COMP SCI 784: Stage-2

Amrita Roy Chowdhury	Ekta Sardana	Roney Michael
roychowdhur2@wisc.edu	ekta@wisc.edu	rmichael2@wisc.edu

In this report we describe in detail our work on developing a brand name extractor based on the sample data we were provided, for Stage-2 of the COMP SCI 784 project for the Spring 2016 semester. Overall we achieved **precision and recall measures** of 96.99% & 97.55% respectively on our validation runs on an average, and **99.05% & 96.30% respectively** on our held-aside test set. Detailed metrics are listed at the end of section II (*How it Works - Quantitative Measures*) of this report.

I. Measures

We explain our measures below with examples as applicable, where quoted, the examples are formatted as:

```
>>Product name
>>Actual brand
>>Predicted brand
```

- **Precision** is the ratio of true positives to total predicted positives. This implies:

$$Pr = TP/(TP + FP)$$

- **Recall** is the ratio of true positives to total actual positives. This implies:

$$Re = TP/(TP + FN)$$

- A **true positive** (TP) occurs when the brand name exists in the product name and the extracted (predicted) brand name is equal to the actual brand name.
  - Example:

```
>>Phottix Battery Grip for Nikon D40 D40X D60 D3000 With Vertical
Button
>>Phottix
>>Phottix
```

- A **false positive** (FP) occurs in following cases:

    - There is no brand name in the product name and it is predicted that the product has a brand name.
    - There exists a brand name in the product name but the predicted brand name is not equal to the actual brand name.
    - Example:

```
>>LG G3 Case - GreatShield [LEGACY] Slim Fit Dual Layer Hybrid Cover
with Kickstand for LG G3 (+Screen Shield) - Black
>>GreatShield
>>LG G3
```

- A **true negative** (TN) occurs when there is no brand name in the product name and it is also predicted that the product doesn't have a brand name.
    - Example:

```
>>Mobile Battery Pack USB Black
>>
>>
```

- A **false negative** (FN) occurs when there exists a brand name in the product name and it is predicted that the product doesn't have a brand name.
    - Example:

```
>>Bill Hicks & Co LTD SO PC900 LARGE PORTABLE CASE W/ ELECTRONIC LOCK
>>Bill Hicks & Co LTD
>>
```

II. How it Works

The Data

Before proceeding to write code, we tried to familiarize ourselves with the data we would be working with.

The data consisted of 10000 product pairs (20000 products) from which we randomly sampled 350 products without replacement. The brands for each of these 350 products for the golden data were then manually identified by the group, cross-verified, and then again randomly split into a test set 'I' and a development set 'J', of 120 and 230 instances respectively. We kept the test-set completely aside and further split the development set in each of our trial runs into training and validation sets of 170 and 60 respectively, based approximately on the 3-1 split commonly adopted in Machine Learning for training and development sets[1]. The training and validation sets would be used for code development while after we completely finalized the code, the two would be used together as a combined training set I when testing against the test set of 120 products in the development set J.

In addition to using the data as stated above, we also eyeballed the remaining 19650 in order to get a rough idea of how effective our heuristics might be. However, at no point in our development cycle did we open up the final test set; that was used solely for outputting the final metrics.

The Algorithm

Dictionary Matching

The core idea of our brand name extractor was string matching based on the provided dictionary of 8442 brand-names. As a first trial, we read in all the brand-names line by line (sorted by some notion of frequency as they were) and added the brand-names from our training data to it as well, with one additional frequency count per product.

We describe below, how we iteratively applied this approach:

[1] Lecture on "Model Selection and Train/Validation/Test Sets" by Andrew Ng, https://www.coursera.org/learn/machine-learning/lecture/QGKbr

- We first tried simple substring matching, terminating our search for each product in the validation set when a first match had been found. This however was quite ineffectual, particularly we discovered on debugging, due to the presence of a brand simply named “G” in the dictionary. Our precision and recall for this case was therefore terrible, being in the range of only about 50%.
- We started with the idea of tokenizing. We modified our sub-string search to only match dictionary entries at word boundaries in the product names. This helped us avoid matching random “G”s for the case stated above and brought up our performance metrics to approximately 70% each.
- We proceeded with the idea that it made sense to look for longest matching substrings instead of simply finding the first match. This was in conflict with the idea of matching the brand with the higher *frequency* in the dictionary, but empirically, it seemed to give better results. In order to improve of efficiency, we pre-sorted the list of brands by length, instead of simply iterating over the entire list each time.
- Partial matching of brand names was another factor to be considered. For example, If we were to identify *HUBBELL WIRING* as the brand name while the actual label in our golden data was *HUBBELL WIRING DEVICE-KELLEMS*, it would be considered a match, but *only if the identified match was already in the dictionary of brand names*. The last part should be of particular note, since this would otherwise consider incorrect matches to be true positives, which has been handled in our brand name extraction algorithm.

Regular Expressions

We realized quite early in our development cycle, that simple string matching was quite ineffective a tool for applying a dictionary based approach to this task. As such, we decided to explore the use of regular expressions to better form our match criterias.

The first place we employed regular expressions was in matching dictionary entries only at word boundaries. After validating this through a now improved performance on our validation run, we then decided to handle simple cases such as differences in spacing or where a hyphen was substituted for a whitespace, Tripp Lite vs. Tripp-Lite for example. We applied this to every brand name we had in our dictionary by maintaining a list of precompiled regular expressions, with one entry corresponding to each dictionary entry. These expressions would:

1. Make use of the `\b` escape sequence to limit matching to word boundaries; and
2. Match for any non-`\w` sequence instead with a sequence of 0-3 such characters.

After validating a small, but discernable improvement in performance due to this use, we decided to apply certain heuristics - coupled with regular expressions - we came up with, to further refine our results. These, along with the reasoning behind them, are explained in the following sub-section; we have taken into account both relative positions (indices) and style (prepositions, upper-/lower-case) in coming up with these.

Heuristics

The heuristics we developed are listed below with the reasoning behind them. Sample product names are quoted at the end of each case for additional clarity. Note that these heuristics are not mutually exclusive. In each example, the correct brand is noted in **bold** and the incorrect one in red.

1. The first heuristic we came up with was that of certain specific prepositions; we noticed that a number of our incorrect results were because our brand-name extractor being thrown off by brands of compatible products mentioned in the product name, such as "... ABC … for … XYZ ...", or "... ABC … compatible … with … XYZ …", where ABC denotes the position we observed for the product of note and XYZ, the product not of interest which ABC would suit.
   To overcome this challenge, we identified the first occurrence of either of the words "for" or "with", and then invalidated any previously found match if the match-index was not less than the index of the preposition match.
   We found this approach significantly brought down our identification of false positives. The `preposition_re` variable in our code caters to this specifically.

   > Universal Windshield Mount for iPhone Samsung Galaxy S and most Mobile Phones - Black

2. The next heuristic we came up with was that of `limitIndex`. Like in the case above, we noticed that we were outputting another distinct category of false positives - ones where the product in question did not in fact have a brand name listed or had a brand name of statistically lower precedence (length, frequency or otherwise) in the product name field, but where extraneous brand names of related/compatible were being identified instead; some of these cases were covered by the preposition check mentioned above, but a sizeable number was not. After revisiting some of the data we had left unlabelled, we identified a pattern that in most cases, these incorrect brand names occurred quite a bit after most of

the essential product info was already listed. From a small sample size of 10 such products we eyeballed where this seemed to be the case, we found 8 of them to have these terms after the 70th character and the remaining two to be after the 60th character. After accounting for some measure of a buffer, we decided to set the limit as 50; if the index at which our match began was beyond this limit, we would invalidate the match.

> **Sophia Global** Remanufactured Color Ink Cartridge Replacements for HP 901XL (Pack of 10)(Refurbished)

3. With these new heuristics in place, we found that it was no longer sufficient to terminate our search for brand names at the first instance a match had been found. We therefore continued our search through all names for each item, which though somewhat inefficient, seemed effective; we note that if our approach is to be applied to enterprise data, this will need to be cleverly worked around.
   With this change, however, we seemingly had a new problem, which was preemption - correct brand names were initially identified for a number of products, but they were later preempted by shorter incorrect matches. To overcome this problem, we came with an idea tangentially similar in essence to the previous heuristic; an index check for allowing preemption - allow for an already identified longer brand name to be preempted by a shorter one if and only if the match index of the new (shorter) one is smaller than that of the previous (longer) match. We found this to be extremely effective in cutting down false positives caused by preemption.

   > **Targus** 4vu Privacy Screen For 12.2 Samsung Galaxy Tab Note Pro Clear - 12.2tablet Pc (ast008usz)

4. Even though most products had the brand names stated as one of its first 3 words in the product name, we found that this was not the case quite often for a certain non-negligible section of products - those that began with the words “off lease refurbished”. We were initially concerned that accounting for this could end up being over fitting, but after grepping through the list of all product names, found this to be the case for 719 of the 20000 total products, roughly 3.6%, something we consider worth accounting for.
   Our method for accounting for this was by computing the match indices only after accounting for product names beginning with these words, taking into account any possible differences in spacing and punctuation. The `refurbished_re` regular expression serves this purpose.

   > Off Lease REFURBISHED **Dell Latitude** E6410 I5 2.4GHz 8GB 320GB DVD Windows 7 Pro64 Bit Laptop Notebook

5. We noticed a certain class of rare brand names with reasonable lengths of alternating upper and lower case alphabets. These were not very common and were therefore not present in our dictionary. As such, it was impossible to identify them by means of vanilla dictionary matching approaches. We adopted a heuristic wherein if we found the pattern `[A-Z]+[a-z]+-?[A-Z]+\\w+` in our product name, that it might be a brand name. This seemed to work reasonably, but in order to be safe from overfitting, we reduced the scope of this expression to only match the very first word in the product name if at all. We enforce this through the use of the `CaSe_re` expression in our code. We added in this check at the very beginning, choosing to add this in as a heuristic start value, which may be preempted by a longer dictionary match, by means of equality from the `<=` in our preemption condition.

   MacCase L15SL-VN 15 inch Premium Leather MacBook Pro Sleeve - Vintage

6. Another set of brand names were those of reasonably rare abbreviations which were not present in our dictionary. This was deployed as only a last resort match check when all previous measures had already failed. In order to identify such brands, when no match has been identified after all previous measures, we settle on a brand identified as the first word of the product name, if it consisted solely of uppercase alphabets and was at least five characters in length. We picked the heuristic limit five here to avoid the words “A”, “AN” and “THE”, as well as common technical abbreviations such as “CCTV” and HDMI”. Setting this conservative a limit meant that we would not be able to identify shorter brands such as “FIFA” or “LEGO” as well, but these were fairly rare enough to not be great concerns; accounting for these would risk overfitting and we therefore chose not to go down that route. Many major 4-letter (or smaller) brands would already be identified through dictionary matching, and from grepping through the unlabelled data, we felt that bringing the limit any lower might do more harm than good.

   DAYTON 6FLV0 Motor Start Capacitor, 243-292 MFD, Round

Quantitative Measures

We had continually evaluated each of our approaches on validation sets during our dev cycle and had only used the test set once we were no longer developing our code. Statistics on the validation sets are however susceptible to randomness due to shuffling, so we validation measures are presented as those from three separate runs with identical code, and their average is stated as the final measure. Test statistics are absolute and are presented from only a single run.

Each validation run consisted of testing against 60 kept-aside items from the dev-set, chosen through random sampling without replacement.

Without Partial Matches

Validation statistics:

```
TP    FP    TN    FN
42    11    6     1
Pre:  79.2452830189
Rec:  97.6744186047

TP    FP    TN    FN
41    13    4     2
Pre:  75.9259259259
Rec:  95.3488372093

TP    FP    TN    FN
41    11    6     2
Pre:  78.8461538462
Rec:  95.3488372093
```

Precision = 78.01%
Recall = 96.12%

Test statistics:

```
TP    FP    TN    FN
91    14    11    4
Pre:  86.6666666667
Rec:  95.7894736842
```

Precision = 86.67%
Recall = 95.79%

With Partial Matches

Validation statistics:

```
TP    FP    TN    FN
54    1     4     1
Pre:  98.1818181818
Rec:  98.1818181818

TP    FP    TN    FN
56    2     1     1
Pre:  96.5517241379
Rec:  98.2456140351

TP    FP    TN    FN
51    2     5     2
```

```
Pre:  96.2264150943
Rec:  96.2264150943
```

Precision = 96.99%
Recall = 97.55%

Test statistics:

```
TP    FP    TN    FN
104   1     11    4
Pre:  99.0476190476
Rec:  96.2962962963
```

Precision = 99.05%
Recall = 96.30%

The reader may note here that both with and without partial matches, **we perform significantly better on the precision measure with our test set than we did with our validation set**. Though a matter of chance, we note this as a reliable indicator that our brand name extractor was not overfit by adding the heuristics as we did in our code development.

Our code is quoted with this report as *Appendix-A*. Furthermore, a detailed result of the test set statistics, stating each product name, its real brand, its extracted brand and the case (TP, FP, TN and FN, represented as ++, -+, +- and -- respectively), with partial matching, is provided in *Appendix-B*.

III. Limits

We are providing a detailed analysis of the different types of product names for which brand extractor gave erroneous results.

1. False Positive

   a. Ultra Slim 18Gbps Active High Speed HDMI Cable with RedMere Technology 6ft Blue
   Quite evidently the brand name is missing from the product name but our extractor returns 'Ultra' as the brand. It is so because the brand name dictionary happens to have a brand 'Ultra'. Hence our extractor fails for brand names which are commonly used english words with semantic meaning . A way to avert this would be some NLP algorithm to understand the context of the word usage.

   b. Keystone Jack - Modular RCA w/Green Center Flush Type (White)
   Here Keystone refers to the keystone module which is a standardized snap-in package for mounting a variety of low-voltage electrical jacks or optical connectors into a keystone wall plate, face plate, surface-mount box, or a patch panel. However the brand dictionary has an entry for the brand "Keystone" hence the extractor wrongly labels Keystone as a brand.

   c. LG G3 Case - GreatShield [LEGACY] Slim Fit Dual Layer Hybrid Cover with Kickstand for LG G3 (+Screen Shield) - Black
   Here the actual product is GreatShield but due to the occurrence of LG before GreatShield,the extractor mislabels it as the desired brand. One heuristic to remove this can be to look for possible brand names after keywords like "for","compatible with " etc and eliminate the from the . However we did not use this in our extractor to avoid over fitting.

   d. Off Lease REFURBISHED IBM Lenovo M58p C2D 3.0GHz 4GB 1TB DVD Win 7 Pro Desktop Computer + 19 LCD
   In case of two brands in a single product name , the extractor returns the first one. A heuristic can be to check whether the word(s) immediately following a brand name is also a brand in the product name attribute. In case of a hit the extractor should return both the names.

2. False Negative

   a. Twisstop Detangler w/Coiled 25-Foot Phone Cord Black
   The brand name is Twisstop but the extractor returns no brand.False Negative errors occurred when the brand name was missing from the brand dictionary. These were for those brands which had no occurrences in the development set, they had instances only in the test set.

   b. Griffin Survivor, Touch5G-Citron & Black
   The actual brand name, as is inculcated in the dictionary as well, is Griffin Technology. However from the information available in the product name, we entered Griffin Survivor as the brand name in the golden data. The extractor fails to capture the correct brand name here and results in a false negative. We did not apply any heuristic to rectify this because it might be the case that two completely different brand names share a few words. Then wrong brand might get identified although the names are highly similar. For example for the two different brands “XYZ ABC…” as appears in the product name and “XYZ EFG” as is seen in the dictionary, just returning brand name as either “XYZ” or “XYZ EFG” is wrong. “XYZ” would be correct only if it exists as an independent brand in the dictionary. However there exists a drawback of our existing extractor along similar lines.

   c. 15ft Premier Series XLR Male to 1/4inch TRS Male 16AWG Cable (Gold Plated)
   The actual brand is Premier Series and the extractor seemingly outputs the correct result “Premier”. But the brand Premier listed in the dictionary is possibly a different brand and so there is a semantic conflict. Hence brand names that have common words can result in errors.

   d. Bill Hicks & Co LTD SO PC900 LARGE PORTABLE CASE W/ ELECTRONIC LOCK
   Bill Hicks & Co LTD is not present in the dictionary. Hence the extractor failed to recognise it. However a heuristic could have been labeling the word(s) followed by Co,LTD,& Sons,Corp, Pvt,Corporation,Inc et al as the brand.

   e. D3p 32039 Victorious2012 Taking The Lead
   Here the brand dictionary has an entry for the full brand name D3 Publisher but

the golden data has D3p as the identified brand. The extractor fails to capture this inconsistency. A way to handle this can be maintaining both the acronym/abbreviation and the full form in the brand dictionary. E.g.: LG - Life's Good, BMW - Bayerische Motoren Werke, KFC - Kentucky Fried Chicken

Future Work

1. An extension to our extractor could have been accounting for spelling mistakes in the brand names in the product name attribute owing to human errors. A simple edit distance based check could suffice. However we might face problems for two different brand names Nikko and NikonNikkon which have very low edit distance between them , the extractor may wrongly label either one of them as an instance of the other. Hence the spell check should be done at the very last only when all the other cases fail to identify a valid brand name. We did not encounter any spelling errors in the dataset used . Hence we did not include it on our extractor.
2. For certain brands like FLEXBoot,Nimbus etc the mother brand is Monoprice just like Amazon is the mother brand for Fire, Echo , Kindle and Apple is the mother brand for iPhone,Macbook etc. The brand dictionary has Monoprice but not the individual brands FLEXBoot et al. The extractor can be extended to label the products, having only specific brand information in their product name , with the umbrella brand itself.

IV. Appendix

Appendix-A

```
import sys
import json
from itertools import groupby
import operator
import pprint
import random
import re

if(len(sys.argv) != 4):
      print >> sys.stderr, 'Usage: python cs784stage2.py elec_brand_dic.txt
_dev_set.tsv _test_set.tsv'

VALIDATION_RUN = False
VALIDATION_SIZE = 60 #Max 230
ALLOW_PARTIAL = True
IS_DEBUG = False

def DEBUG(debug):
      if IS_DEBUG:
            print debug

#http://stackoverflow.com/a/3313605/1492106
def sublistExists(list1, list2):
      return ''.join(map(str, list1)) in ''.join(map(str, list2))

brands = dict()

with open(sys.argv[1]) as f:
      for line in f:
            line = unicode(line, errors='ignore') #For character which are
not utf-8
            data = line.strip().split('\t') #.lower()
            brand = data[0]
            brands.update({data[0]:(brands.get(data[0], 0) + int(data[1]))})
      f.close()

global _dev_set
global _test_set

_dev_set = []
_test_set = []

with open(sys.argv[2]) as f:
      lines = f.readlines()
      if VALIDATION_RUN:
```

```
            if(VALIDATION_SIZE < len(lines)):
                  random.shuffle(lines)
            for line in lines[0:VALIDATION_SIZE]:
                  data = line.strip().split('\t') #.lower()
                  _test_set.append(data)
            lines = lines[VALIDATION_SIZE:]
      f.close()
      for line in lines:
            line = unicode(line, errors='ignore') #For character which are
not utf-8
            data = line.strip().split('\t') #.lower()
            _dev_set.append(data)
            for d in data[2:]:
                  brands.update({d:(brands.get(d, 0) + 1)})

if not VALIDATION_RUN:
      with open(sys.argv[3]) as f:
            lines = f.readlines()
            f.close()
            for line in lines:
                  line = unicode(line, errors='ignore') #For character which
are not utf-8
                  data = line.strip().split('\t') #.lower()
                  _test_set.append(data)

#brands_list     =     sorted(brands.items(),     key=operator.itemgetter(1),
reverse=True)
brands_list = sorted(brands.items(), key=lambda x: len(x[0]))

symbol_re = re.compile('[^\\w]+', re.IGNORECASE)
brands_re = []

for brand, count in brands_list:
      brands_re.append(re.compile('\\b'    +    symbol_re.sub('[^\\w]{0,3}',
brand) + '\\b', re.IGNORECASE))

#Heuristics
preposition_re = re.compile('\\bfor|with\\b', re.IGNORECASE)
limitIndex = 50
refurbished_re = re.compile('^Off\s*Lease\s*REFURBISHED\s*', re.IGNORECASE)
CaSe_re = re.compile('^[A-Z]+[a-z]+-?[A-Z]+\\w+\\b')
CASE_re = re.compile('^[A-Z]{5,}\\b') #CAPS, but not A, AN, THE, etc.

TP = 0
FP = 0
TN = 0
FN = 0

for item in _test_set:
      if len(item) is 2:
            item.append('')
```

```
	try:
		matched = CaSe_re.search(item[1]).group(0)
	except:
		matched = None
	for idx, brand in enumerate(brands_re):
		match = brand.search(item[1])
		if match is not None:
			if matched == None:
				matched = match.group(0)
			else:
				matchIndex = refurbished_re.sub('', item[1]).index(match.group(0))
				matchedIndex = refurbished_re.sub('', item[1]).index(matched)
				if matchIndex <= matchedIndex:
					matched = match.group(0)
	if matched is None:
		try:
			matched = CASE_re.search(item[1]).group(0)
		except:
			matched = None
	if matched is not None:
		flag = False
		if matched.lower() in item[2].lower():
			flag = True #Exact match with brand options
		else:
			if ALLOW_PARTIAL:
				if sublistExists(symbol_re.split(matched.lower()), [x.lower() for x in symbol_re.split(item[2])]):
					flag = True #Partial brand. E.g.: StarTech for Startech.com, Lenovo for Lenovo ThinkPad
		matchedIndex = item[1].index(matched)
		if matchedIndex > limitIndex:
			#Do not check beyond limit
			matched = None
			flag = False
		prepositionMatch = preposition_re.search(item[1])
		if prepositionMatch is not None:
			prepositionIndex = item[1].index(prepositionMatch.group(0))
			if prepositionIndex < matchedIndex:
				#print '>>>>>>>', prepositionMatch.group(0), matchedIndex, prepositionIndex
				matched = None
				flag = False #Compatibility. E.g.: 6ft A to Mini-B 8pin USB Cable w/ ferrites for Pentax Panasonic Nikon Digital Camera

	if matched is not None:
		if flag == True:
			DEBUG('\n>>' + item[1] + '\n>>' + item[2] + '\n>>' + matched + '\n++')
```

```
                    TP = TP + 1
              else:
                    try:
                        label = item[2]
                    except IndexError:
                        label = ''
                    DEBUG('\n>>' + item[1] + '\n>>' + label + '\n>>' + matched
+ '\n-+')
                    FP = FP + 1
        else:
              if not item[2]:
                    DEBUG('\n>>' + item[1] + '\n>>\n>>\n+-')
                    TN = TN + 1
              else:
                    DEBUG('\n>>' + item[1] + '\n>>' + item[2] + '\n>>\n--')
                    FN = FN + 1

Pre = (100.0 * TP)/(TP + FP)
Rec = (100.0 * TP)/(TP + FN)

print '\nTP\tFP\tTN\tFN'
print str(TP) + '\t' + str(FP) + '\t' + str(TN) + '\t' + str(FN)
print 'Pre:\t' + str(Pre)
print 'Rec:\t' + str(Rec)
```

Appendix-B

```
>>Mobile Battery Pack USB Black
>>
>>
+-

>>Phottix Battery Grip for Nikon D40 D40X D60 D3000 With Vertical Button
>>Phottix
>>Phottix
++

>>Tripp Lite Heavy-Duty Computer Power Cord - 14AWG 15A (NEMA 5-15P to
IEC-320-C13) 6-ft.
>>Tripp Lite
>>Tripp Lite
++

>>HP Pro x2 612 G1 2-In-One Notebook PC - Intel Core i3 4012Y 1.5 GHz, 4GB
RAM, 128GB SSD, 12.5 Touchscreen Display, Wind
>>HP Pro x2
>>HP
++

>>HP Scanjet 1000 Sheetfed Scanner - 600 dpi Optical - 48-bit Color - 8-bit
Grayscale - USB
>>HP Scanjet
>>HP
++

>>Gaiam Hydrangea Wood Case for iPhone 5 - Brown/Black
>>Gaiam
>>Gaiam
++

>>Raxxess CMP - Cable Management Panels
>>Raxxess
>>Raxxess
++

>>CTA DB-VM200 CTA BN-VM200 900 Mah 7.4V Rechargeable Battery
>>CTA
>>CTA
++

>>ProMaster 16GB SDHC 2-pack Class 10 Performance Memory Card
>>ProMaster
>>ProMaster
++

>>Cables To Go 02798 6 Feet DB25M to C36M Parallel Printer Cable
>>Cables To Go
```

```
>>Cables To Go
++

>>Off Lease REFURBISHED HP ProBook 6460b 2.6GHz i5 4GB 250GB DVD Win 7 Home
WiFi 14 Laptop Notebook
>>HP ProBook
>>HP ProBook
++

>>Seagate Enterprise NAS HDD ST5000VN0001 - Hard drive - 5 TB - internal -
3.5 - SATA 6Gb/s - 7200 rpm - buffer: 128 MB
>>Seagate
>>Seagate
++

>>Bill Hicks & Co LTD SO PC900 LARGE PORTABLE CASE W/ ELECTRONIC LOCK
>>Bill Hicks & Co LTD
>>
--

>>Epson T088420 88 Ink EPST088420
>>Epson
>>Epson
++

>>50FT 24AWG Cat6 550MHz UTP Bare Copper Ethernet Network Cable - White
>>
>>
+-

>>Lexmark C7700CH High-Yield Toner LEXC7700CH
>>Lexmark
>>Lexmark
++

>>Targus AWV1234US Screen Protector
>>Targus
>>Targus
++

>>Belkin Home-Office BE108230-12 8-Outlets Surge Suppressor Receptacles: 8
3550J
>>Belkin
>>Belkin
++

>>Reading the iPod as an Anthropological Artifact: Click Wheels and Cargo
Cults
>>
>>
+-

>>Clarion MWRXCRET Marine Remote Extension Cable, 25
```

```
>>Clarion
>>Clarion
++

>>Zebra Z Series ZM400 - Label printer - direct thermal / thermal transfer -
Roll (11.4 cm) - 203 dpi - up to 600 inch/mi
>>Zebra Z Series
>>Zebra
++

>>BRADY THT-37-483-10 Therm Transfer LabelPoly0.5x1.5 G5546484
>>BRADY
>>BRADY
++

>>Fosmon Digital Optical Coax to Analog RCA Audio Converter Adapter
>>Fosmon
>>Fosmon
++

>>1ft 16AWG Power Cord Cable w/ 3 Conductor PC Power Connector Socket
(C13/5-15P) - Black
>>
>>
+-

>>Startech.com Usb2dvimm6 6ft Usb Dvi M/m External Multi Monitor Video
Adapter Cable
>>Startech.com
>>Startech.com
++

>>Off Lease REFURBISHED Dell Optiplex GX755SF C2D 3.0GHz 8GB 80GB DVD Win 7
Pro64 Desktop Computer
>>Dell Optiplex
>>Dell Optiplex
++

>>V7 TD31GRY-1N Carrying Case (Messenger) for 10.1 Tablet, iPad, iPad Air,
iPad mini - Charcoal Gray - Scratch Resistant
>>V7
>>V7
++

>>Xerox No. 35A Black Toner Cartridge - Black - Laser - 1500 Page - 1 Each
>>Xerox
>>Xerox
++

>>SpaceSaver 19 Half-U UTP Cat5e Patch Panel, 24 Ports, Dual IDC
>>SpaceSaver
>>SpaceSaver
++
```

```
>>Jill-E Designs Metro Tablet Bag - Black Leather, Silver Trim - 384362
>>Jill-E Designs
>>Jill-E Designs
++

>>Innovera Remanufactured C8775WN 02 Ink IVR75WN
>>Innovera
>>Innovera
++

>>DotDefender PC+Silicone Case for Samsung Galaxy Note 3 - Black
>>DotDefender
>>DotDefender
++

>>QNAP TS-131 - NAS server - SATA-300 / eSATA - Gigabit Ethernet - iSCSI
>>QNAP
>>QNAP
++

>>20FT 24AWG Cat5e 350MHz UTP Bare Copper Ethernet Network Cable - Pink
>>
>>
+-

>>Refurbished Jensen VM9226BT 2DIN 6.2Widescreen DVD Bluetooth Car Stereo
Receiver
>>Jensen
>>Jensen
++

>>EDGE - DDR - 1 GB - DIMM 184-pin - 400 MHz / PC3200 - CL2.5 - 2.5 V -
unbuffered - non-ECC
>>EDGE
>>EDGE
++

>>Samsung IT AA-BS5N11B-US Series 5 Ultrabook Syn Leather
>>Samsung
>>Samsung
++

>>Twisstop Detangler w/Coiled 25-Foot Phone Cord Black
>>Twisstop
>>
--

>>NETGEAR N150 Wireless USB Adapter
>>NETGEAR
>>NETGEAR
++
```

```
>>Jill-e Designs JILL-E Laptop/Camera Backpack (Tan/Brown)
>>Jill-e Designs
>>Jill-e Designs
++

>>Nintendo DSI System Wallet(DSi)
>>Nintendo DSI
>>Nintendo
++

>>Tripp Lite 7-Port USB 3.0 SuperSpeed Hub with Dedicated USB 2A Charging
Port - USB - External - 8 USB Port(s) - 7 USB 3.
>>Tripp Lite
>>Tripp Lite
++

>>Sharp PN-Y325 - 32 Class ( 31.6 viewable ) - commercial use - PN-Y Series
LED-backlit LCD flat panel display - 1080p (
>>Sharp
>>Sharp
++

>>Kensington Pro Fit 72354 Wireless Mid Size Mouse - Wireless - Radio
Frequency - Scroll Wheel - Right-handed Only
>>Kensington Pro Fit
>>Kensington
++

>>3.5mm Stereo Plug to 2 x 6.35mm (1/4 Inch) Mono Jack Splitter Adaptor -
Gold Plated
>>
>>
+-

>>Minuteman PRO-RT PRO1500RT 1500 VA Tower/Rack mountable UPS - 1500 VA/1050
W - 132 V AC - 5 Minute - 2U Tower/Rack Mount
>>Minuteman
>>Minuteman
++

>>Kensington K39336US KeyFolio for iPad 1&2
>>Kensington
>>Kensington
++

>>CybertronPC ViperX5 Desktop PC with Intel Core i5-4690K Quad-Core
Processor, 8GB Memory, 1TB Hard Drive and Microsoft Windows 8.1 (Monitor Not
Included)
>>CybertronPC ViperX5
>>CybertronPC
++

>>Corlink AA1419034-COR Avaya Compatible 1000BASE-CWDM SFP 1GE 2G FC 1490nm
```

```
80km DOM SMF LC
>>Corlink
>>Corlink
++

>>C2G 12ft Velocity 3.5mm M/M Stereo Audio Cable - Mini-phone Male -
Mini-phone Male - 12ft - Blue
>>C2G
>>C2G
++

>>HDMI Port Saver Adapter (Male to Female) - Swiveling Type
>>
>>
+-

>>LG 42LB6300 42 1080p 120Hz Direct LED Smart HDTV
>>LG
>>LG
++

>>Pyle Dual Professional Audio Link Cable - Phono Male - RCA Male - 5ft
>>Pyle
>>Pyle
++

>>IN WIN EM-Series EM035 - Mini tower - micro ATX 350 Watt ( ATX12V/ PS/2 )
- black/silver - USB/Audio - IW-EM035.CH350TB3
>>IN WIN
>>IN WIN
++

>>PANDUIT TGTRIW Raceway Transition Fitting
>>PANDUIT
>>PANDUIT
++

>>75FT FLEXboot Series 24AWG Cat6 550MHz UTP Bare Copper Ethernet Network
Cable - Orange
>>FLEXboot
>>
--

>>Xerox XER106R01507 Phaser Toner Cartridge 12000 Page Yield Cyan
>>Xerox
>>Xerox
++

>>TRIPP LITE Smart1500LCD UPS, Line-Interactive, 1500VA/900W, LCD
>>TRIPP LITE
>>TRIPP LITE
++
```

```
>>Wacom Intuos Creative Stylus - Stylus - blue - for Apple iPad; iPad Air; iPad mini; iPad mini with Retina display; iPad
>>Wacom Intuos
>>Wacom
++

>>Tview HCAPACITOR Hid Capacitor 9-16v *yellow And Blue* Sold As Pair
>>Tview
>>Tview
++

>>Canon PIXMA 9539B002 MG6620 Wireless Photo All-in-One Inkjet Printer - 4800 x 1200 dpi Color 600 x 600 dpi Black - 3.0-inch LCD Display - USB - 100-240V AC 50-60 Hz - Black
>>Canon PIXMA
>>Canon
++

>>RCA 25 Stereo Audio Cable
>>RCA
>>RCA
++

>>HP 3-bin Stapling Mailbox - Printer mailbox with stapler - 900 sheets in 3 tray(s) - for Color LaserJet Enterprise Flow
>>HP
>>HP
++

>>Califone International Cls729-4-03 4-Person Wireless Listening System - Green
>>Califone International
>>Califone
++

>>Paladin Tools Cable Tie Gun
>>Paladin Tools
>>Paladin Tools
++

>>Panasonic KX-TG9542B Dect 6.0 2-Line Cordless Phone w/ Link-to-Cell & 2-Handsets + 2 Line Handset For KX-TG954X + Blue Planet Wireless Bluetooth Headset and Focus $10 Gift Card
>>Panasonic
>>Panasonic
++

>>Antenna Croc Laptop Sleeve for Macbook
>>Antenna
>>Antenna
++

>>Mortal Kombat Soundtrack
```

```
>>Mortal Kombat
>>
--

>>Unit Origami: Multidimensional Transformations
>>
>>
+-

>>Comprehensive Pro AV/IT High Speed HDMI Cable with ProGrip, SureLength,
CL3- Dark Green 1.5ft - HDMI for Network Device
>>Comprehensive
>>Comprehensive
++

>>Cisco ASA 5505 SSL / IPsec VPN Adaptive Security Appliance - 8 x 3 x 1 x
Management
>>Cisco
>>Cisco
++

>>LG G3 Case - GreatShield [LEGACY] Slim Fit Dual Layer Hybrid Cover with
Kickstand for LG G3 (+Screen Shield) - Black
>>GreatShield
>>LG G3
-+

>>POMONA 6728-2 Banana Jack to Plug Adapter 1-6/7 In. L
>>POMONA
>>POMONA
++

>>Raritan Computer Interface Module - RJ-45 Female Network, Type A Male USB,
HD-15 Male Video
>>Raritan Computer
>>Raritan Computer
++

>>EDGE - DDR3 - 8 GB - DIMM 240-pin - 1600 MHz / PC3-12800 - registered -
ECC
>>EDGE
>>EDGE
++

>>Visiontek 900308 Radeon 4350 Graphic Card - 512 MB DDR2 SDRAM - PCI
Express x1 - 800 MHz Memory Clock - 2560 x 1600
>>Visiontek
>>Visiontek
++

>>Tripp Lite SMART2200RMX SmartPro UPS Line Interactive 2200VA 100-110-120V
2U Rack-Tower
>>Tripp Lite
```

```
>>Tripp Lite
++

>>Intel Xeon E5-2697 v2 Dodeca-core (12 Core) 2.70 GHz Processor Upgrade -
Socket R LGA-2011 - 3 MB - 30 MB Cache - 8 GT/s
>>Intel Xeon
>>Intel
++

>>Dell Toner Cartridge - Black - Laser - High Yield - 2000 Page - 1 / Pack
>>Dell
>>Dell
++

>>Westcott Photo Illusion Lighting Kit Enhanced Version with New Software
>>Westcott
>>Westcott
++

>>ASUS AM1I-A - Motherboard - mini ITX - Socket AM1 - USB 3.0 - Gigabit LAN
- onboard graphics (CPU required) - HD Audio (
>>ASUS
>>ASUS
++

>>ASUS RADEON R9 280 DC2T 3GB DDR5 (Refurbished)
>>ASUS
>>ASUS
++

>>LENMAR CLZ323M Cellphone Battery1100mAhFor Motorola
>>LENMAR
>>LENMAR
++

>>Tripp-Lite AVR750U Line-Interactive UPS
>>Tripp-Lite
>>Tripp-Lite
++

>>Kenwood DDX470 6.1 In-Dash DVD/MP3/USB Receiver w/ Bluetooth and Streaming
>>Kenwood
>>Kenwood
++

>>Off Lease REFURBISHED HP EliteBook 2540p i7 2.13GHz 2GB 160GB DVD/CDRW Win
7 Pro Laptop Computer PC
>>HP EliteBook
>>HP
++

>>Lexmark #17 (10N0016) Black Compatible Ink Cartridge
>>Lexmark
```

```
>>Lexmark
++

>>Mobile Edge NeoGrid Sleeve for Apple iPad and 10 Tablets, Assorted Color
Stitching
>>Mobile Edge
>>Mobile Edge
++

>>Off Lease REFURBISHED Dell Optiplex 780 3.0GHz C2D 2GB 160GB DVD Win 7 Pro
Ultra Small Computer PC
>>Dell Optiplex
>>Dell Optiplex
++

>>BellO Digital In-Ear Headphones, Assorted Colors
>>BellO Digital
>>BellO Digital
++

>>Off Lease REFURBISHED Dell Optiplex 755 Core 2 Duo 2.3Ghz 2GB 80GB DVD
Windows 7 Pro Desktop Computer
>>Dell Optiplex
>>Dell Optiplex
++

>>Xerox 113R00693 High-Yield Toner XER113R00693
>>Xerox
>>Xerox
++

>>Samsung MLT-D203U - Black - original - toner cartridge - for ProXpress
M4020ND, M4020NX, M4025NX, M4070FR, M4070FX, M407
>>Samsung
>>Samsung
++

>>Pyle Professional Audio Link Cable
>>Pyle
>>Pyle
++

>>Envizen Digital EM63 EVO 4 GB Tablet - 7 - Wireless LAN - Allwinner Cortex
A7 A23 1.20 GHz - 512 MB RAM - Android 4.2 J
>>Envizen Digital
>>Envizen Digital
++

>>Seagate Backup Plus Portable STDR2000100 2 TB External Hard Drive - USB
3.0 - Portable - Black
>>Seagate Backup Plus
>>Seagate
++
```

```
>>Evercool Case Fan EC6025 - 60mm Dual PWM Fan - F-EC6025H12BP
>>Evercool
>>Evercool
++

>>SONY, ICDPX440, DIGITAL VOICE RECORDER, 4GB
>>SONY
>>SONY
++

>>Dell G907C Toner Cartridge - Cyan - Laser - 3000 Page - 1 Pack
>>Dell
>>Dell
++

>>Kensington Virtuoso Metro Stylus and Pen KMW39393
>>Kensington Virtuoso
>>Kensington Virtuoso
++

>>Hewlett-Packard HEWQ5953A Print Cartridge- 10000 Page Yield- Magenta
>>Hewlett-Packard
>>Hewlett-Packard
++

>>SonicWALL TZ 210/NSA 240 Rack Mount Kit
>>SonicWALL
>>SonicWALL
++

>>Off Lease REFURBISHED Dell Optiplex 330 1.8GHz CD 4GB 80GB DVD Win7 Home
Desktop Computer w/ 19 LCD
>>Dell Optiplex
>>Dell Optiplex
++

>>Sabrent SBT-UPPC USB to Parallel ( Printer IEEE 1284 ) Cable
>>Sabrent
>>Sabrent
++

>>Acer C720-34054G03aii 11.6 LED (ComfyView) Chromebook - Intel Core i3
i3-4005U 1.70 GHz - 4 GB RAM - 32 GB SSD - Intel
>>Acer
>>Acer
++

>>Peerless Ext-101 1 Black Fixed Length Extension Columns For Jmc Series
Ceiling Mounts (ext101)
>>Peerless
>>Peerless
++
```

```
>>Tech Armor HD Clear Screen Protector - For iPhone 6 - SP-HD-APL-IP6-3
>>Tech Armor
>>Tech Armor
++

>>Belkin Mouse Pad Blk Waverestgel (Pack of 2)
>>Belkin
>>Belkin
++

>>SanDisk Optimus Extreme - Solid state drive - 100 GB - internal - 2.5 -
SAS 6Gb/s
>>SanDisk
>>SanDisk
++

>>Xpresskit GM IMMBLZR OVRRD INTERFC DEIXK06
>>Xpresskit
>>Xpresskit
++

>>SAMSUNG UN55H7150 55in 1080p 120Hz Apps QuadCore 3 (Refurbished)
>>SAMSUNG
>>SAMSUNG
++

>>Discwasher RD1141 CD/DVD Laser Lens Cleaners with 1 Brush
>>Discwasher
>>Discwasher
++

>>Corlink 7SW-000-COR Adva Compatible 1000BASE-CWDM SFP 1GE 2G FC 1470nm
80km DOM SMF LC
>>Corlink
>>Corlink
++

>>6ft A to Mini-B 8pin USB Cable w/ ferrites for Pentax Panasonic Nikon
Digital Camera
>>
>>
+-

>>Kensington KeyFolio Pro 2 Keyboard KMW39512
>>Kensington KeyFolio
>>Kensington
++

>>Xerox XER106R01438 Toner Cartridge- Phaser 7500- 17- 800 Page Yield-
Yellow
>>Xerox
>>Xerox
```

```
++

>>Netgear R6200-100NAS Dual Band Gigabit Wireless Router + Netgear WiFi USB
Adapter + Cat5e Network Cable + Blowoff Air Duster
>>Netgear
>>Netgear
++

>>1000FT 24AWG Cat6 550MHz UTP Stranded In-Wall Rated (CM) Bulk Ethernet
Bare Copper Cable - Black
>>
>>
+-

>>Intellinet 300N Wireless Access Point - 300 Mbps, MIMO, Bridge, Repeater,
Multiple SSIDs and VLANs, External,IEEE 802.1d
>>Intellinet
>>Intellinet
++

>>Universal Windshield Mount for iPhone, Samsung Galaxy S, and most Mobile
Phones - Black
>>
>>
+-

TP    FP    TN    FN
104   1     11    4
Pre:  99.0476190476
Rec:  96.2962962963
```