijan Azad, Associate professor Olayan School of Business	Member of Committee

Date of thesis/dissertation defense: February 13, 2020

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: Nassar Rami Maher
Last First Middle

⊘ Master's Thesis ◯ Master's Project ◯ Doctoral Dissertation

☒ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes
after : **One ---- year from the date of submission of my thesis, dissertation, or project.**
Two ---- years from the date of submission of my thesis, dissertation, or project.
Three ---- years from the date of submission of my thesis, dissertation, or project.

02/20/2020

Signature Date

AN ABSTRACT OF THE THESIS OF

Remi Maher Nassar for Master of Science in Business analytics
Major: Business analytics

Title: Supporting affordance detection and extraction from online product reviews

Affordances have proved very useful in conceptualizing features of products and the action potential (or action possibility) of those features to users: a pen provides writing functionality, and its action potential is the write-ability—an *affordance*. This simple formulation is powerful, and yet it is quite deceiving because an affordance is very hard to pin down in practice. That is, beyond simple objects like pens, how do we establish and articulate affordances for novel items like a smartphone? In fact, most scholarly works presume an affordance exists and spend little time justifying its existence. Therefore, rather than presuming their existence, it would be worthwhile to dedicate research effort to discovering affordances empirically in a rigorous and theoretically grounded manner.

A fairly underexplored source and a potential mine of "naturally" occurring affordances can be the text of online product reviews. Indeed, this thesis proposes a framework to detect and extract affordances from the text of online product reviews. We employed an online tool that aggregates product reviews from Amazon.com. We then used the dataset of product reviews to annotate the potentially occurring associated affordances. Then three analysts used a tool to assign affordances to the extracted text fragments. We then analyzed the identified affordances as well as the inter-rater agreements associated with these affordances. Subsequently, employing pattern recognition algorithms and techniques, we generated a dataset and used it to identify the potential existence of pseudo-grammatical and part-of-speech patterns in text. We then performed frequency count analysis and visual analytics techniques to highlight dominant patterns that stand out. First, the results point to a useful, albeit preliminary, methodology to extract and identify affordances in text data. Second, the results show the potential existence of distinctive pseudo-grammatical and part-of-speech patterns that can occur as a basis for identifying and extracting affordances via text of online product reviews.

In summary, this thesis contributes to the extant affordance detection and extraction literature in two ways. First, our custom-developed publicly accessible application software for affordance annotation with a robust user interface/user experience is available to others for use and replication. Second, our proposed framework is a starting point for building a universal, robust rule set that will help in the identification and extraction of affordances from a variety of text sources beyond online reviews.

CONTENTS

ILLUSTRATIONS

CHAPTER I

INTRODUCTION

A. Introduction

The way we deal and interact with objects in our daily lives is different between people and can even change for the same individual, depending on the context. That concept is what was coined as an affordance by J.J. Gibson. An affordance is, in the simplest term, the immediate perceived use of an object or device by a specific user in a given context (Gibson, 1977).

To further explain this concept, we shall take a pen as an example to showcase the use case and concept of affordances. For the everyday man or woman, a pen is a tool that allows them to write or, in other words, it provides *writability*. Now assume that the same pen is given to the fictional character of James Bond, in his hands, such a tool would allow for self-defense or can be used as a lethal weapon—e.g., *killability*. This example, though extreme, hopefully, helps clarify the concept.

Although the concept of affordances grew originally from psychology and, this has not stopped it from gaining traction in many other disciplines (product and website design, software and video game design, cognitive science research, etc.). The study and application of affordances have been on the rise, especially in product and interface design (Pucillo & Cascini, 2013). Due to its popularity and widespread diffusion in product design and technology-based research in many disciplines, being able to assign affordances to specific features of products or having a framework to do so can of significant benefit in several domains of inquiry.

Given the potential for affordances to be deployed in the service of theories and applications within the realm of design, several preliminary attempts at inferring them are grounded in image processing (e.g., Maier& Fadel, 2007), as well as few in product reviews through text (e.g., Chou & Shu, 2014).

Therefore, inferring affordances of product features from product reviews that may lead to a general method of doing affordance extraction is a worthy and important goal. Toward that end, this thesis proposes a framework that would that uses textual patterns and pseudo-grammatical rules that would be relevant to affordance detection using online product reviews.

B. Motivation

As we said earlier, the concept of affordances has been adopted in the field of product design—a product can be a tangible item such as a phone or an intangible one such as a webpage or a video game. Because of this widespread use, there appears to be a growing need to detect affordances. Because this would be valuable as it could help identify user perception of the product features in question in a highly data-based and evidence-based manner. This type of "naturally-occurring" affordance detection is superior to other methods because it would take account of how users may utilize the product in user-contexts, sometimes even unintended by the designers. Using their knowledge of how the end-user perceives the product, designers can double down on certain features to create a superior version of the product that caters to the affordances viewed by users.

One potential, openly accessible, user-generated, source of data that can help us extract naturally-occurring affordances comes in the form of online user product reviews. In particular, users are usually verbose and will not hesitate to speak their minds when reviewing a product online and are quite vocal regarding their experiences, and as such online product reviews would serve as a great starting point for our research.

C. Goal and scope

Knowing the importance and value of affordances in product design and having an idea of the type of dataset that is useful for that process, i.e., online product reviews, for practical reasons, we narrow down the scope to reviews of Nokia phones as a use case. The reviews chosen would include different models of this manufacturer's phone, whereby there is an expectation for users to identify a diverse set of affordances.

1. Research question

As we have already mentioned, current efforts for affordance extraction include an image processing approach (Pucillo & Cascini, 2013) or are more rooted in manual and statistical methods (Chou & Shu, 2014). As such, the thesis aims to answer the following research question: "What types of textual patterns, including part of speech acts and pseudo-grammatical rules, are relevant to the detection of affordances in text?"

2. General methodology

To achieve the goal set above and answer the question, we shall proceed by building an online tool that would help in the semi-automated processing of affordances from online reviews. Several analysts will be utilizing the app along with the use of an inter-rater reliability[1] score to obtain a distinctive set of affordances. The development of the online tool was necessary since we could not find any usable affordance dataset/annotated dataset online, to carry out our research. Using these affordance data collected from our analysts, we would build an algorithm that utilizes part-of-speech tag pattern recognition, to try and identify affordances from texts of reviews and propose a general pseudo-grammar-like structure for future work.

3. Research contribution

The main contribution lies in the identification of key patterns, in the form of part of speech tags, in text that may help denote affordances by providing a set of structures and grammars that embody affordances in text. Additionally, this thesis will provide an application software that will allow a relatively smooth processing of affordances from text, which would help future research to work with similar datasets. Finally, we enumerate other techniques that were used but were not deemed to have produced useful results.

[1] More on inter-rater reliability: https://en.wikipedia.org/wiki/Inter-rater_reliability

D. Structure of the document

With our goal set, we shall first explore the affordance concept in more depth. Then, we will look in more detail at works that would tackle the affordance detection process on multiple fronts such as on imagery as well text to gain a further understanding of the existing research on affordance detection.

Subsequently, we will explore the multiple review datasets before narrowing down our work to a specific set and using it as a base to query the actual reviews of interest.

With the review dataset on hand, we will then deploy the online application software built to have analysts annotate affordance and explain its design and functionality in detail.

Then, we shall discuss the distinctive steps taken to process the data obtained from analysts. The first is having an analyst agreement on affordance to provide a more solid basis since there is potential for individual idiosyncratic identification of affordances. Following that step, we will proceed by additional steps of data exploration and data extraction. Using the new and base fields, we would develop and run a pattern recognition algorithm on the part-of-speech tag aspect of the dataset to identify a set of pseudo-grammatical structures that affordances potentially express in textual form. Finally, we shall discuss the results of the overall experiment.

CHAPTER II

BACKGROUND AND RELATED WORK

A. What is an affordance

1. Origins of the term

The term affordance, created and established by J.J. Gibson in 1979, is a term that sees its use in many fields, having slightly varied definitions and interpretations based on the context it is used in (Gibson, 1979). The loose interpretation is primarily linked to the fact that Gibson himself, in his own works, never really established a final and definitive definition of the term. Hence why the concept's meaning is often up for debate within communities.

We will study Gibson's original definition of the term, as explained by Keith S. Jones in his article titled "What is an affordance" (Jones, 2010). In his article, he explains that the affordance is an ecological concept coined by JJ Gibson to study the interaction of an animal with the environment and objects. Gibson primarily studied the subject of “objects and events have inherent meaning, which is detected and exploited by the animal without mental calculation, that is, a direct-perception view” when defining affordances (Jones, 2010).

Gibson evolved the concept as he published books but never finalized the meaning leaving the exact definition to be ambiguous and debatable. Starting out as the concept of valence, which assumes objects have inherent meaning and could affect behavior. The idea after involved the environment and how it could control behavior. Moreover, after transitioned to how “animals seem to have an immediate awareness of the possibilities afforded by environmental objects." And soon evolved to “It implies the complementarity of the animal and the environment” (Jones, 2010).

2. Application of the term

Moving away from the more subjective and philosophical roots of the term into more practical and tangible approaches, the paper by Maier and Fadel titled identifying affordances elaborates on four methods to identify affordances noted as pre-determination, direct experimentation, indirect experimentation, and automated identification (Maier & Fadel, 2007). In the end, they concluded that affordances are instinctive and straightforward to pick up on elaborating that an infant should be able to easily identify what they and as such designers should be able to identify and manipulate the affordances of their products, further cementing the original notion proposed by Gibson (Maier & Fadel, 2007).

Affordances have come to play a role in product design, growing its own subfield of study. In one such study by Pucillo and Cascini, the authors reflect on the importance of the user experience and how it can derive pleasure or frustration based on the interaction with the product. Thus, its affordances should be designed with such ideas in mind (Pucillo & Cascini, 2013).

B. Affordance detection efforts

1. Image-based machine learning approaches

When looking for approaches regarding affordances and machine learning, convolutional neural networks, and automation in several general results would often lead to a more image-based take on the topic. Such approaches would use images of objects and would have a machine learning algorithm try to deduce affordances based on the available data. Take, for example, the work done by Anh Nguyen, Dimitrios Kanoulas, Darwin G. Caldwell, and Nikos G. Tsagarakis, where they employ a convolutional neural network to detect the affordance in particular objects. The intent behind the CNN is to feed the detected affordance to a robot which would use the result to figure out how to grip the project in question (Nguyen, Kanoulas, Caldwell, and Tsagarakis,2017).

2. Text-Based Affordance detection

A unique approach was taken regarding the study of affordances by Shu and Chou from the University of Toronto in Canada. In their paper, they use the Canadian tire corpus as their subject for the study. They were able to deduct the main recurring affordances specific to the product at hand through the use of keywords from those affordances to perform K-means clustering. In addition, they ran a manual inspection of reviews. They managed to isolate phrases with a higher chance of leading to an affordance, such as “as opposed to,” “can actually,” “doubles as” and “than usual.” With these results on hand, they speculate that a more machine learning based approach should be able to identify more complex patterns (Chou & Shu, 2014). They note that the absence of an annotated dataset for training algorithms was an obstacle in their work as they had to rely on manual methods.

3. Pattern recognition in text

The use of various machine learning algorithms has been employed before to identify action verbs in text. This approach was made by Mark Steedman, where he elaborates that natural language and intended actions are related systems in psychological theory. As such, it should be feasible to extract action verbs from text using autonomous methods (Steedman, 2002). He then proceeds to identify affordances as an action while taking some caution doing so, given the vague and differently interpretable nature of the words. As such, his approaches will act as inspiration and a template for our work.

C. The gap and the process

1. The gap

As we have seen, the attempts to detect affordances have taken many forms, from manual work identifying affordances in text to automating an image recognition system to highlight the potential affordances within a given object. Besides, we know that understanding and finding affordances for products is an essential task as it has become deeply rooted in design philosophies of the modern age. Also, it should be noted that there is no presence of an online dataset of affordances. Furthermore, given that the current generation has a growing online community that would not hold back on voicing its opinions regarding specific products, we shall proceed to undertake the task of supporting affordance detection in online text-based reviews. Our goal behind this is setting the basic rules for a pseudo-grammar that would help distinguish affordances. This approach would serve a building block in the modeling of text affordance, which would help in closing the gap.

2. Thesis process

In order to fill the gap mentioned above, we will undertake a set of tasks to obtain the final results. Figure 2.1 shows the process map of the work done for the thesis and sets the structure of this document form this point onwards. The first two blocks of the process fall under data collection and will be elaborated on in the next chapter. The third block covers the pre-processing and cleaning of the data. The pattern analysis algorithm or the fourth step of the process will be covered in chapter 4, and chapter 5 will cover the results of the entire process.

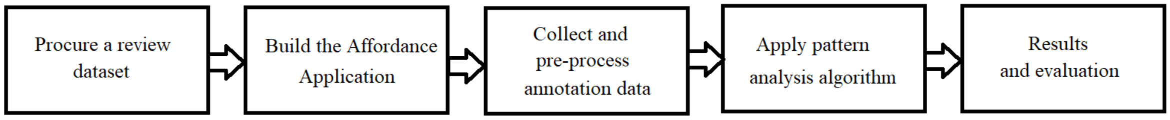


Figure 2.1 thesis process map

CHAPTER III

DATA COLLECTION

METHODOLOGY

To be able to analyze data regarding affordances, we must first obtain that data. As we already highlighted, no such data existed online, and as such, we undertook the task of creating our own affordance dataset. Figure 3.1 shown below portrays a high-level view of the process taken.

We would first start by identifying an appropriate review dataset that houses products and reviews. Afterward, we would extract the review data of a product of choice (Nokia phones in our case). Then build an application for affordance annotation that would, with the help of several analysts[2], aid us in the creation of a dataset that would help us accomplish the goal of the thesis.

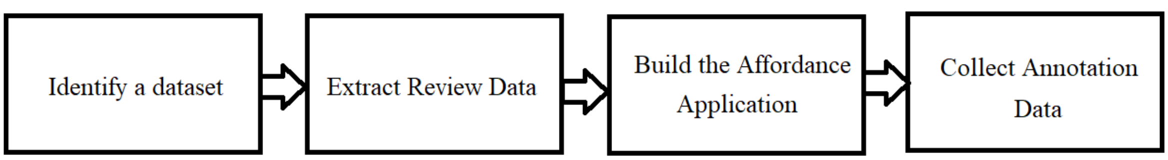


Figure 3.1 Data collection process flowchart

[2] Refers to the people that would use the affordances annotation application

A. Preparing the dataset

The review acquisition attempt underwent several iterations and passed through several sources until we settled on a review set that was deemed suitable for the experiment to be conducted. The initial intent was to utilize an amazon data set containing the Sonos play one product family as an early exploration of the reviews showed very dense reviews that potentially included a significant number of affordances. Also, having different or evolving affordances with every model which would help us in obtaining a more extensive range of affordances form the same item line as it develops. We would eventually pivot and settle for a different set of Nokia phone reviews after going through several review sets. The review sets we explored but eventually dropped from consideration included the following:

- the Stanford analysis network project or SNAP for short[3].
- Kaggle, unlocked mobile phone reviews from amazon[4].
- Kaggle, reviews of iPhone X[5].

[3] can be found at the following website (https://snap.stanford.edu/data/web-Amazon.html).
[4] found at the following website (https://www.kaggle.com/PromptCloudHQ/amazon-reviews-unlocked-mobile-phones)
[5] found at the following link (https://www.kaggle.com/kewalkishang/iphonexreview)

1. Amazon S3 dataset

We decided on the dataset of choice as the Amazon Storage Service (Amazon S3), which operates under Amazon web service (AWS). From there, we found a set of reviews of a large number of amazon products dating from 1995 to 2015[6]. To see the validity of the set, we downloaded a sample and processed it. The single entry contains the following variables that we found to be relevant: a review date, a helpful score (e.g., five out of six found this review helpful), review body, and product ID.

From the same source, we had access to a product dataset that would link to the product ID mentioned earlier. The product dataset had the following tables: Product ID, brand, category, and title.

2. Customized review dataset

The data set boasted around 56 GB worth of reviews and up to 4 GB worth of product information. We processed the data and stored it in a local database, removing all unnecessary fields and doing additional cleaning as well as establishing a table for the products and linking them to the review table. Details regarding the design of the review database are available in appendix 5.

We did a count of the available products and grouped by brands and focused on: LG, Samsung, and Nokia. Looking at the number of reviews and products available, we

[6] can be accessed from the following link (https://s3.amazonaws.com/amazon-reviews-pds/readme.html).

decided that the Nokia phones would best be suited for our purposes. The final set chosen covered 55 different products from Nokia and had a total count slightly less than 500 reviews.

From these reviews, we chose to eliminate any review that did not contain any action verbs to minimize the number of sentences that would yield little to no value. This process was done by extracting the part-of-speech tags, which are the tags that assign grammatical terms to words (i.e., JJ corresponds to adjectives, VB corresponds to verb base, more details in future sections) from the reviews and checking if there was at least one tag that corresponded to a verb (i.e., was of the form VBx where x could be one of many extensions).

B. The affordance annotation application

The application was built out of necessity as the main dataset that the study would tackle was not available in any format, be it online or otherwise. The practical use case of the application is to provide sentences to users and have them annotate what they would consider an affordance. The main goal is to have three users look at the same set of sentences and annotate them and then cross evaluate the results to build the dataset for the analysis, which would contain affordances where at least two out of the three users consider that it is an affordance.

With the application goal in mind, we would use the review database, which would hold all the data from the original dataset we obtained from the amazon s3 website to provide our users with sentences to annotate. As we mentioned previously, we would only use around 500 of those reviews. The total number of reviews that would fit our decided criteria of a Nokia phone came in at 463 different reviews after eliminating any reviews that did not contain any verbs to maximize the number of reviews that would have value.

Given the application's purpose to build a dataset for the analysis with the help of multiple users, the best approach was to have the application created with the intent of being used via a web browser for allowing collaborative annotation of the dataset. As such, it was developed using HTML5, PHP, and used a MySQL server to handle all database transactions.

The application is currently deployed on a server belonging to Dr.Fouad Zablith[7]. In the following segment, we will elaborate more on the application and its design.

1. Application UI

<u>a. Login page</u>

Upon accessing the link above, the user will be directed to the login page of the web app. Due to the limited number of users needed for this experiment as well as the back-end work required to initialize a dataset for each user, we decided that we would pre-create user accounts and provide the usernames and passwords to the participants.

---

[7] Access the application on the following link (https://linked.aub.edu.lb/apps/affordance_annotation/login.php).

Figure 3.2 shows a screenshot of the login page, which showcases the bare minimum required of a login page such as username field and password field. The page also includes a link to a simple instruction manual (Appendix 1) to help users navigate the app as well as detail the concept of affordances to help them with the task:

Figure 3.2: Login page of the affordance collection application

b. Main page

The main page of the application is shown below, with circled numbers pointing at the key features that will be explained underneath each image. This section contains two images, one showing the application page as a whole and the other delving deeper into the main aspect of the page where most of the work will happen.

i. General view

Figure 3.3: The main page layout of the affordance collection application

The numbers on the image refer to:

Part (1) of figure 3.3 refers to the main box where the affordance annotation is handled. Another image down below will go in-depth regarding its features and usage. Note that there are five recurring instances of this box in the image, each containing different text as the application loads five sentences at a time for the user to annotate.

Part (2) of figure 3.3 is the submit button found at the bottom of the page. It is used to refresh the sentences on screen after the user has finished annotating affordances. Upon hitting the submit button, the collected data will be fed into the database responsible for handling data storage, and new sentences will be loaded on the screen. Further details on the database will be covered in a future section.

Part (3) of figure 3.3 is a progress bar that shows users the percentage of sentences for which they have completed the annotation process.

ii. Sentence box view

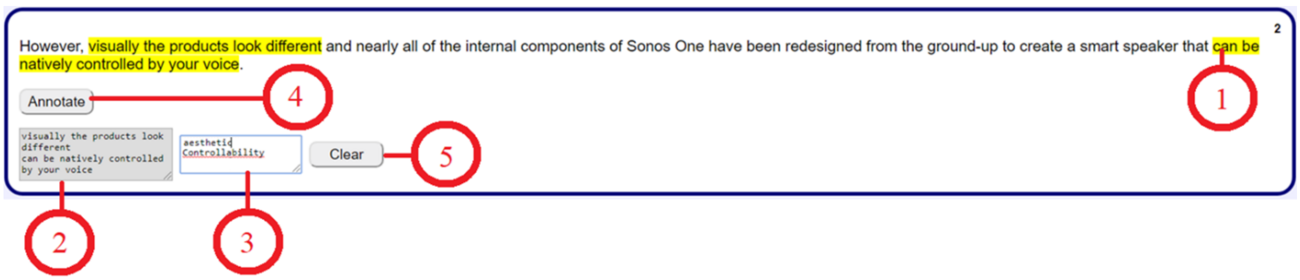


Figure 3.4: A focused view on the individual sentence box

The numbers on the image above refer to:

Part (1) of figure 3.4 is the highlighted text post annotation. The user first highlights the text with their mouse cursor and then presses the annotate button (denoted as (4) in the image) to annotate the affordance they found, turning the background color of the text to yellow to give them the user a clear visual indication that the highlighting worked. Note that there are two separate blocks of text highlighted in yellow, this is allowed as a single sentence may contain multiple affordances.

Part (2) of figure 3.4 is a box meant to store the annotated affordance and is designed to be un-editable by the user. The purpose of this box is to show the user the exact text they highlighted as well as give them a sense of order regarding which affordance was highlighted before the other as each affordance is placed on a separate line allowing a clear distinction between affordances. The second point is relevant to the next element we will explain.

In part (3) of figure 3.4, unlike the previous text box, this one is meant for the user to interact with directly and not just as a means of display. In this box, the user types out a formalized term for the affordance they highlighted. In the example given in the image, the expression “can be natively controlled by your voice” is formalized as “*controllability*”. Note that there are two terms in the box; this is because there are two highlighted affordances. The affordance in the text and the formalized term are placed in the same order in their respective boxes.

Part (4) of figure 3.4, The annotate button that would paint the highlighted text background in yellow when clicked as well shows a copy of this text in the grey box numbered 2 in the image.

Part (5) of figure 3.4 is a clear button to reset this specific box to its original state before any highlighting and typing were done. The clear button will only affect the sentence block, which houses it and not the entire page.

2. The application database

Now that we have seen how the front end of the affordance collection application functions, we will proceed in the following section to further elaborate on the backend of the application.

A different database from the review/product database was created with the intent of housing all the necessary data to run the application. This new DB was called “annoatation_affordance”. Figure 3.5 shows all the tables of this database and their fields while also showing the relationship between those very same fields. The tables of this database are:

- User
- Review
- ReviewSentence
- ReviewSentence_1300
- ReviewSentence_1400
- ReviewSentence_2000
- Annotation
- AnnotationText

We will elaborate further on each table while highlighting its fields and the purpose they would serve in achieving the goal of this thesis.

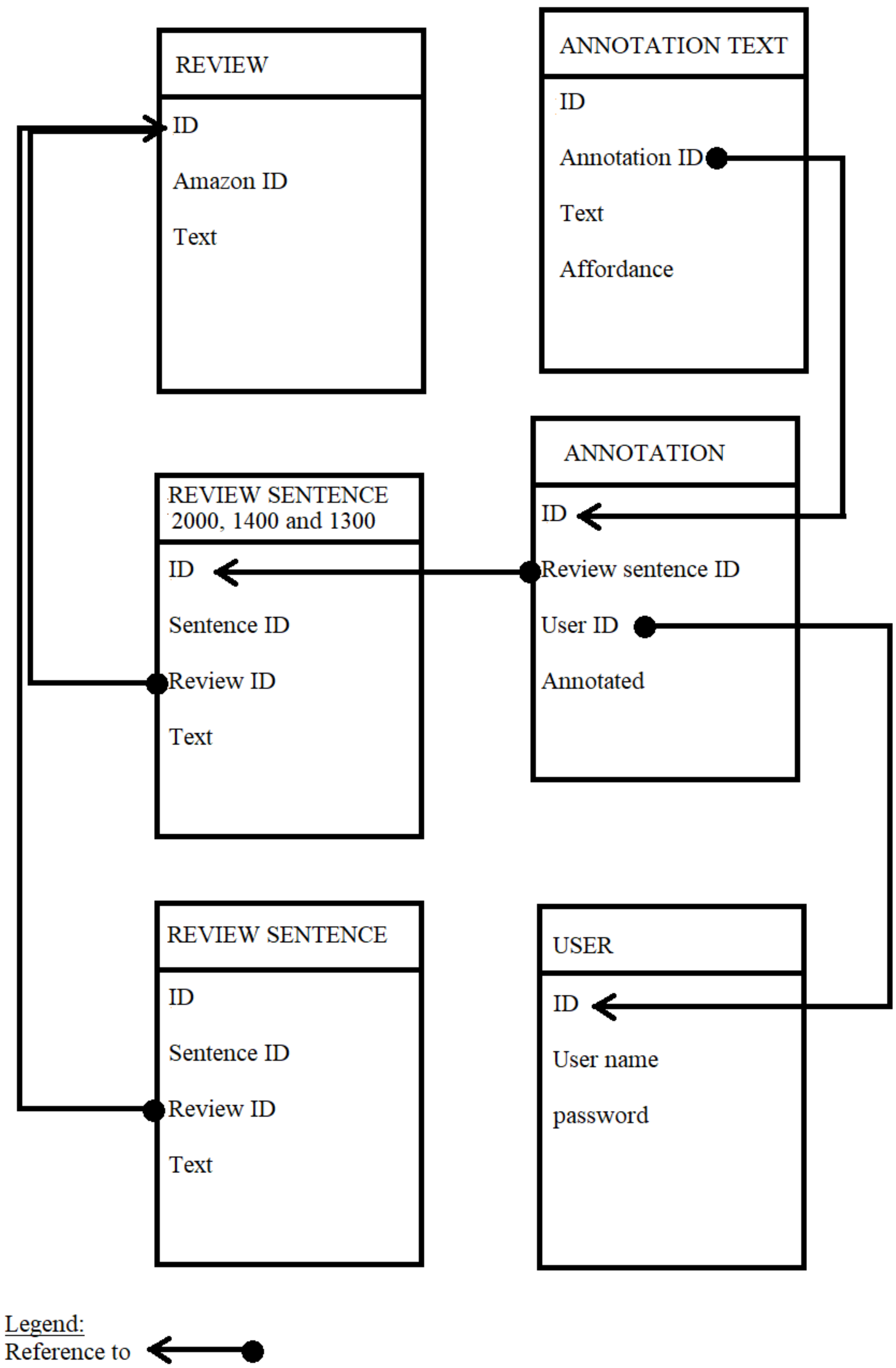


Figure 3.5 The affordance application database design layout

a. User table

The first table in the application is the User table, which keeps track of the users created for the experiment. It contains three main fields:

- ID
- UserName
- Password

The fields are standard fields for a user and do not require further explanation. One note to make is that the username and ID are unique values, and the database autogenerates the ID upon record creation.

We would create nine different user accounts. Each three would share one of these extensions in the username:

- _1300
- _1400
- _2000

The relevance of these will be further explained in the reviewsentence table section.

b. Review tables

First off, the review table would house the full reviews extracted from the metadatareview database discussed previously. This table would hold the 463 reviews that related to the Nokia phone items that were queried from the database, and its fields are:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus the rows are ranked by the order in which they were inserted.
- AmazonID: The amazon ID refers to the asin field of the original review table from the metadatareview database and would allow us to track the exact product that is subject to the specific review in question
- Text: The main content of the review.

In the second table, the reviewSentence table, we would split the reviews from the Review table from this database into sentences using the python nltk library[8] and store each sentence as its own standalone record. This table houses a total of 4797 rows. The fields of this table are:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus, the rows are ranked by the order in which they were inserted.
- SentenceID: a partially unique ID to each sentence within the same review. By this, we mean that if sentences belonged to the same review, this ID would denote the order of the sentences within the review.
- ReviewID: a reference to the original review from where the sentence was extracted. This ID is a reference to the ID in the review table from this database.
- Text: The main content of the sentence.

[8] Found in https://www.nltk.org/

The following three tables share the same structure with the table previously discussed but are named reviewSentence_2000, reviewSentence_1400, and finally, reviewSentence_1300. They all have the same four fields: ID, SentenceID, ReviewID, and Text. The key difference lies in the fact that these three tables are mutually exclusive subsets of the ReviewSentence table. By this, we mean that the original ReviewSentence table's contents were split into these three tables. The sentences were split then sampled randomly, and as such, when shown in the annotation tool, will not have cohesive meaning.

The purpose of this was to allow each three users to work on a different set of sentences and to lower the number of time users had to spend working on the affordance collection task as we found out it was quite time-consuming. Recall that each username had an extension attached to it, that extension would serve to tell the server which set of sentences the user would have access to. By this, we mean that the application would have each three users annotate the same sentence as long as they had the same user name extension.

The sets denoted with 2000, and 1400 contain 2000 and 1400 sentences respectively, but this is not the case for the set denoted with 1300, which held 1379, which is the remaining amount when subtracting the first two sets. The name 1300 was put in place for convenience's sake to simplify access to the set.

e. Annotation tables

The first table, called annotation table, serves as a reference for the server that handles showing users their sentences to know which sentence has been highlighted by which user and is at the core of the sentence loading mechanism of the application. We will detail the fields of this table then elaborate on how it was populated and the intent behind it:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus, the rows are ranked by the order in which they were inserted.
- ReviewSentenceID: a reference to the ID of the sentence from the reviewSentence_X where X can refer to any of the three possible options of 1300, 1400, 2000. The specifically chosen set is based on the extension of the username
- UserID: a reference to the user ID from the User table
- Annotated: used as a Boolean variable having a value of 0 or 1 and denotes if the user has had the sentence loaded on the page. If the user has seen the sentence on the web page and pressed the submit button, the value is switched to 1. The server uses this field to keep track of the sentences that were shown to the user as well as show the user all records that have the value as 0 in sets of 5.

The way this table was populated is mainly based on the multiple versions of the review sentence tables and the user. For each user, there exists X amount of records in this database with X referring to 2000, 1400, or 1379. As we said in the ReviewSentenceID section, this choice is based on the username extension. The reason behind this was mainly to be able to track each user's progress independently via the annotated field without causing any overlap with other users. This in tandem with the fact that the application server-side code can automatically know which user belongs to which review set makes the usage of the review sentence ID of the multiple tables feasible even if there is a duplication of IDs as the application will know from which source to draw from.

Unlike the previous tables, which were all pre-populated before the experiment began, the annotation text table starts off empty. It is the main table of interest for the analysis segments. This table will record all the highlighted affordance in the review sentences as well as the formalized affordance term submitted by the user, if possible. The fields of this table are:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus the rows are ranked by the order in which they were inserted.
- AnnotationID: a reference to the annotation table ID, which allows us to track both the user and the sentence from where this record was generated. In the case of multiple affordances, each is treated as its own record, a process done by the server before insertion.
- Text: the highlighted and annotated text by the user, which should denote an affordance.
- Affordance: The formalized term of the affordance provided by the user when possible.

3. The data collection process

Although the application was built to be used by a maximum of nine users, dispersed over three sets of sentences, our goal would have us only use one of the sets, the one having 2000 sentences and relying on only three users for a proper cross-evaluation of the validity of the affordances once they finished their annotation task.

Once our three users finalized their task, we would revisit the affordance annotation table and extract its' data, which would serve as the base for the dataset that would be used in the analysis. The next segment will explain how the raw affordances collected by the three users would be processed to obtain a final set, and the steps taken in the analysis.

CHAPTER IV

DATA PRE-PROCESSING

AND ANALYSIS METHODOLOGY

The end of the previous chapter saw us acquire an affordance dataset generated by users. In this chapter, we will put the dataset generated by the users to several analysis methods in order to generate insights and reach the goal set by the thesis. The main focus of this chapter is to detail and explain the aforementioned analysis methods taken in handling the data.

Figure 4.1 shows a detailed roadmap that will lay down all the approaches taken. As seen in the figure, we start with our affordance dataset from the last chapter and perform a simplified inter-rater agreement measure[9]. The measure looks at cross-user matching to obtain affordances where two or more users agree. Then we perform an agreement matrix check to see how well users agree on what is deemed by analysts to be an affordance to validate the dataset. Using the pre-processed dataset, we would create new fields and perform general data cleaning before putting the dataset to test via our main algorithm to narrow down on the key grammatical patterns that affordances exhibit.

[9] More inter-rater reliability: https://en.wikipedia.org/wiki/Inter-rater_reliability

Finally, using the findings provided by the techniques used in this section of the thesis, we would move on to the results and evaluation of those results in the following chapter.

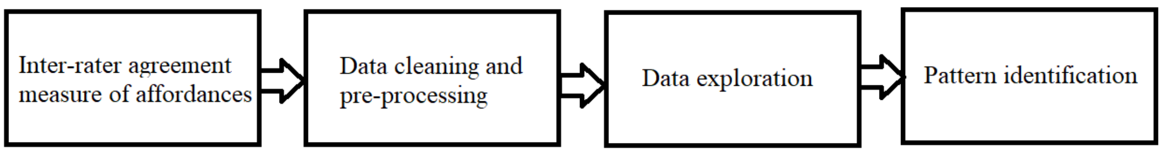


Figure 4.1 Data analysis methodology process

A. Inter-rater agreement and dataset creation

In the following section, we will take the affordances generated by the analysts and, by checking for a consensus between at least two or more analysts, we would create a final working dataset. Afterward, we would check the agreement rate on what analysts consider to be the affordance with the intent of finding out how diverse or homogenous the set may be.

1. Creating a final dataset

Having all the analysts of the set of 2000 sentences finishing their annotation, we proceeded to extract all the data they have collected from the annotationText table from the annotation_affordance database.

Given the subjective nature of affordances, we relied on multiple analysts to annotate what they believe is an affordance. To have a more concrete definition of affordances, we would employ a simplified inter-rater agreement measure. This value serves to measure the degree consensus within a given set of ratings made by several judges. Though the original value dives into more complex statistics and is applied on more concrete sets, we decided to use the measure as a base an apply a more simplified variation due to the small number of categories and subjective nature of the affordance concept.

With this information in mind, we would consider a record to be a valid affordance if at least two analysts (in this case, the analysts is the rater or judge) agree that the statement is an affordance (in this case, the affordance is the rating). We highlight below the affordances collected by each analyst, denoted here as User 0, 1, and 2.

The individual number of affordances detected by each user are as follows:

- User 0: 248 detected affordances
- User1: 175 detected affordances
- User2: 1315 detected affordances

As we said, to establish a more concrete and agreed upon affordance, we shall proceed to cross-examine the different results from each user with each other to find where they agree and where they would disagree regarding affordances. To that end, we would first join the datasets based on the sentence ID and then check the sentence similarity as a single sentence could hold multiple affordances, and thus a match on sentence ID would not reflect a match on affordance.

When checking for sentence similarity, we realized that we obtained different sized affordances from different analysts that were still conveying the same core idea. As a result, we decided to use a similarity rate[10] as the crux of inter-rater agreement measure, and after experimenting with the cutoff rate, we decided that a 35% cutoff rate would be best. That is to say that if the highlighted affordance form both analysts in question reaches at least a 35% similarity rate, then that sentence would qualify for the final working set we would use. To better illustrate this, we shall demonstrate with an example from the data set:

-User 1 annotation: "N8 is able to play all sorts of vdo file formats"

-User 2 annotation: "able to play all sorts of vdo file formats"

In this case, the similarity was 0.93333, and as such, the statement would be considered valid for the final working set.

After running the above process on all permutation of the three user sets, we obtained the following results:

- Cross between user 0 and 1: 26 total affordances
- Cross between user 0 and 2: 190 total affordances
- Cross between user 1 and 2: 44 total affordances

After obtaining these sets, we ran one final operation to concatenate them while accounting for possible duplication of values, and the last set that we got would see itself house 219 affordances.

[10] Sequence matcher used https://docs.python.org/2/library/difflib.html

2. Agreement matrix

The next step in our simplified inter-rater agreement is to check the level of consensus. With the intent to further validate the agreement between analysts, we decided to observe how often they would agree that a sentence would hold an affordance and how often they would agree that it does not.

With this idea in mind, we decided to build "agreement matrices" between every two analysts to see how often their ideas of an affordance would align. The term "user" will be used to denote analysts in the following segments as it aligns with the naming used in the datasets and source code.

<u>a. User 0 and 1 agreement matrix</u>

Between user 0 and 1, we had the following results:

- 56 sentences in which both users agree there is an affordance
- 1670 sentences in which both users agree there is no affordance
- 169 sentences in which only user 0 found an affordance
- 105 sentences in which only user 1 found an affordance

The total number of agreed-upon sentences is a sum of agreement on found and not found affordances, and thus, in this case, the final result is 1726 sentences out of 2000, which amounts to an 86.3% agreement rate.

<u>b. User 0 and 2 agreement matrix</u>

Between user 0 and 2, we had the following results:

- 182 sentences in which both users agree there is an affordance
- 1007 sentences in which both users agree there is no affordance
- 43 sentences in which only user 0 found an affordance
- 768 sentences in which only user 2 found an affordance

The total number of agreed-upon sentences is a sum of agreement on found and not found affordances, and thus, in this case, the final result is 1189 sentences out of 2000, which amounts to a 59.45% agreement rate.

c. User 1 and 2 agreement matrix

Between user 1 and 2, we had the following results:

- 104 sentences in which both users agree there is an affordance
- 993 sentences in which both users agree there is no affordance
- 57 sentences in which only user 1 found an affordance
- 846 sentences in which only user 2 found an affordance

The total number of agreed-upon sentences is a sum of agreement on found and not found affordances, and thus, in this case, the final result is 1097 sentence out of 2000, which amounts to a 54.85% agreement rate.

The agreement rate would average around 67%, which we considered a valid percentage to form the final dataset, which we would inevitably use for the data analysis process of this experiment. The agreement rate highlights that users have a core understanding of the concept of affordances but still retain their view regarding this topic, which is what we are seeking. This value, in turn, highlights the complex nature of the affordance concept.

B. Data pre-processing and exploration

The previous section provided an affordance dataset created from the user annotated affordances through a custom made simplified inter-rater agreement measure. In this section, we will first showcase the pre-processing steps taken to flesh out the data and have better working fields. From there, we would use the new set first to explore and get an idea and a general feel of the data.

1. Data cleaning and field creation

a. Trimming the final set

Having the final dataset containing 219 records, we proceed to choose the relevant fields that we would be using selectively. After the merge of the multiple user collected data, we were left with several duplicates of the fields. Recall that users had sentences from the same review compared and chosen to be part of the same set if a 35% similarity rate is achieved. There was no processing done on the text directly, and all the final dataset would have a text and affordance field for both users. To clarify, the dataset would have the following fields if left unchecked:

- user 1 text

- user 2 text
- user 1 affordance
- user 2 affordance

Due to this duplication, we decided to trim down the number of fields in the dataset.

To that end, we opted for a straight forward approach of taking the text element that had a larger length between the two users. The driving thought here is rooted in the idea that both text values for both users already cover the same text element within the review but at different percentages. Thus the longer of the two would yield more overall value and is more likely to encapsulate the totality of the other statement.

As for the affordance field, given that, not all statements would have affordances, and the fact that no processing was planned to be executed on this column, we chose to duplicate the value of the user that had the most affordances typed out as a temporary placeholder and would return to it if time allows.

b. New field creation

To help with more advanced analysis, we proceeded to extract additional fields from the affordance text from the final set. The new fields added would tackle part of speech tagging, stemming, and stopword handling. The final fields of the dataset would be:

- Text: the original text of the affordance
- Affordance: the formalized affordance term written by the user

- Words: an array that contains a tokenized version of the original text field where each element within the array is a word in the text. The main purpose of this array is facilitating certain tasks such as track word location in the text via array index.
- Pos_tags: a part-of-speech tag version of the original text stored in an array format. The part-of-speech tags break down the words into their basic function as words in the English language and assigns them a tag such as NN for nouns and VB for a verb base (tag definition available in appendix 2).
- Stemmed_words: an array containing the stemmed version of words from the text field. Stemmed words are simply the root of the word in question with any additional suffix removed.
- Stem_no_stop: an array of words that uses the Stemmed_words array and removes all stopwords from it. Stopwords are words that have a very high presence within a language and do not provide any benefits when analyzed.

2. Data exploration

With the dataset preparation complete, we proceed to run a data exploration process on the several columns we have synthesized. The analysis is based on the affordance text column and its derivatives.

Below is a list of the multiple approaches taken and some additional notes when necessary.

- Ran a count of the most occurring words in the base untampered text (Figure 5.1), and as expected, the most recurring words are the most common ones in the English vocabulary, i.e., stopwords.
- Ran a count of the most occurring words in the text after stemming and removing stopwords (Figure 5.2), and the result shows a massive distinction in terms compared to the previous step where the analysis was done with all words. We notice the overwhelming presence of "I" along with several phone features such as phone, screen, camera in addition to some verbs that would reflect how users would interact with these features.
- The average count of words for an affordance is 14.9 words.
- Ran a count of the most occurring tags from the pos_tags field. (Figure 5.3)

With the data exploration done, we get a basic set of information regarding our dataset that provides some minor but interesting insights. With this step done, we move on to the central pillar of the thesis, the data analysis.

C. Data analysis

All the steps mentioned earlier served primarily to allow for the analysis and processing of the affordance data to derive results that would inevitably push us closer to our goal. With that in mind, we would try different approaches that would yield results. However, only one approach would contribute directly to the end goal of the thesis, the pattern recognition algorithm.

In this segment, we will first lightly brush on the other techniques used and then dive deeper into the inner workings of the pattern recognition algorithm.

1. Honorable mentions

In our attempts to derive meaningful and valuable insights from our dataset, we would attempt a handful of methods and techniques that deserve an honorable mention as they do provide their own sets of results, albeit a bit deviated from the main goal of the thesis. These are:

- TF-IDF
- K-means clustering
- Pattern analysis in clusters
- Predictive neural network

The details and findings of all these methods are available in appendix 6.

2. Pattern recognition

With the multiple methods mentioned above, the pattern recognition approach stood atop the list of techniques that would drive us towards our goal. In this section we will cover the inner workings of the techniques and explain in detail how the recognition of pattern works.

In simple words, the purpose of the algorithm is to find the most recurring pattern of part of speech tags, which are indicators of the grammatical function of words within the affordance text. The algorithm that takes as input the pos tags column and the number of tags that the pattern would contain. Figure 4.2 shows the code for the pattern finder function, as well as explain the purpose of each line of code in the comments. Nonetheless, we will elaborate more on this algorithm by explaining every step of the process and providing an example.

Looking at the parameters first, the input for the algorithm is the list of part-of-speech tags derived from the affordances and the number of tags per pattern. The number is a representation of how many tags the patterns will have and based on which we will consider a pattern a valid entry for the final output. The latter being a dictionary containing patterns acting as key and a count of how many times they occurred as the value.

The algorithm would process each affordance tag list on its own, and given the small size, the runtime was not an issue. The number of tags per pattern chosen in this case is three. This number implies that a valid pattern would include three tags; for example, NNP VBZ JJ is an entry in the final output.

```python
#pos_df is the list of affordance tags and pat_num is the
#number of tags to be considered in a patter
def pattern_finder(pos_df, pat_num):
    #the empty dictionary to be return
    pat_dict = {}

    #go over the tags of every affordance
    for tags in pos_df:

        #check is a ta list is empty in order to skip it
        if pd.isnull(tags):
            continue
        #split the tag list of the current affordance into
        #individual tags
        tags = tags.split(" ")
        # check if the affordance has less tags than the desired
        #pattern number
        if len(tags) < pat_num:
            continue

        #end range is a temporary value that tell teh algorithm to
        #stop before reach the very end
        end_range = len(tags) - pat_num

        #here we build the pattern of tags
        for i in range(end_range):
            pattern = ""
            for j in range(pat_num):
                pattern = pattern+tags[i+j]+" "

            #check if the pattern is in the dictionary if yes,
            #add 1 to the count
            if pattern in pat_dict:
                pat_dict[pattern] += 1
            #if not, add it to the dictionary
            else:
                pat_dict[pattern] = 1

    return pat_dict
```

Figure 4.2 the code for the pattern finder function

To explain how this algorithm works, we shall use the following affordance as an example:

Text: “N8 is able to play all sorts of vdo file formats”

POS: NNP VBZ JJ TO VB DT NNS IN NN NN NNS

Moreover, as we stated earlier, we will also use an empty dictionary as a storage variable and will be used as output. The current state of the dictionary, at the start being empty and denoted by :

Output : {}

The program will scan the tags in sets of three, which is the tags per pattern number set in the input, starting from the beginning of the tag list, so initially, the program will cover the underlined three tags in the following:

<u>NNP VBZ JJ</u> TO VB DT NNS IN NN NN NNS

After scanning, the program will check if the underlined sequence is in the output dictionary. If it is not found, the sequence itself is added as the key of the dictionary element and sets the value as 1. On the other hand, if it is found, the value of that sequence is incremented by one. The dictionary is updated to the following:

Output : {NNP VBZ JJ : 1}

In the output example above, the left-hand side denotes the key, which is the pattern, and the right-hand side is the number of occurrences so far.

In the next iteration, the program will cover the next sequence of three, starting by the element immediately after the first element of the first sequence and not the three elements following the first sequence. In our example, that would be:

NNP <u>VBZ JJ TO</u> VB DT NNS IN NN NN NNS

The dictionary state is:

Output : {NNP VBZ JJ : 1,

VBZ JJ TO: 1}

Given that the second pattern scanned was not in the dictionary, it is added and given a count of 1.

In the case a pattern was found that matched an existing one, the dictionary will update the appropriate key (pattern) by incrementing the value by one for each detection. Assume that we came across another NNP VBZ JJ, the dictionary would become:

Output : {NNP VBZ JJ : 2,

VBZ JJ TO: 1}

The algorithm will repeat these actions until there are no more tags to scan in the current affordance. After this, the algorithm will move to the next affordance and repeat the process while maintaining the same dictionary. Finally, the algorithm will return a dictionary full of a wide range of patterns with their respective occurrence count.

Now that we explained how the algorithm functions, as well as having covered all the necessary steps need to reach this point, we finally have an output from which we can derive valuable results. The dictionary produced by the algorithm will act as the primary source of insights that we will cover next in chapter 5.

CHAPTER V

EVALUATION AND RESULTS

In the last chapter, we discussed the inner workings of the techniques we utilized; we move on now to discuss the results and insights we yielded from the experiment. We will first discuss the contributions of the by-product of this experiment, the affordance annotation application, followed by the general data exploration, and finally, we will dive into the analysis results of the pattern recognition algorithm.

A. Affordance annotation application

The affordance annotation application, born out of necessity for this experiment, is one of the achievements accomplished by this thesis and has its potential for further use. The application serves as a useful tool for any future research that would be done on the topic of affordances, and that would require a large amount of text data. The tool is built to be user-friendly and simple to navigate and includes an instruction manual that is always accessible from the webpage.

The tool is built with elementary and basic tools. Those tools have several online communities that would help anyone unfamiliar with the languages used in the making of this app makes it easy to edit to fit specific needs such as adding analysts, changing sentences and reviews, and so forth.

Though it went without any proper use, the affordance formalization box provided in the application can be used as the main subject of interest in a future research paper.

Having the input of multiple analysts serves to provide a more robust and reliable dataset rather than having it skewed to reflect one analyst's personal definition of an affordance. The latter approach would have some degree of bias and subjectivity attached to it, which would inevitably tamper with the results.

B. Data exploration results

The data exploration yielded results that helped us gain a basic understanding of the dataset. Of these results, we have the average length of an affordance in words and the most recurring words. The average word count would be 14.9 and the most occurring words, after remove all common words in the English dictionary (such as the, to, and, etc.) would be "I"," Phone" and "use" as seen in figure 5.2, in addition, we see that the most occurring tag is the noun tag (NN) (figure 5.3). We speculate that the heavy presence of nouns is mainly due to the nature of the topic where users are discussing phones and their features, and as such, those would be the most widespread.

As for the affordance word count average, affordances can have very different sizes, and as such, nothing conclusive can be said, but we can assume that a statement is more likely to be an affordance if it has a size near 14.9 words.

An argument can be made regarding the words as the most common words found in the affordances were "I", "Phone", "Use", "this", "screen", "camera", "app" and "take".

These terms reflect the original affordance concept coined by Gibson which requires a user, in this case, the “I”, an environment portrayed here by the interaction with the “phone” along with the nature of the interaction provided by the verbs and finally the stimulus which is denoted by the phone features such as “app”, “camera” and “screen” (Gibson, 1977). This helps us further confirm that the data collected is less likely to be random sentences and thus actually have what could be more concrete affordances. In addition, we can make the argument that a written affordance by a user in an online review would word the user interaction with the object and the specific feature in question.

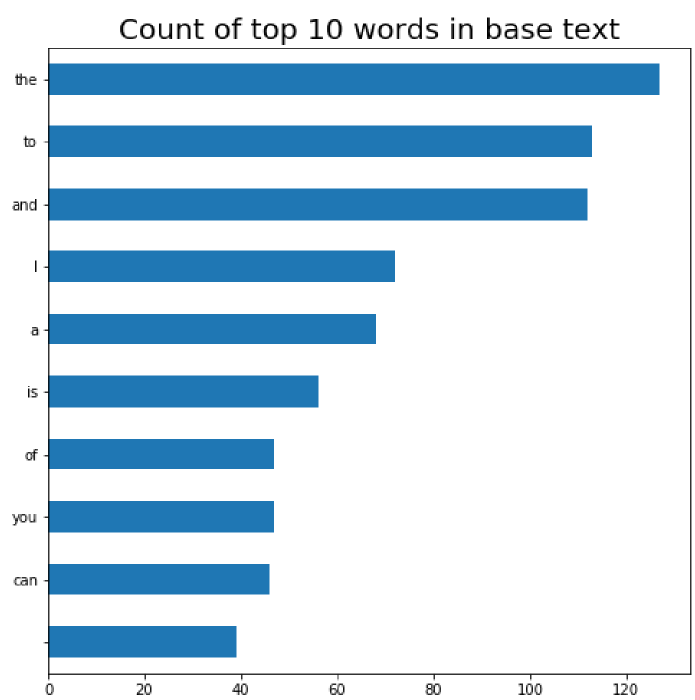


Figure 5.1 Count of the top 10 words in base text

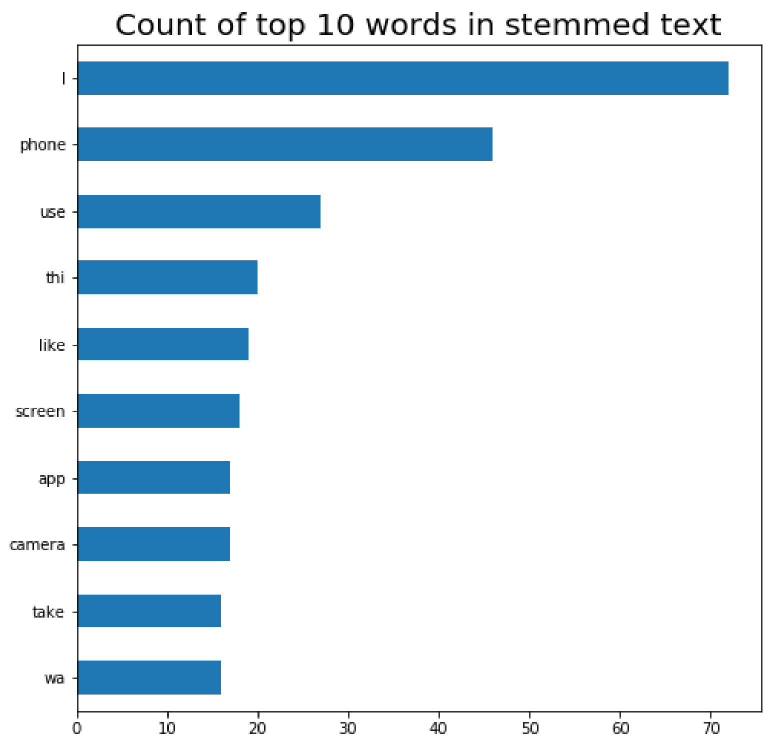


Figure 5.2 Count of top 10 words in stemmed text

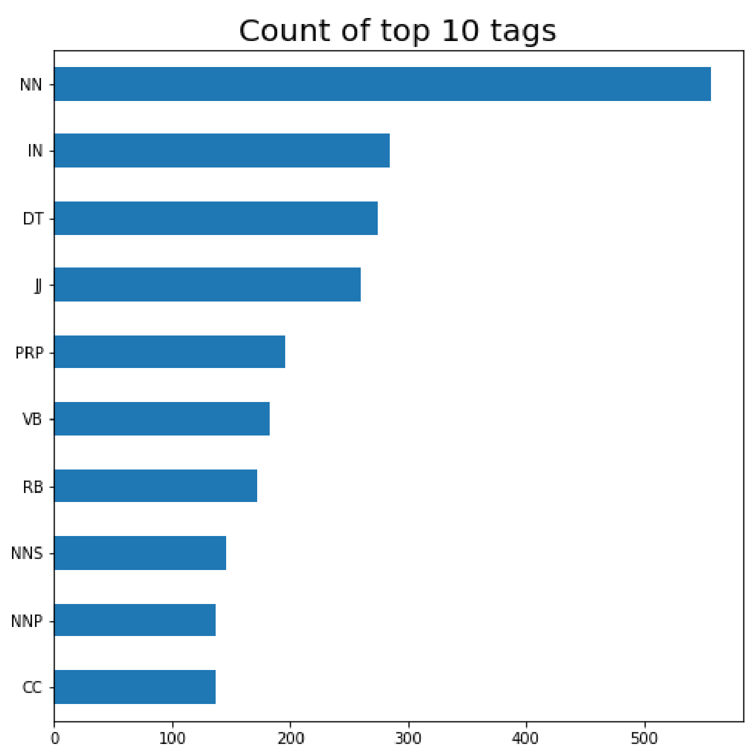


Figure 5.3 Count of the top 10 tags

C. Pattern recognition results

The pattern finder function described in the previous chapter was the most notable of all our attempts at deriving valuable insights from the affordance data. The function was run on the complete set of affordance tags, and the results are displayed in figure 5.4. The exact counts of the top 5 tag patterns are as follows:

- PRP MD VB: 42
- IN DT NN: 39
- DT NN IN: 34
- DT JJ NN: 32
- NN IN DT: 27

These tags refer to personal pronoun (PRP) such as I, he, she, a modal (MD) such as could and will, (VB) is the base form of a verb like to take for example, (IN) is a preposition such as except and about, nouns (NN) in a singular form like phone and screen, and adjectives (JJ) such as big, and finally determiners (DT) such as a, this, one.

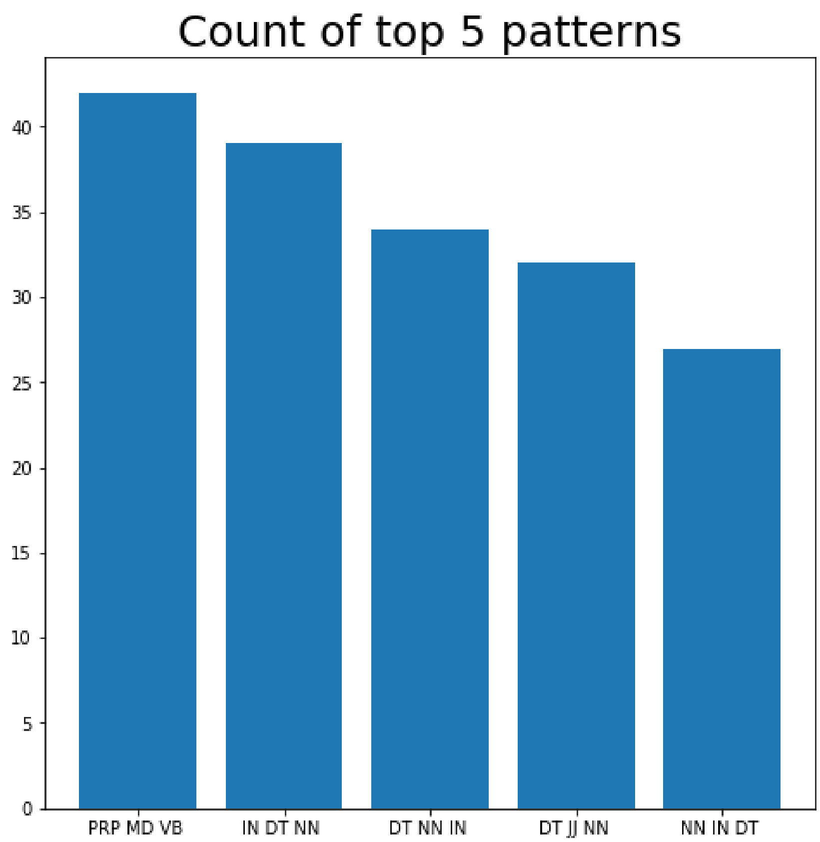


Figure 5.4 Count of the top 5 patterns

Looking at the results for pattern recognition, we see the rise of 5 main patterns that are the most recurring within the sentences. The patterns are:

- PRP MD VB such as “I can use” and “It can provide”
- IN DT NN such as “on any phone” and “with this device”
- DT NN IN such as “a year after” and “the apps above”
- DT JJ NN such as “A little difficult” and “any decent phone”
- NN IN DT such as “speaker in the” and “finger across”

The nature of some of these patterns would not give much value as 2 out of the three terms, not words that yield much weight, such as determiners and prepositions. Still, nonetheless, the vast presence of these patterns may be a clue to more effectively identifying affordances. Looking at the two patterns that do not fit the criteria described above, we are left with DT JJ NN, which is a determiner, an adjective and a noun, and PRP MD VB which is a preposition, a modal and a verb in base form. Out of these two available options, the highest-ranking one in terms of count, PRP MD VB is the one that provides the most interest as the alternative is a simple adjective and noun combination, which is to be expected in a product review.

Checking the word count for all the terms in Figures 5.5, 5.6, and 5.7 that belong to the PRP MD VB sequences revealed a widespread of words per tag. First, off the most recurring pronoun is "you," which is reasonable since users are addressing those who have an interest in purchasing the product by saying, "you can do…". The pronoun "I" follows in second place, which serves as a way for users to describe their personal experiences as well as being the most recurring word in the dataset, and in third place, we have "it", which is most likely a reference to the product or one of its features. The modal "can" dominated the count within its category, taking around 75% of the total amount of MD occurrences and serves as a way to express how the user or the product can accomplish something which is denoted by a verb with no real standouts when the count was done.

This result also aligns with Gibson's view (Gibson, 1979) as users are refereeing to their interactions with the feature/product or the feature/product's abilities to perform a task that they deem valuable.

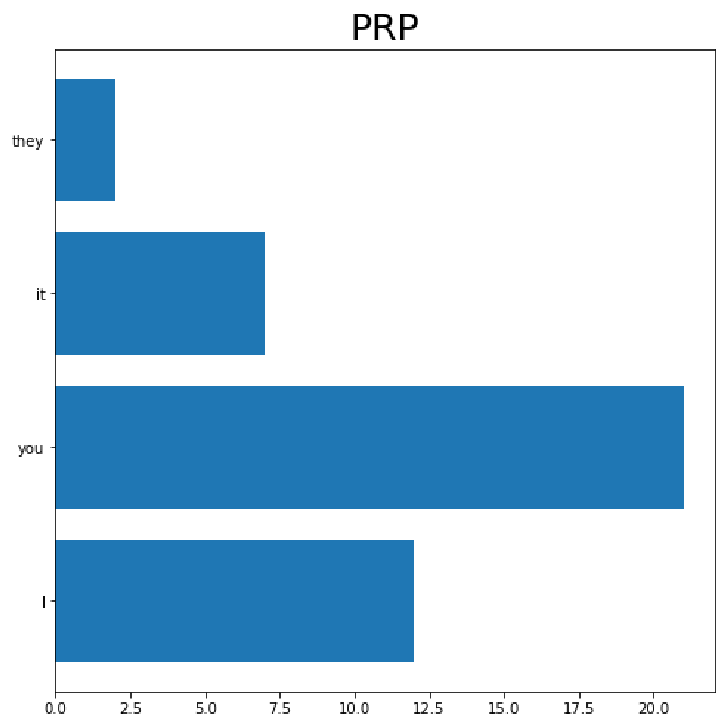


Figure 5.5 Count of PRP words in the pattern

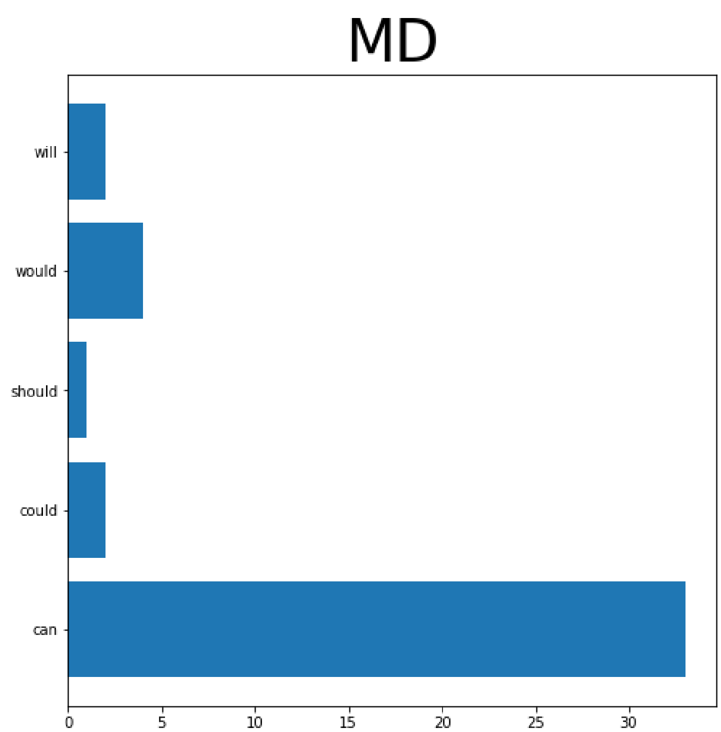


Figure 5.6 Count of MD words in the pattern

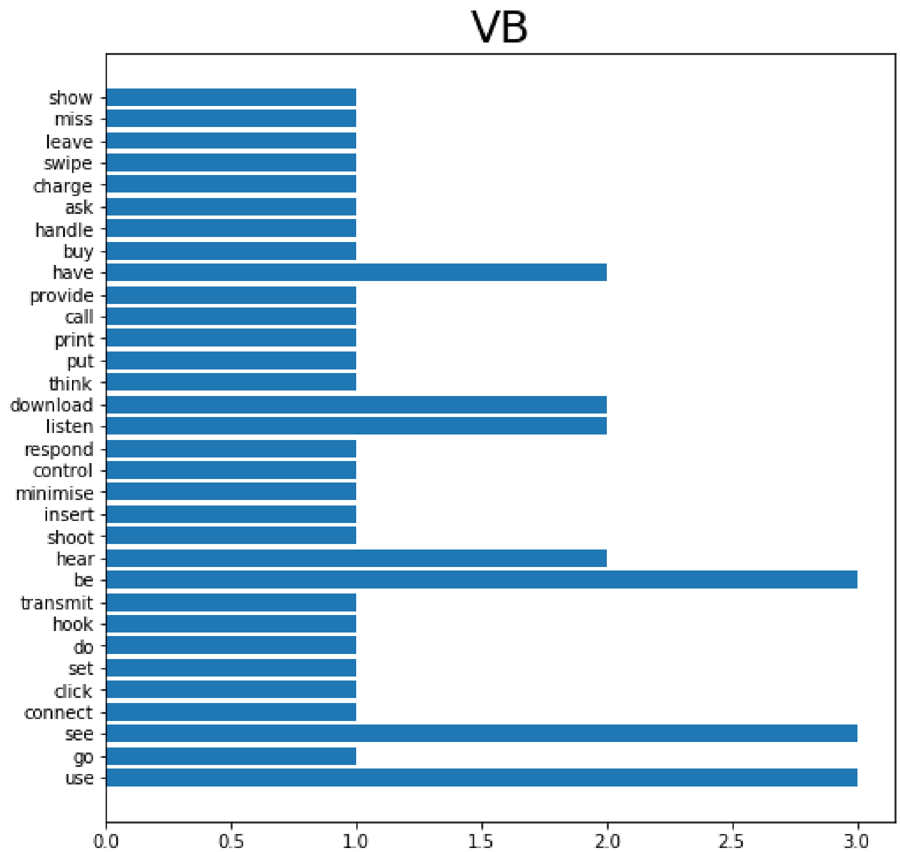


Figure 5.7 Count of VB words in the pattern

Although we mentioned that most of the other patterns would not yield any significant results due to the abundant presence of stopwords or filler words like determiners and prepositions, we notice that they all have a common element, which is a noun (NN). From this observation, we first notice an alignment with the most occurring tag in the text, which is the noun tag (NN). Second, when observed in tandem with the previous pattern, the high presence of nouns in affordances serves to further align with Gibson's concept in which the affordance stems from an environmental stimulus (Gibson, 1979) such as an object or feature, in our case, represented by the nouns. Furthermore, when tying in the previous pattern PRP MD VB with the nouns NN, we can argue that we get a fuller representation of the affordance concept with an organism, action, and stimulus. To further confirm our theory, we look at the list of nouns present in all 4 patterns:

year: 1	device: 1	battery: 1	phone: 1
desk: 2	speaker: 1	ringtone: 1	music: 1
darkroom: 1	PC: 1	restaurant: 1	car: 1
return: 1	photo: 2	movie: 1	TV: 1
signal: 1	contact: 1	touch-screen: 1	user: 1
rate: 1	difficulty: 1	dog: 1	switch: 1
message: 1	zoom: 1	application: 1	place: 1
finger: 1	stop: 1	file: 1	configuration: 1
world: 1	keyboard: 2	time: 1	back: 1
attention: 1	edge: 1	apps: 1	iPod: 1
side: 1	map: 1	need: 1	window: 1
home: 1	bit: 2	ease: 1	bummer: 1
internet: 1	couple: 1	sharing: 1	bargain: 1
screen: 2	page: 1	volume: 1	computer: 1
calendar: 1	wireless: 1	camera: 1	built: 1
feed: 1	button: 1	lieu: 1	None: 1
feel: 1	xenon: 1	game: 1	price: 1
lag: 1	version: 1	one: 1	inability: 1
day: 1	voice: 2	level: 1	lot: 2
shortcut: 1	app: 1	menus: 1	thing: 2
qwerty: 1		country: 1	

Figure 5.8 List of nouns found in patterns

along with their occurrences

The first observation is a very distinct list of words, as the highest duplication rate is 2. Second is the overwhelming presence of phone-related nouns, which is what we speculated and thus help make a more robust case for our earlier assumption. However, some nouns stand out, but this may shed light on the last element of the affordance. As we know, Gibson had three elements in his definition of the affordance, the organism, and stimulus, which we already covered, and lastly, the environment (Gibson, 1979). The nouns that stood out are “country”, “restaurant” and “dog”. A case can be easily made for country and restaurant as places where phones would be used, but “dog” was an outlier. Upon further investigation, we found that it stemmed from “a fast-moving dog” which was inevitably referring to the camera’s ability to take pictures. Although the examples are sparse, the nouns take on a dual value as both feature and environment.

In conclusion, we can argue that the results are aligned with the affordance concept. Thus, by using these results, we can employ the patterns above to help narrow down the search for affordance by checking for the presence of any of these patterns in reviews, especially the pattern PRP MD VB. Also, distinct and dominant words such as “can” under the modal tag can also be searched to narrow down the search further and possibly increase the accuracy. The previous point can be utilized along with a search for a specific noun that constitutes a feature whose affordances we seek to find or a specific environment to study. It is also worth noting that the presence of dominant patterns in the affordance text supports the idea posed in the Canadian tire corpus research (Shu & Chou, 2014) that certain expressions and terms are statistically more likely to be present in affordances.

D. Summary of results

To summarize, this experiment provided a tool that can help in the creation of a dataset of affordances, something that was not available previously. We also found several patterns that are recurring in affordances with a highlight on PRP MD VB, which shadows Gibson’s take on the term (Gibson, 1979) and the heavy presence of the term “can” in most of the affordances belonging to the pattern along with an overwhelming presence of nouns.

From here, we can return to the final goal set by the thesis, which was building a grammatical structure for affordance. We propose the following as the model on which affordances are expressed in textual sources. A statement may contain an affordance if it includes a preposition (PRP), a modal (MD), and a verb (VB) in this order, acting as the organism and the action. Moreover, this string of grammatical structures will be accompanied by a handful of distinct structures containing nouns (NN), which will be the object or stimulus of the affordance. A case can also be made that the nouns may be a reference to the environment, but most of the results, as we have seen, tend to be skewed towards the feature.

Additionally, some keywords can further increase the likelihood of an affordance. These are the modal “can” and a reference to oneself, such as “I” under prepositions (PRP), and “you” primarily in the context of a review to communicate the experience to potential buyers.

CHAPTER VI

CONCLUSION

A. Research summary

In this thesis, we have proposed a framework for detecting and extracting the affordances of artifact features based on online product reviews. The approach was rooted in the original formulation and subsequent elaborations of the notion of affordance proposed by Gibson's ecological psychology of perception (Gibson, 1977).

Outside ecological psychology, especially affordance scholars in engineering and product design, have shown interest in being able to search for needed features of products within online reviews by customers and users. For example, Chou and Shu (2014) used reviews to detect affordances about car tires (Chou & Shu, 2014). Others have focused on automating the detection of affordances from images (Nguyen, Kanoulas, Caldwell, and Tsagarakis, 2017). Still, others have further advocated that affordances rooted in product features could be a viable means of finding out about users' views of affordances (Pucillo & Cascini, 2013). Our proposed approach attempts to build on these works but offers a practical approach to extract affordances from the text by looking at grammatical cues and attempting to build a pseudo-grammar to help with affordance identification, something which would act as the initial building block in future text-based affordance extraction.

The initial steps in the thesis (shown in figure 2.1) were focused on two main tasks, creating an affordance dataset and processing the data. The first task (shown in figure 3.1) involved creating the affordance dataset. We started first by obtaining an online review dataset that fits our criteria, followed by developing an affordance annotation application software to be used by three analysts to build an initial affordance dataset. The second major task is the analysis (shown in figure 4.1) of the dataset. We did the following steps: processed the data, checked for analysts' inter-rater agreement, performed additional necessary pre-processing, applied the pattern recognition algorithm, followed by the presentation of the results and reflections.

There were approximately 2,000 sentences fed to the affordance annotation application software and, through the work of the analysts, and after the pre-processing, this yielded 219 text segments that appeared to contain affordances. We ran our analysis on the final dataset of affordances, which constituted the eventual "dataset" for pattern recognition and analysis.

We summarize here our key results that were based on analyzing five part-of-speech patterns. The most occurring pattern (PRP MD VB) appeared to be a commonly recurring structure within the text as the "syntax" for any affordance. That is, a personal pronoun (PRP), modal (MD), and verb (VB) sequence would be the most observed pattern of text to signify an affordance as well as having distinctive nouns to highlight the actors for whom an affordance is available. In addition, pronouns (PRP) such as "I", "you" and "it" are common in affordance text patterns in the context of product reviews, and "can" being the modal (MD) that would be most likely to occur within an affordance with a rate of approximately 75%. Perhaps unsurprisingly, the verbs (VB), (shown in figure 5.7), were distinct and did not converge towards any specific verb.

In regards to the remaining four patterns, two out of three of the terms are common terms, such as determiners and prepositions, which would imply that they would probably not yield valuable or useful patterns. However, the fact that these patterns or lack thereof are occurring may in itself be an important finding that offers useful clues to identifying affordances.

In addition, as a by-product of this thesis, we have developed a tool that can be used to parse text into those containing affordances in a more systematic way and with a more intuitive user interface.

B. Research limitations

Though the results were useful, several factors hindered our approach. The analyst recruitment did not proceed according to the original plan and, as such, limited the size of the final working set. Initially, the intent was to have 6 to 9 analysts. However, given the time-consuming nature of the task (which could take hours unlike a standard survey), we could not retain the six. Eventually, we ended up with three analysts who carried out the task. In addition, the “subjective” aspect of the affordance construct rendered the process of explaining the concept to analysts who wanted more precise answers not provided by the instruction manual (Appendix 1) difficult as we did not want to taint the user with any bias. Similarly, this so-called “subjective” aspect, would imply that the inter-rater agreement among analysts on most affordances would be less than the ideal (i.e., less than 100%).

Indeed, this expectation was borne out in a smaller final set of extracted affordances—54%-85% more or less.

A note caution in terms of reviews is important to highlight: it mainly covered the Nokia phone as a product, and as such, the results here could be replicated via similar phones.

C. Future research directions

The framework provided by this thesis is an aggregation of steps taken to better understand and extract affordances within text in the context of Nokia phone reviews. With the availability of the affordance annotation application, affordance datasets can be obtained for any product and given more time and resources can be larger than the dataset used in this research. Although the thesis may be primarily a pattern recognition work of scholarship, the affordance annotation application software can be used for topics outside the realm of analytics, which implies the application software can prove of value in other research domains.

On the other hand, more research can be done on the same set but with a more focused view on automating affordance classification and detection. One of our other attempted approaches involved a neural network (Appendix 6) for classifying affordances in text. The results were mainly exploratory and were merely reported as side-note. Nevertheless, we managed to show that the affordance dataset can be utilized for such an approach, which can be pursued in future research.

We took a preliminary step, and we are aware that the proposed model of identifying if a text statement may contain an affordance is meant to be further expanded on and elaborated. The simple part-of-speech text processing approach can be developed into a more robust set of rules or routines that can reliably identify an affordance within text. This thesis is a small building block in the modeling of text affordances and, with future efforts, can serve as a stepping stone for a comprehensive and coherent affordance identification ruleset.

APPENDIX

Appendix I:

Affordance annotation application instructions

What is an affordance?

How would someone use a chair?

You can use it to sit.

Or as clothes rack

And in certain scenarios, you can use it:

To change a broken light bulb,

or to escape some pesky critters

All these actions **afforded** by this chair that you instinctively pick up on are called **Affordances**.

Affordances can be context-sensitive. For example, if the light breaks, you can use this chair as a step ladder to reach the broken light and fix it. The chair afforded you the ability to reach said light.

Let us explore a few more examples:

First let's consider a stick, for a human, it can be used as:

A walking cane for the blind or elderly or take better selfies for your Instagram fans

But for a hungry monkey, it would use it to pick up ants from their nest:

Now let’s look at a pen:

In your day to day life, you would probably use it to write:

But when assaulted mid-way through writing your award-winning novel to-be by an armed robber, it turns into a tool of self-defense:

Another example where affordances are taken into the design of objects:

Here are two doors.

You can immediately recognize how to interact with both due to the shape of the handle without ever being told how to it.

The handle design, along with your life experiences and knowledge are what allowed you to figure it out.

This is the affordance of these doors. The left one affords you to **push** it while the right is meant to be **pulled**.

So, we can summarize by saying that affordances are how a certain **feature** of a product be of any use to **you specifically** in a specific **context**.

This was a quick explanation of affordances. Below is a detailed explanation of the term.

More about affordances:

"The term affordance refers to the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used... Affordances provide strong cues to the operation of things."

-Don Norman.

An affordance, in the simplest terms, is what the object in question allows the user to do without any explicit instructions on how to do so.

When looking for affordances in the reviews, try to keep an eye for statements like “allows me to”, “lets me do”, “can be used to” and others along the lines of allowing the user to achieve a certain objective. Although these statements help in the identification of affordances, they are not a requirement as users can express affordances in a variety of ways.

Example:

“So now the Sonos One and Alexa is working. I can ask it questions and add stuff to my shopping list. Yay. The range at which Alexa can hear you clearly is something like 15-20 ft around a corner and 25-30 ft if you're unobstructed. Any more than that and you'll run into issues.”

From: https://www.amazon.com/gp/customer-reviews/R8R0W93SEJAKP/ref=cm_cr_dp_d_rvw_ttl?ie=UTF8&ASIN=B074XLMYY5

In the example above, we can see that the user uses the Alexa feature of the product to add items to his/her shopping list. This is not a built-in function of the product, and not all owners of the product would use it in this manner. Thus, the Sonos One affords this user to add items to their shopping list.

Sometimes we get a general affordance that can be formalized into a specific term:

“The sound profile of Sonos One and Play:1 are extremely similar. However, visually the products look different, and nearly all of the internal components of Sonos One have been redesigned from the ground-up to create a smart speaker that can be natively controlled by your voice.”

In this example, the user states that the product can be controlled by voice. In this case, we can formalize the affordance into a single term such as “controllability”.

Note that **not all** affordances can or should be formalized.

Using the app:

Main page:

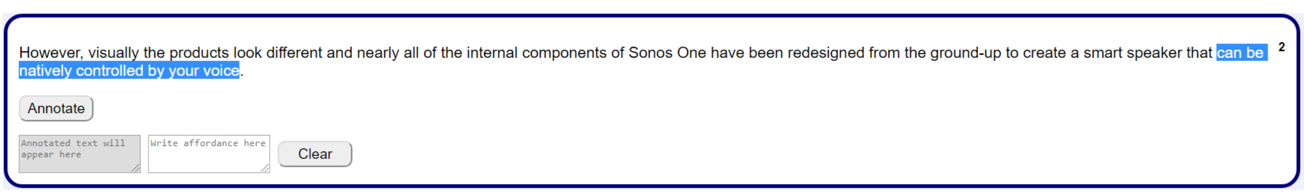
Upon accessing the main page, you will be asked to input the username and password that were provided. After this step you should arrive at this page:

Each box represents a separate sentence that may or may not contain an affordance.

In case there are no affordances, you can leave the fields empty and move on to the next box/sentence.

Affordance selection:

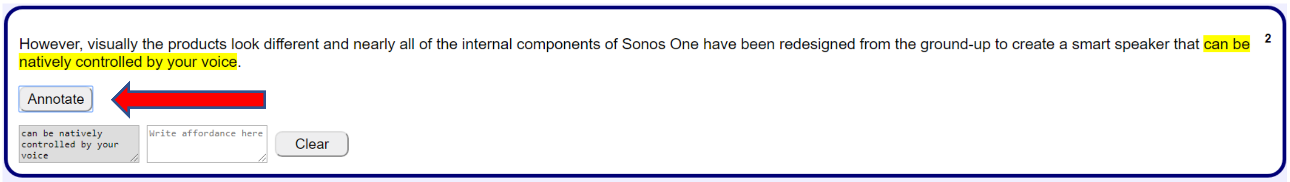
If you managed to find an affordance in the sentence, you highlight the set of words that form the affordance using the mouse pointer. The image below shows this in action:



Buttons and text boxes:

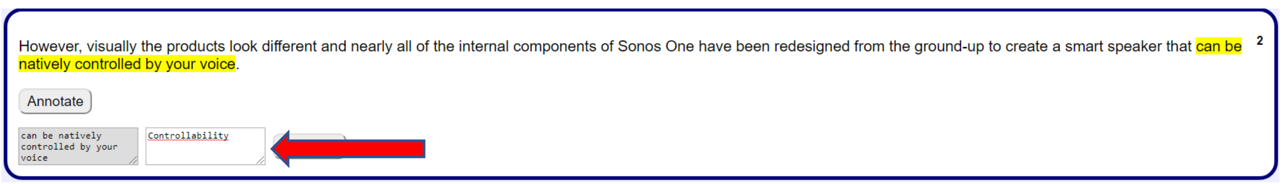
After highlighting the desired affordance text, you must click on the “Annotate” button within that same sentence box to confirm the highlighted selection as an affordance. Upon doing this, the selected text will be highlighted in yellow and that same text will appear in the grey box bellow the button.

Example:



If the affordance can be formalized as a single word, you should write it in the white box next to the grey one that replicates the highlighted text. If not, the box is to be left empty.

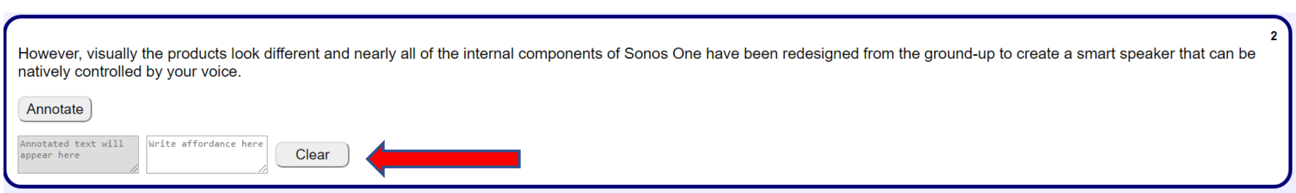
Example:



Note that both boxes can be expanded to see their full content by clicking the lower left corner and dragging to the desired size.

In case you made a wrong selection, you can reset the entire content of a specific sentence box by clicking the clear button. Note that clear will return the box to its original state and so all selections, highlights and written affordances will be removed from the specified box only.

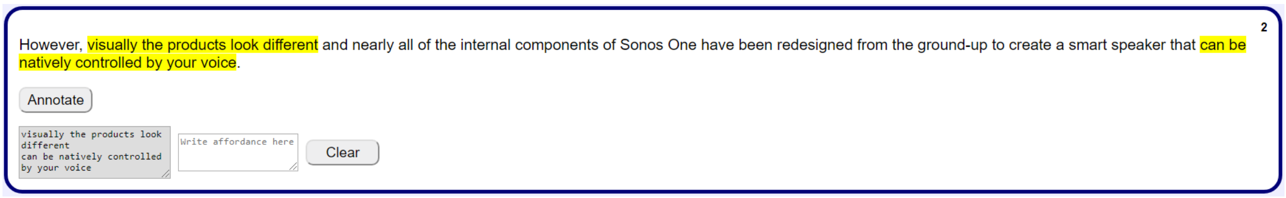
Example:



Special case: Multiple affordances in a sentence:

In some cases, there may be more than one affordance in a single sentence. If you happened to find one such example, you can redo highlighting process for that same sentence box while making sure that the two selections do not overlap.

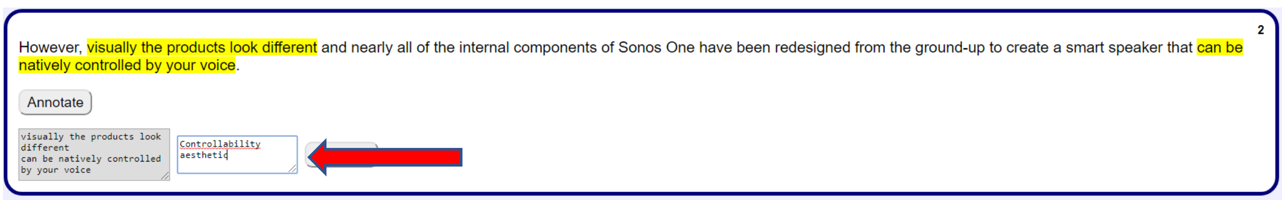
Example:



If you can formalize both affordances, you must write the first term, return to a new line by pressing enter and write the second term and so on.

Note the selection here is for demonstration purposes and is not a proper affordance.

Example:



Notice how the second highlighted in the sentence element appears first in grey box. This is because it was the first to be selected and annotated in this example. And as such the written affordance "controllability" goes first in the box to maintain the order and association between the two.

As recap, do not overlap your selections when handling multiple affordances per sentence and make sure to maintain the order between the selection box and the affordance writing box.

Missing affordances:

Some users may express that a product or a feature did not fulfil what's expected it of it, in that case we call this a missing affordance and should be highlighted the same as a normal affordance.

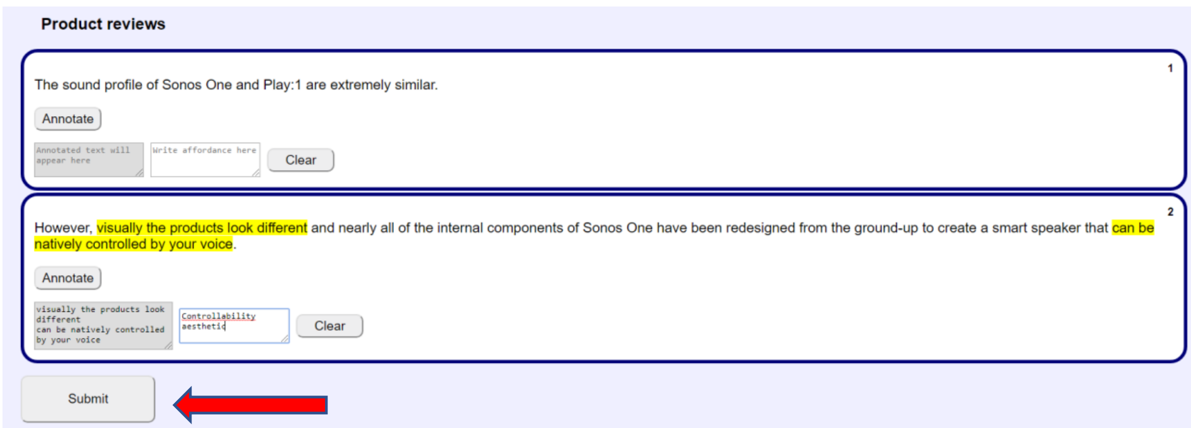
Example:

"Controlling it by voice is great and all but I thought I could also do it by phone because I don't want to repeat the same thing 20 time for Alexa to finally get it.

Submitting:

When you are done with all the selections for the available boxes, click the "submit" button at the bottom of the page and you will submit your current set of sentences with their annotations and receive a new set of sentences to work with.

<u>Example:</u>



When no new sentences appear after submitting, that means you went through the entire list of available sentences and have completed your task. Congratulations.

Appendix II:

List of part of speech tag definition[11] first found in chapter 4.1.b:

CC coordinating conjunction

CD cardinal digit

DT determiner

EX existential there (like: "there is" ... think of it like "there exists")

FW foreign word

IN preposition/subordinating conjunction

JJ adjective 'big'

JJR adjective, comparative 'bigger'

JJS adjective, superlative 'biggest'

LS list marker 1)

MD modal could, will

NN noun, singular 'desk'

NNS noun plural 'desks'

NNP proper noun, singular 'Harrison'

NNPS proper noun, plural 'Americans'

PDT predeterminer 'all the kids'

POS possessive ending parent's

PRP personal pronoun I, he, she

PRP$ possessive pronoun my, his, hers

RB adverb very, silently,

RBR adverb, comparative better

RBS adverb, superlative best

[11] tag definition taken from : https://pythonprogramming.net/natural-language-toolkit-nltk-part-speech-tagging/

RP particle give up

TO to go 'to' the store.

UH interjection errrrrrrrm

VB verb, base form take

VBD verb, past tense took

VBG verb, gerund/present participle taking

VBN verb, past participle taken

VBP verb, sing. present, non-3d take

VBZ verb, 3rd person sing. present takes

WDT wh-determiner which

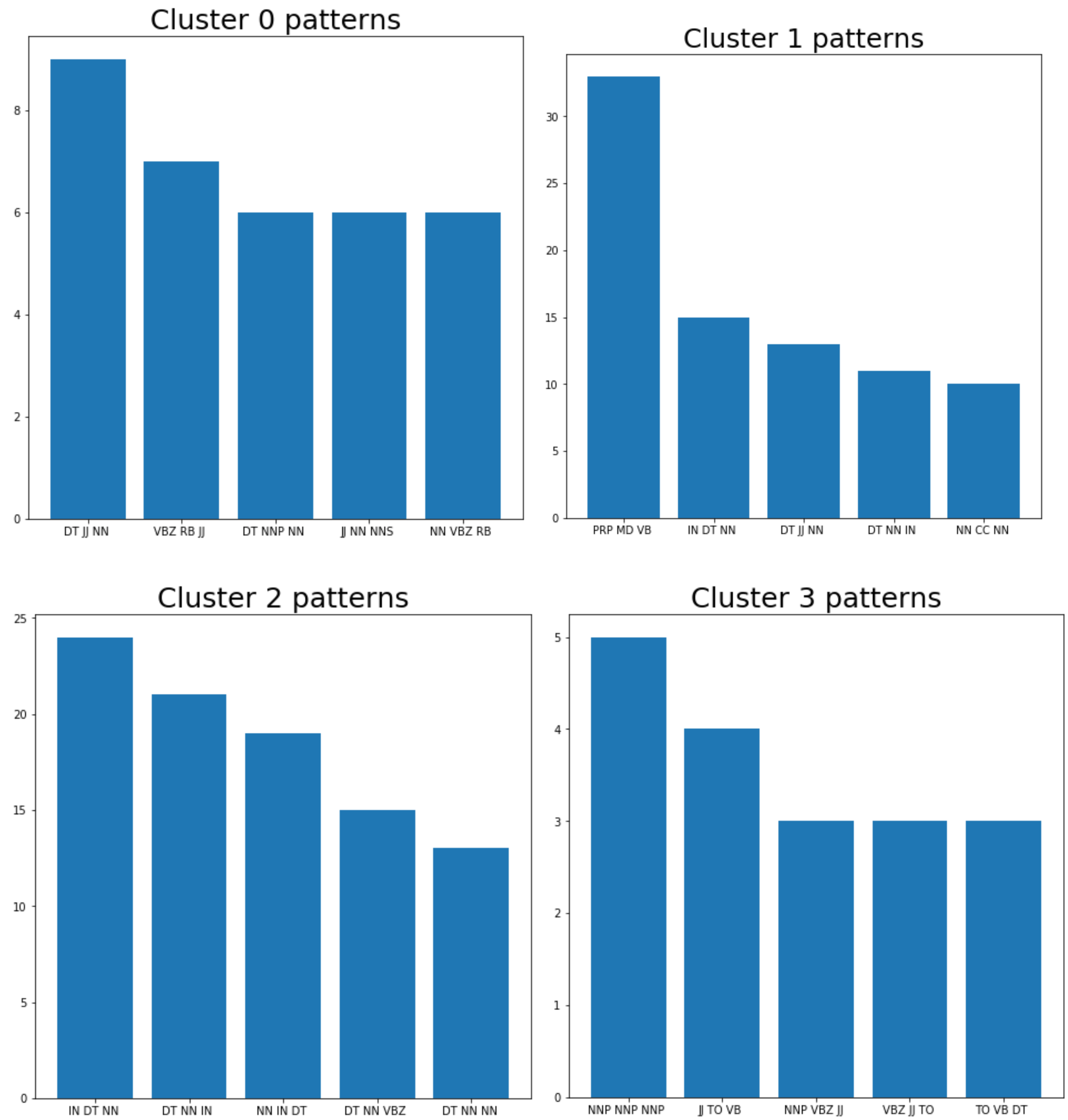
WP wh-pronoun who, what

WP$ possessive wh-pronoun whose

WRB wh-abverb where, when

Appendix III:

Cluster based pattern recognition charts:



Appendix IV:

Find below the link to the affordance annotation application along with the final state of the database from which the data for this thesis was extracted:

https://github.com/Remi115/affordance-annotation.git

Appendix V:

The review database

The following section will detail the design of the multiple databases built to store and handle the dataset chosen for this experiment. The primary purpose of this database was to query and organize the data obtained from the amazon S3 website, both the set for the products and the reviews, which would eventually lead to the 500 chosen reviews to be used for the core experiment.

The dataset obtained from the amazon S3 was originally two different sets, one containing reviews of products that are coded, and the other has the products with several details along with the appropriate code used in the review set. Both were queried to obtain the final dataset of 500 reviews mentioned earlier.

The database was named “metadatareview” and housed two tables, the review table which was appropriately called “review” and the product/item table called “item.” The database was built locally using wamp server and phpMyAdmin. The fields used in the tables are a curated and edited version of the original set obtained from the amazon s3 website. Figure 3.2 shows the database schema that will be further explained below.

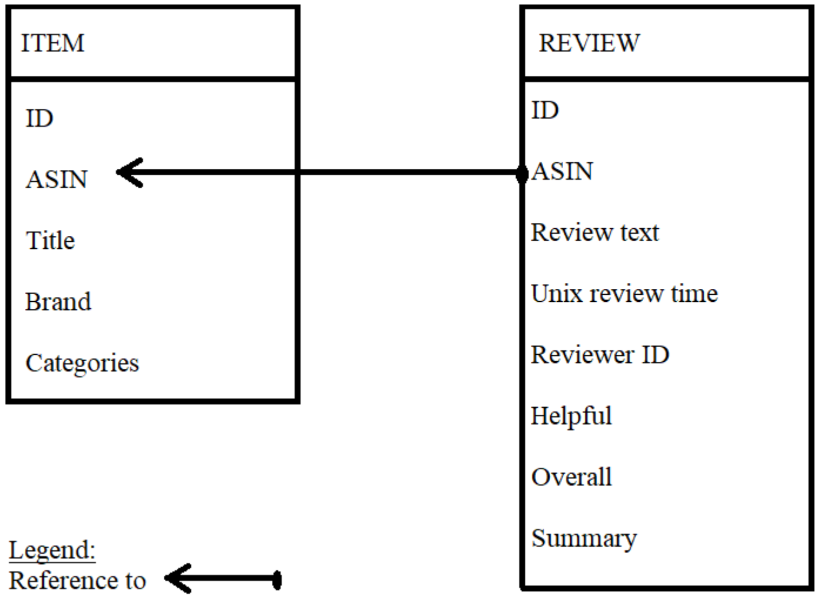


Figure 3.2 The metadata review database structure

1. The item table

The item table would see itself host a total of 4,715,044 rows, each corresponding to a unique item in the Amazon store and would have the following fields:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus, the rows are ranked by the order in which they were inserted.
- Asin: the code provided by the original set to denote the item in different sets. In our case, that would be the review set. The code is a combination of letters and numbers.
- Title: the name of the product in question.
- Brand: the brand the product falls under.

- Categories: an array detailing all the categories the product falls under (example: "Sports & Outdoors, Other Sports, Dance, Clothing, Girls, Skirts" are all categories for a single product and act as a single entry in the category field)

The following is an example of a record in the item database:

- ID: 4707212
- Asin: B00KQW6AB0
- Title: Nokia Lumia 630 Black Factory Unlocked
- Brand: Nokia
- Categories: Cell Phones & Accessories, Cell Phones, Unlocked Cell Phones

2. The review table

On the other hand, we would see the review table boast a mighty total of 41,338,564 rows each, corresponding to a review for any of the products from the previous table. The review table has the following fields:

- ID: an id that is autogenerated by the local database used in this experiment. The ID is set to autoincrement. Thus, the rows are ranked by the order in which they were inserted.
- Asin: the code provided by the original set to denote the item in different sets, in this case, that would refer back to the product list mentioned previously. The code, like before, is a combination of letters and numbers.
- ReviewText: The main content of the review.
- UnixReviewTime: a Unix value of the original time the review was posted.

- ReviewerID: a unique ID for each user that is determined by amazon.
- Helpful: a field used to count the total number of users that rated the product as helpful and takes the format of two numbers separated by a colon. For example, a value of 5:8 would mean that out of eight ratings, five were helpful, while three were not.
- Overall: a rating that is given to the product by the reviewer and takes a value from 1 to 5.
- Summary: the headline or title of the review that is seen on the website.

An example of a record in this table would be:

- ID: 81188635
- Asin: 0000000116
- ReviewText: Interesting Grisham tale of a lawyer that takes millions of dollars from his firm after faking his own death. Grisham usually is able to hook his readers early and, in this case, doesn't play his hand too soon. The usually reliable Frank Mueller makes this story even an even better bet on Audiobook.
- UnixReviewTime: 1019865600
- ReviewerID: AH2L9G3DQHHAJ
- Helpful: 5:5
- Overall: 4.0
- Summary: Show me the money!

Appendix VI:

Data analysis techniques

1. Clustering

Of the other approaches taken was the attempt to cluster the affordances using a combination of term frequency-inverse document frequency (TF-IDF). The goal here is to pinpoint the terms that hold value in the collection of affordances and using the result of this analysis in a k-means clustering process to see if the algorithm can provide any additional insights.

<u>a. TF-IDF for key term extraction</u>

In order to be able to cluster the affordances, we first need to extract the relevant and important terms from the affordances that will act as the main input for the clustering using later on. To accomplish this task, we would use the term frequency-inverse document frequency[12] weight to define these terms. In short, the TF-IDF weighing process puts weight on how important a word is for the individual document (the affordance in our case) when compared to the whole set of available documents or corpus.

When approaching the TF-IDF process, we were faced with mixed results as the default parameters provided accounted for large documents, and yet our sentences average

[12] More details on TF-IDF : http://www.tfidf.com/

out at around 15 words per record, which does not align with the original purpose. To that end, we proceeded to tweak the parameter until we got tangible results. The algorithm was fed a snowball stemmer[13] and the English stopword list along with the base untampered affordance text, then the part of speech tags, and return the key terms in the affordance set.

b. K-means clustering

After obtaining the key terms form the TF-IDF process, we use these terms as the features for the next step, which clustering. K-means clustering[14] would be the algorithm chosen for this task. The k-means clustering algorithm, in short, works by first specifying the number of clusters the algorithm should return, and then based on the number of times our feature terms have been mentioned, the algorithm will place each affordance within a cluster.

With the feature terms above extracted from the text, we run the k-means clustering algorithm on the result set of the previous set. After some experimentation, we decided to have the algorithm cluster the terms into 4 clusters. This process is also, like in the TF-IDF case, done on both the affordance text and the part-of-speech tags separately.

2. Clustering process results

---

[13] Stemmer details and algorithm: http://snowball.tartarus.org/algorithms/porter/stemmer.html
[14] More detail on k-means: https://www.experfy.com/blog/k-means-clustering-in-text-data

The first step in the clustering process was using the TF-IDF technique to pinpoint the key terms in the set of affordances and then the key tags in the set of pos tags of the affordance text.

The key terms for affordances were:

App, camera, game, great, internet, keyboard, like, lot, music, Nokia, phone, pictur(the stemmed version of picture), play, screen, use, video, work.

These key terms can help us understand or gain some insight regarding user perception of the phone and the features they deem to be most important. This can help us narrow down what the topic of affordances would most likely be when looking for Nokia phone affordances.

Using these terms as the features of the clustering algorithm we would, after some tinkering, decide to set the output to 4 clusters and the results were as follows:

- Cluster 0 count: 34
- Cluster 1 count: 17
- Cluster 2 count: 158
- Cluster 3 count: 10

And the top occurring terms for each cluster are:

- Cluster 0 words: 'phone', 'use', 'apps', 'music', 'great', 'lot'
- Cluster 1 words: 'screen', 'keyboard', 'like', 'game', 'use', 'phone'
- Cluster 2 words: 'camera', 'apps', 'pictures', 'music', 'use', 'video'

- Cluster 3 words: 'internet', 'phone', 'use', 'like', 'work', 'game'

Running the algorithm multiple time would always yield similar results with most affordances gravitating towards the cluster with the picture term.

Looking at the terms per cluster, we notice that each one tends to have some specific theme even though there is some degree of overlap.

Starting with the biggest cluster, cluster 2, the running theme seems to be linked to the standard phone applications. Still, more veered towards the camera function with the presence of camera video and picture in the mix. It hence may explain the sheer size of the cluster and give insight into what users mainly expect from a Nokia phone, which is a good camera functionality in this case. Examples from his cluster include:

- “ the camera does a pretty good job of guessing the best conditions”
- “ The camera is amazing and takes fantastic pictures.”

As for the rest of the clusters, due to the small number of reviews they hosted, we will cover them briefly:

Cluster 1 seems to focus on the hardware and the gaming aspect with the presence of “game”, “screen” and “keyboard”.

Cluster 3 focuses on a mix of work, games, and is the only cluster to reference the internet.

Cluster 0 seems to be similar to cluster 2 but has more positive connotations with “great” being a recurring term.

In conclusion, the clustering process provided a result specific to phone affordances rather than general affordances. The cluster formed would see their elements converge towards a similar feature of the phone to which affordance related to. We managed to note that most phone affordances associated with the standard phone apps such as music, apps, camera, and so forth. With this, we can say that a clustering approach on affordances can help specify the features that are most commonly used and discussed by users in reviews. As well as get a general feel of what the topic of the affordances would revolve around.

The cluster-based pattern recognition would yield less meaningful results, which can be found in appendix 3. We notice in the results that the top 2 sequences PRP MD VB and IN DT NN from the full set take over clusters 1 and 2, respectively. The fourth-ranking sequence dominates the cluster 0 albeit with low counts, while cluster 3 stands as an outlier with a unique sequence but has lower counts on all the sequences. From here, we can speculate that clustering will aggregate the different affordance types together, which might help if the goal is to locate a unique set of tags that might constitute an affordance.

3. Predictive convoluted neural network

Another approach we would take in our analysis is the use of a convoluted neural network to see if it is possible to automate the detection of affordances or predict their presence in a sentence. A convolutional neural network[15] (or CNN for short) is an artificial intelligence algorithm that intends to simulate the human brain in terms of neurons firing

[15]More on CNN: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

and was initially used for image recognition where the algorithm would give a weight to certain aspects of the image, and over several iterations called layers, would boil down to a single result. We intend to use the same technique on all our review sentences from the set of 2000 in order to have the algorithm be able to label them as containing an affordance or not.

We use Tensor flow in this experiment and use a pre-trained text embedding layer to simplify the conversion of sentences into embedding vectors. Embedding vectors[16] are, to keep it brief, a more streamline representation of text in the form of numbers that would allow the neural network to process it. The embedding model used here is found on TensorFlow hub[17], which is a library of modules meant to be reused with the intent of going through the process of transfer learning and using algorithms that already were trained to save on data size and time. The model we used can be found at the following link google/tf2-preview/gnews-swivel-20dim/1.

As we stated earlier, we would use the entire set of review sentences, both those containing affordances and those that do not. Using the annotation application database, we can figure out which sentences had affordance and thus make a new dataset that would have two columns. The first would house all the reviews, and the second would be a binary value (0 or 1) that would specify if the sentence contains an affordance, one if an affordance is present and 0 otherwise.

[16] Embedding vectors : https://towardsdatascience.com/neural-network-embeddings-explained-4d028e6f0526
[17] TensorFlow hub: https://www.tensorflow.org/hub

Using the above set, we would create two new sets. The first set would have sentences that have affordances, and the other would have the rest of the sentences. These two sets are further broken down into training, testing and validation dataset that consists of 60%, 20% and 20% of their respective sets (those with and without affordance respectfully).

With the data split, we would then run it through a simple convoluted neural network whose layers consisted of the following:

- The embedding layer
- A dense layer that uses the relu activation method
- Another dense layer that uses the sigmoid activation method

Running the neural network, we obtained an accuracy of 0.909 and a loss of 0.299.

4. Neural network results

The neural network experiment serves as an example that, once an adequate number of sentences containing affordances is collected, a neural network can be employed in order to automate the affordance detection process. The experiment yielded a 90% accuracy rate which serves as the driving argument behind our belief that such a model can be further elaborated on.

Unlike the manual detection process undergone by the research regarding the Canadian tire corpus by Chou and Shu (Chou & Shu, 2014), this process does not require any direct interaction and does build on the idea they offered regarding automating the system for affordance detection using a more mechanized approach for recognizing patterns.

Though the neural network we build in this experiment will classify a sentence as containing an affordance or not, it is only done on the raw original sentences, both those containing affordances and those that do not. We specify which sentences do contain affordances for the deep learning algorithm to train it. The same process is not done on any part of speech tags, which we assume might yield a different set of insight on how to approach affordances. Given that the model managed to provide results regarding simple text with an accuracy of 90%, we believe that an approach regarding pos tags would lead to similar results, which are also the belief held by Chou and Shu, as stated earlier.

The convoluted neural network built for this experiment was a simple one. Still, as stated earlier, its primary purpose was to demonstrate that affordance classification can be automated with the help of a dataset of affordance, which is now more easily accessible using the affordance annotation application.

REFERENCES

- Chou, A., & Shu, L. H. (2014). Towards extracting affordances from online consumer product reviews. *ASME Paper No. DETC2014-35288*.

- Faraj, S., & Azad, B. (2012). The materiality of technology: An affordance perspective. *Materiality and organizing: Social interaction in a technological world*, *237*, 258.

- Gibson, J. J. (1977). The concept of affordances. *Perceiving, acting, and knowing*, *1*.

- Jones, K. S. (2003). What is an affordance?, *Ecological psychology*, *15*(2), 107-114.

- Karat, John & Karat, Clare-marie & Ukelson, Jacob. (2000). Affordances, Motivation and the Design of User Interfaces. Communications of The ACM - CACM.

- LEE, C. F., CHEN, L. H., & YOU, M. (2009). Framework of product affordances and perceptual information. *Bulletin of Japanese Society for the Science of Design*, *56*(1), 93-100.

- Maier, J. R., & Fadel, G. M. (2007). Identifying Affordances. In *DS 42: Proceedings of ICED 2007, the 16th International Conference on Engineering Design, Paris, France, 28.-31.07. 2007* (pp. 841-842).

- Majchrzak, A., Faraj, S., Kane, G. C., & Azad, B. (2013). The contradictory influence of social media affordances on online communal knowledge sharing. *Journal of Computer-Mediated Communication*, *19*(1), 38-55.

- Nguyen, A., Kanoulas, D., Caldwell, D. G., & Tsagarakis, N. G. (2016, October). Detecting object affordances with convolutional neural networks. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 2765-2770). IEEE.

- Pucillo, F., & Cascini, G. (2014). A framework for user experience, needs and affordances. *Design Studies*, *35*(2), 160-179.

- Steedman, M. (2002). Plans, affordances, and combinatory grammar. *Linguistics and Philosophy*, *25*(5-6), 723-753.