Extracting products' qualitative properties from user reviews: the U.S. automotive market case.

Capita Selecta report - Supervisor: Slinger Jansen
Micaela Garcia Parente, student number 5951976

1 Introduction

In many industries, qualitative properties of a product play a significant role in users' buying decisions, and in their subsequent satisfaction with it. As an example, when searching for a new car, customers often take into account aspects like family-friendliness, appearance, and comfort aside from technical features.

However, when compared to the latter, qualitative properties are often harder to be retrieved and indexed by online retailers. One major factor involved is that such properties are highly associated with stakeholders' perceptions, rather than manufacturer's product descriptions. Therefore, users often rely on external sources to inform their decision process. According to Smith and Anderson (2016), 82% of U.S. adults at least sometimes read online reviews before purchasing items for the first time.

A systematic way to identify qualitative properties of products is expected to produce benefits for both online retailers and users, by offering means to increase the efficiency of the purchase process. The present research aims at contributing to this problem by focusing on the case of the U.S. automotive industry. More specifically, Natural Language Processing (NLP) tools are applied on user reviews to answer the following research question:

> *How can qualitative properties of passenger vehicles be identified so that retailers can provide a more efficient decision system for their customers?*

Current alternatives typically let users specify the desired technical functionalities (such as transmission type, number of doors, etc), whereas more insightful input can only be obtained by potential customers from other sources, like reviews and/or specialized articles. Therefore, letting users directly search automotive alternatives by non-technical needs represents a relevant business opportunity.

As an example use case, consider the fictional case of a single young professional who wants to buy a car to commute to work everyday. She is mainly searching for a car that fits her budget, but an attractive exterior would also be interesting. She is not familiar with car models, so her main sources of information would be online articles or friends and family.

In this case, a possible user flow on the retailer website would be:

1. Choose from relevant tags that fit her situation. Ex: "Commute car", "Attractive design", "Fuel-efficient".
2. Learn about the options that fit her criteria.
3. Refine search by adding more tags or filtering the results.
4. Go to dealer website.

The main data mining challenge is to extract and to structure such tags from car-related data, so that in the future they can be combined with traditional functional specifications already available to provide a better car search for users. In this research project, the focus is on extracting such qualitative aspects specifically from user reviews.

2 Related work

A few studies were conducted to extract knowledge from online product reviews. Hu and Liu (2004) used machine learning to mine customer reviews and provide a summary of positive and negative feedback according to product features explicitly mentioned by users. An alternative method was presented by Popescu and Etzioni (2005) with higher precision. Poria et al (2014) contributed to the theme by proposing a rule-based approach that also supports implicit aspects of the reviews.

Finally, Moghaddam (2013) proposed a novel technique to opinion mining from online reviews, which uses bag-of-opinion phrases models instead of bag-of-words topic models. According to their conclusions, the first model outperforms the second and represents a more efficient way to analyse reviews.

This research will take advantage of Moghaddam's work by incorporating opinion phrases (instead of raw sentences) as part of the modeling phase.

3 Dataset

User reviews were obtained from the american online portal Edmunds, by using a web crawler[1]. The resulting dataset consists of 8.749 user reviews about 489 car models, manufactured in 2016 and 2017. Table 1 presents some examples of reviews present in the dataset.

Table 1: Examples of user reviews from dataset

Car	**User review**
Brand: Honda Model: Fit Year: 2016	"I love my cute new car. I've only owned it for a week, but it's been great. Easy to drive, comfortable, plenty of storage despite its small size."
Brand: Honda Model: Pilot Year: 2017	"Sporty, spacious, good fuel economy. It is a great family car."
Brand: Lexus Model: ES300H Year: 2016	"Overall the interior is just elegant and beautiful. I would definitely recommend this car to anyone who is looking to buy a reasonably priced, entry-level luxury sedan that combines reasonable performance and gas efficiency."

4 System

This is a Natural Language Processing (NLP) problem that requires analyzing substantial amounts of text towards a specific goal. There are a number of NLP toolkits available that provide functionalities like parsing, Part-of-Speech (POS) tagging and lemmatization, each of them with their own advantages and weaknesses.

This project was implemented in Python using a set of NLP tools. Figure 2 provides a general architecture of the system. Further details about each step and toolkits used will be presented in the succeeding subsections.

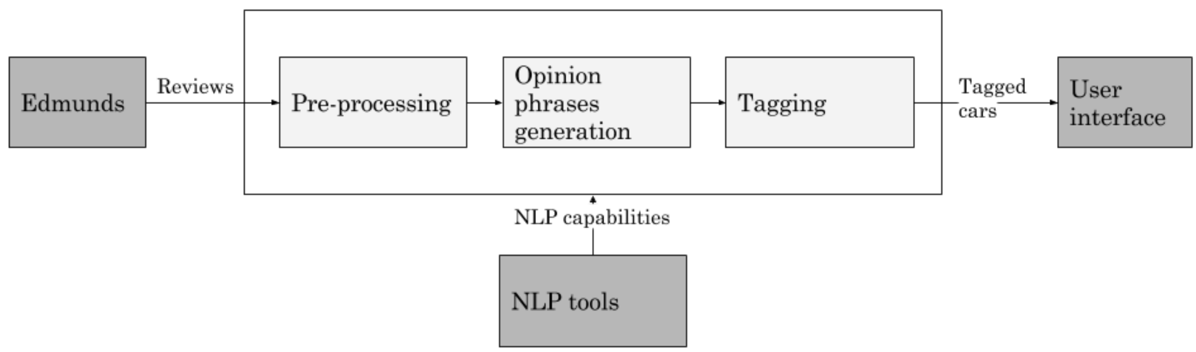

Figure 2: Functional architecture of the solution

[1] The "Beautiful Soup" library for Python (https://pypi.python.org/pypi/beautifulsoup4) was used to parse HTML pages.

4.1 Pre-processing

A number of steps need to be applied to pre-process the raw dataset before modeling. More specifically, sentence tokenization, POS tagging and dependency parsing were performed by using spaCy[2] toolkit.

In NLP, tokenization is the process of splitting a piece of text into chunks (called tokens) that can be characters, words, or sentences. In our case, sentence tokenization was used to split each user review into a set of sentences for later analysis.

As an example of sentence tokenization, the user review "Worst navigation system on the planet. It locates about half the destinations I've tried to input. The system makes no suggestions just says it can't locate the address." is split into three sentences. These sentences are then ready for POS tagging and dependency parsing.

Part-of-speech tagging refers the annotating words in terms of their part of speech, such as noun, adjective, adverb, etc. Together with dependency relations, these will be the core information needed for the opinion phrases generation part.

Dependency parsing refers to identifying grammatical relations in the text. The following list describes the grammatical relations that will be used for the opinion phrases generation (Marneffe & Manning, 2008).

- **Adjectival complement (acomp):** An adjectival phrase which functions as the complement, e.g. "The seats are comfy" parsed to acomp(are, comfy).
- **Adjectival modifier (amod):** An adjectival phrase that serves to modify the meaning of a noun phrase, e.g. "It is a family car" parsed to amod(car, family).
- **Nominal subject (nsubj):** A noun phrase which is the syntactic subject of a clause, e.g. "The seats are comfy" parsed to nsubj(comfy, seats).
- **Conjunct (conj):** A conjunct is the relation between two elements connected by a coordinating conjunction, such as "and", "or", etc, e.g. "The car is uncomfortable and noisy" parsed to conj(uncomfortable, noisy).
- **Direct object (dobj):** A noun phrase which is the object of the verb, e.g. "You can feel the power" parsed to dobj(feel, power).
- **Negation modifier (neg):** A relation between a negation word and the word it modifies, e.g. "The system is not intuitive" parsed to neg(intuitive, not).
- **Compound modifier (compound):** A noun that serves to modify the head noun, e.g. "The arm support is great" parsed to compound(support, arm).

4.2 Opinion phrases generation

As Moghaddam (2013) explains, opinion phrases are a pair *<h, m>* composed of a head term *h* and a modifier *m* that expresses some sort of opinion about the aspect. Some examples are *<interior, luxurious>* and *<car, compact>*.

Opinion phrases can be extracted by using dependency patterns between words found in a sentence. Based on Moghaddam (2012) work, the following dependency patterns were used to extract opinion phrases from the car user reviews.

Primary rules

1. amod(noun, adjective) → <noun, adjective>
2. acomp(verb, adjective) + nsubj(verb, noun) → <verb, adjective>
3. conj(adjective, verb) + nsubj(adjective, noun) → <noun, adjective>
4. dobj(verb, noun1) + nsubj(verb, noun2) → <noun1, verb>

Secondary rules

5. <head1, modifier> + conj(head1, head2) → <head2, modifier>
6. <head, modifier1> + conj(modifier1, modifier2) → <head, modifier2>
7. <head, modifier> + neg(modifier, not) → <head, not + modifier>*
8. <head, modifier> + compound(head, noun) → <noun + head, modifier>
9. <head, modifier> + compound(noun, head) → <head + noun, modifier>

[2] https://pypi.python.org/pypi/spacy

* If possible, the modifiers were converted into their antonyms, generating the pair <head, modifier antonym>.

Such rules were implemented using Pandas[3] library for data manipulation and NLTK's WordNet[4] interface for antonyms checking. Wordnet is a lexical database for English, from where it is possible to source conceptual-semantic and lexical relations between concepts (Princeton University, 2010). Table 2 presents an example of opinion phrases generated from a sentence present in a user review in the dataset.

Table 2: Example of opinion phrase generation for a user review

Sentence	**Dependency relations**	**Patterns**	**Opinion phrases**
"The cabin feels claustrophobic and small."	nsubj(feels, cabin); acomp(feels, claustrophobic); conj(claustrophobic, small)	2, 6	<cabin, claustrophobic> <cabin, small>

Implementing these rules result in an extensive set of opinion phrases, and raises the challenge of removing false opinion phrases, namely pairs containing non-aspects. Non-aspects are words that don't represent a meaningful aspect of the car, such as "thing", "problem", etc. A number of complex techniques were presented in the literature to deal with this problem, but for this project a list of common non-aspects observed was generated for cleaning.

Some examples of opinion phrases generated for cars in the dataset were:

- Honda Fit 2016: excellent performance, super roomy, comfortable seat, economical car, small commuter
- Porsche 911 2017: sexy car, instant response, outstanding performance, sporty car
- Chevrolet Tahoe 2016: great family car, good cargo space, expensive luxury car, large SUV, comfortable seat

4.3 Tagging

When analyzing the opinion phrases generated, it is possible to notice that there are many different phrases with similar meanings. Table 3 presents examples of different opinion phrases found in the dataset that are semantically similar.

Table 3: Semantically similar opinion phrases

Opinion phrases	**Common meaning**	**Possible tag**
daily trip, daily drive, daily commuter, commuter car, everyday commute car, everyday car	Car suitable for daily travelling, such as commuting to work.	Everyday car
comfortable interior, comfy seat, comfortable position, adjustable seat, enough interior room, ample leg room	Car with comfortable interior.	Comfortable car
luxurious car, classy leather interior, rich feel, rich look, solid luxury	Car with luxury-looking characteristics	Luxury car

[3] https://pypi.python.org/pypi/pandas
[4] http://www.nltk.org/howto/wordnet.html

The main goal of the tagging process is to use opinion phrases with close meaning to categorize cars according to their qualitative properties. This final step is expected to contribute to a user-centered car search, where potential car buyers can find vehicles based on their needs and preferences even if they do not have a solid technical understanding about cars.

Moghaddam (2013) used WordNet's shortest path distance between two words as an indicative of semantic closeness and a way to normalize similar aspects (in their work, if the shortest path distance between two words is 2 or less, these words are grouped together). However, normalizing different opinion phrases still remains a technical challenge, as these often comprehend pair of words that are not semantically close between themselves.

As an example, if the opinion phrases "daily drive" and "commuter car" are analysed, the following results are obtained through WordNet:

- The shortest path distance between "car" and "drive" is 16.
- The shortest path distance between "daily" and "commuter" is 10.

For comparison purposes, the shortest path distance between "car" and "tree" is 13. Therefore, the method proposed by Moghaddam (2013) is not applicable in this context. We choose to performing the tagging phase by establishing empirical rules based on opinion phrase content.

A semantic visualization of the most frequent heads and modifiers from the opinion phrases generated gives further insight on relevant qualitative properties frequently mentioned by users in the reviews. Figure 3 depicts a semantic tree built with TreeCloud (Gambette & Véronis, 2010) for the heads present in the opinion phrases mined. The different colors indicate relevant semantic groupings that served as input for the tagging process.

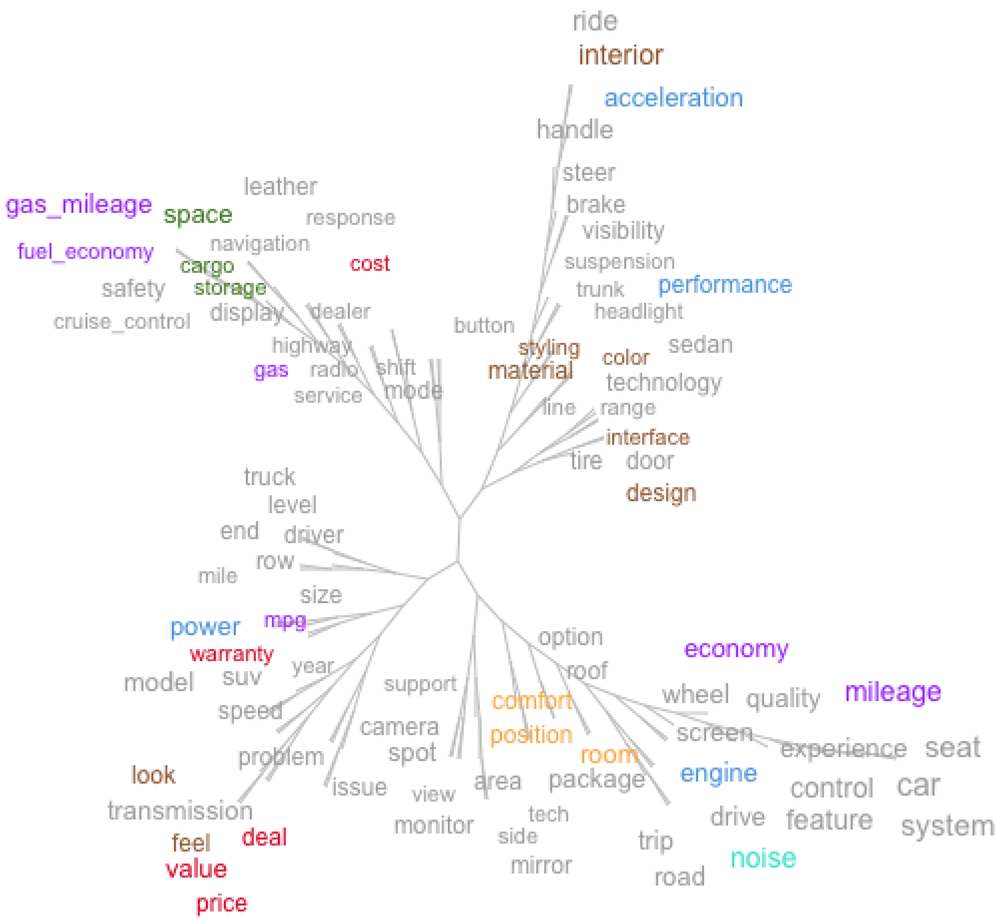

Figure 3: Semantic tree for opinion phrases heads

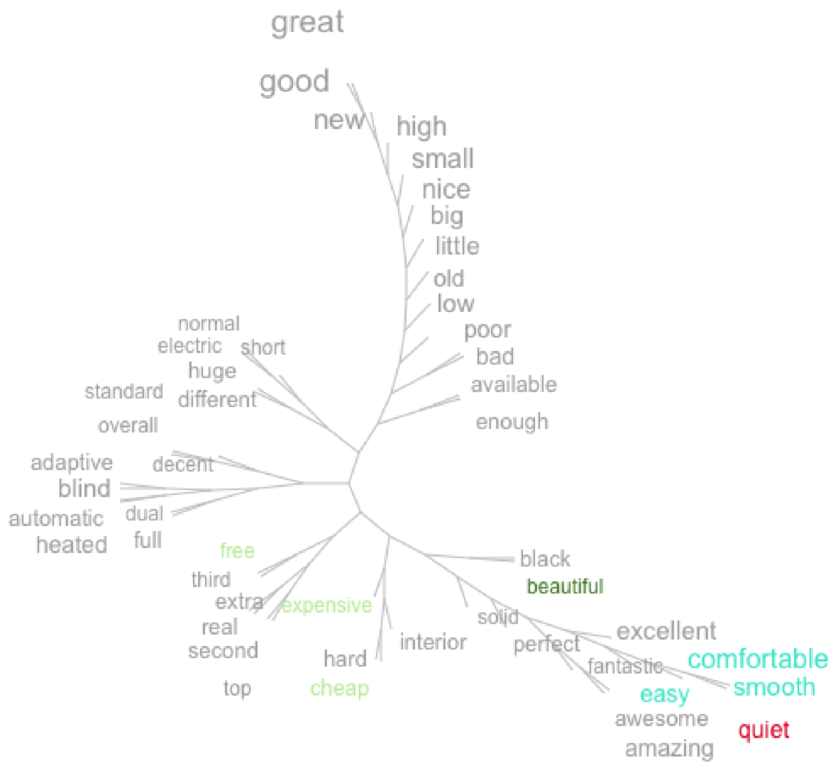

Figure 4: Semantic tree for opinion phrases modifiers

Based on the insights generated from the semantic visualizations, a list of tags was created, and words with similar meanings were associated to them, as documented in Appendix 8.1. This was used to tag the cars based on opinion phrases generated from their user reviews. After the tagging process, all the cars receive a score in each of the categories, ranging from 0 to 100%. This score is calculated based on the number of words for each category found in their opinion phrases, normalized by each car.

The tagged cards are finally displayed to the final user with a simple ranking algorithm that takes into account how they score in the categories selected by the user. Additionally, a threshold of 40% was defined, so that when the user selects one or more categories, they will not see as result any cars that score below this mark in the categories.

To illustrate this process, consider that cars 1 and 2 were profiled into the categories as Table X describes. In this situation, if the user selects only "Category 2", the application will first filter the cars with a score of at least 40% in this category, which yields cars 1 and 2. As Car 1 score is higher, it will be displayed first. When more than one category is selected, the product of scores in the selected categories is considered for ranking the cars.

Table 4: Example of tagged cars

	Category 1 (word 1, word 2...)	Category 2 (word 3, word 4...)	Category 3 (word 5, word 6…)
Car 1	56%	64%	20%
Car 2	10%	40%	50%

5 Results

The exploration of tagged cars mentioned in the last section is publicly available through an interactive prototype[5], developed using Shiny, a web application framework for R.

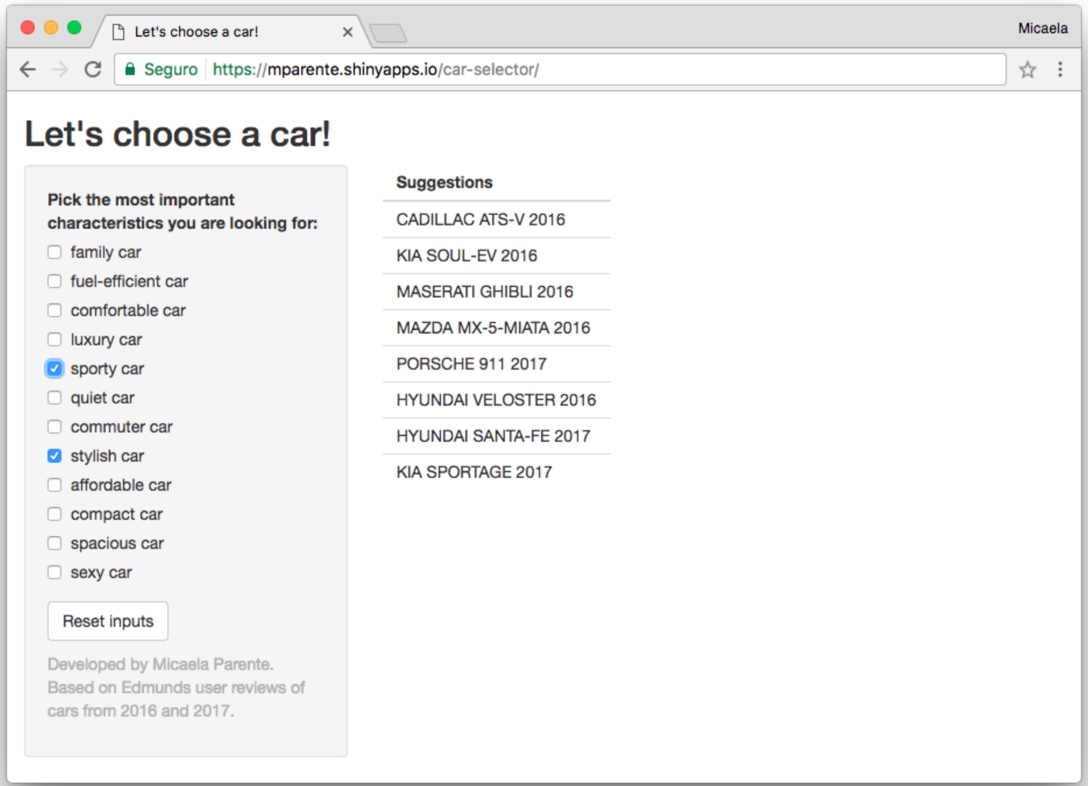

Figure 5: Car selector prototype

By choosing from the available tags, the user can explore car options that correspond to the selected labels, and further refine their purchase decision.

5.1 Examples

This section describes some example results obtained directly from the tool, based in different tag selection combinations.

Although the output is promising, there are still some limitations to be discussed. For instance, it was observed that cars from 2016 tend to rank better in the suggestions. This can be related to the fact that there are more reviews availables for these cars than for 2017 models, which also influences the existence of relevant opinion phrases generated.

[5] https://mparente.shinyapps.io/car-selector/

Figure 6: Top 3 results for “Family car” + “Fuel-efficient car”


Nissan Juke 2016


Nissan Pathfinder 2017

Ford Focus ST 2016

Figure 7: Results for “Commuter car” + “Compact car”


Fiat 500e 2016


Mazda CX-3 2017

Figure 8: Top 3 results for “Luxury car” + “Stylish car”


Cadillac XTS 2016

Volvo S60 2016


Lincoln MKZ 2017

5.2 Validation

As an initial validation, the tool was tested with one potential user with the following characteristics:

- Man, 25 years old
- Full-time professional
- Commutes 40km everyday to work
- 7 years of driver’s license

The user was first interviewed about his habits and needs. As it was reported by him, the car would be used daily for commuting, and occasionally for longer trips in the weekends. Additionally, it was also said that the car should not represent a large monthly cost. Therefore, the most important tags defined by the user, when looking at the prototype, were “fuel-efficient car”, “affordable car”, and “commuter car”.

After the user experimented the tool and analysed the car suggestions, the following feedback was provided:

- The tool is easy to understand and to use.
- The tags presented are comprehensive for his needs.
- Some tags sound overlapping, such as “commuter” and “compact”. It is not clear what separate them.
- After receiving the suggestions, the user was expecting to click on car names to see list of dealers and prices.
- After checking the suggestions separately, the user was favorable about the cars. According to him, being able to see the price would be determinant to help his decision. Experimenting with other tags was also mentioned.
- He would also like to see used cars included in the results.
- According to him, the exact year is not so important: “What is most interesting to me is knowing the best model to my needs. Usually the cars will not change much from one year to another. I would just search for the most affordable one, either used or new.”

The result of this initial validation raises interesting points for future researches, as the expansion of the dataset to include used cars and the validation of the tags proposed with experts and potential users. The next section outlines the main limitations of the research and relevant points for further studies.

6 Conclusion

The problem of extracting qualitative properties from user reviews represents a significant business opportunity in the automotive industry. In this context, this project made the following contributions: (i) the creation of a structured dataset of user reviews of new cars from 2016 and 2017, (ii) the application of NLP techniques and opinion phrases to the automotive industry, (iii) the visualization of common (qualitative) topics that emerge from user reviews about cars, (iv) the proposal of a car tagging process based on opinion phrases generated from the reviews, and (v) the development of an interactive prototype to enable the validation of this solution to the aforementioned problem.

As it emerged from the results and the initial validation conducted, there are still points that can be further explored by future studies. Firstly, the dataset can be enriched both horizontally (by joining car specifications such as price, year, dealers and brand nationality) and vertically (by also capturing data about used cars and older models). Secondly, it is advised that future studies conduct an extensive validation of the proposed solution with experts and potential customers. The domain knowledge gathered can help improving the existing tags, as well as giving insight on users purchase process.

Finally, once the data is enriched, the tagging model can be further refined to improve the current rule-based approach. Solutions like word embeddings might perform better on a bigger corpus of reviews, and functional specifications may also be tested as features to improve the tagging process.

7 References

Hu, M., & Liu, B. (2004, August). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 168-177). ACM.

Marneffe, M. & Manning, C. (2008, September). Stanford typed dependencies manual. Retrieved from https://nlp.stanford.edu/software/dependencies_manual.pdf

Moghaddam, S. (2013). Aspect-based opinion mining in online reviews (Doctoral dissertation, Applied Sciences: School of Computing Science).

Philippe Gambette & Jean Véronis - Visualising a Text with a Tree Cloud, In Locarek-Junge H. and Weihs C., editors, Classification as a Tool of Research, Proc. of IFCS'09 (11th Conference of the International Federation of Classification Societies), p. 561-570, 2010

Poria, S., Cambria, E., Ku, L. W., Gui, C., & Gelbukh, A. (2014, August). A rule-based approach to aspect extraction from product reviews. In *Proceedings of the Second Workshop on Natural Language Processing for Social Media* (pp 28-37).

Popescu, A. & Etizioni, O. (2005). Extracting product features and opinions from reviews. In Proceedings of Conference on Empirical Methods in Natural Language Processing (pp 3-28).

Princeton University (2010). “About WordNet”. Retrieved May 17, 2017 from https://wordnet.princeton.edu/.

Smith, A. & Anderson, M. (2016, December). Online shopping and e-commerce. Retrieved March 07, 2017 from http://www.pewinternet.org/2016/12/19/online-reviews/.

8 Appendix

8.1 Rules used for tagging

Tag	Opinion phrases contain:
Family car	Family
Fuel-efficient car	Efficient, good mileage, economic, great mileage, good economy, great economy, good efficiency, great efficiency
Comfortable car	Comfortable, comfort, adjustable, accommodating
Luxury car	Luxury, luxurious, rich, classy, premium
Sporty car	Sporty, peppy, sport
Quiet car	Quiet, silent
Commuter car	Commute, everyday
Stylish car	Great design, beautiful, modern, stylish
Affordable car	Good deal, good price, great price, great value, good value, affordable, inexpensive, cheap price, cheap car
Compact car	Compact, small car, little car, small size
Spacious car	Roomy, spacious, good space, excellent space, great space, big space, amazing space, large space
Sexy car	Sexy