Understanding Scanned Receipts

Eric Melz
300 S Reeves Dr.
Beverly Hills, CA 90212
eric@emelz.com

Abstract

Tasking machines with understanding receipts can have important applications such as enabling detailed analytics on purchases, enforcing expense policies, and inferring patterns of purchase behavior on large collections of receipts. In this paper, we focus on the task of Named Entity Linking (NEL) of scanned receipt line items. Specifically, the task entails associating shorthand text from OCRd receipts with a knowledge base (KB) of grocery products. For example, the scanned item “STO BABY SPINACH” should be linked to the catalog item labeled “Simple Truth Organic™Baby Spinach”. Experiments that employ a variety of Information Retrieval techniques in combination with statistical phrase detection shows promise for effective understanding of scanned receipt data.

1 Introduction

Tasking machines with understanding receipts can have important applications such as enabling detailed analytics on purchases, enforcing expense policies, and inferring patterns of purchase behavior on large collections of receipts. In this paper, we focus on the task of Named Entity Linking (Hachey et al., 2012) of scanned receipt line items. Specifically, the task entails associating shorthand text from OCRd receipts with a knowledge base (KB) of grocery products. For example, the scanned item “STO BABY SPINACH” should be linked to the catalog item labeled “Simple Truth Organic™Baby Spinach”.

2 Related Work

A literature review reveals virtually no published work in this specific domain. While there is a body of work researching text extraction from scanned receipts (e.g. Huang et al., 2019), the work is primarily focused on Named Entity Recognition (NER) instead of Named Entity Linking (NEL). That is, systems are considered successful if they can identify text items such as store locations, totals, etc, but they are not evaluated with respect to the interpretation of the extracted text.

Although no papers exist on linking scanned entities, there is literature in other areas that appear potentially relevant to the subject task. This includes work on general-purpose techniques for building abbreviation dictionaries, acquisition of medical abbreviations (e.g., “COPD” $\rightarrow$ “Chronic Obstructive Pulmonary Disorder”), and normalization of social media content (e.g., “ur coooool” $\rightarrow$ “you are cool”). The follow sections summarize a few papers in these areas.

2.1 Language Independent Acquisition of Abbreviations

(Glass et al., 2017) describe a language-independent technique for acquiring abbreviations and their expansions, by exploiting Wikipedia redirect and disambiguation pages. They begin by motivating the acquisition of abbreviations, noting that the explosion of social media has made the need for abbreviations increasingly important. They also note that a token such as “ACE” could have multiple expansions, including “accumulated cyclone energy” and “American Council on Education” in addition to the word “ace” (as in “Ace of spades”).

The authors present related work, noting that most of the previous work for abbreviation detection and expansion extraction has been in the domain of English biomedical text. A common strategy is to identify occurrences where an abbreviation is explicitly paired with its expansion for example through a pattern involving a parenthetical such as <short form> (<long form>) or <long form> (<short form>). Other approaches consider the contexts of short form and long form occurrences, pairing short forms with long forms according to their distributional similarity by measuring the cosine of their context vectors. Another approach uses supervised learning, considering features such as string similarity and other characteristics of the short and long forms.

The authors work is based on previous work by (Jacquet et al., 2014) who describe a technique for mining abbreviations by making use of Wikipedia redirection pages. The authors observe that, due to the use of only redirect pages for the gold standard annotation, a shortcoming of the prior work is that each abbreviation only has a single expansion even though multiple different expansions are possible for some of the abbreviations. To remedy this shortcoming, the authors propose mining disambiguation pages in addition to redirect pages to gather multiple possible long-form expansions.

The authors mine redirect and disambiguation pages for abbreviations, while applying several rules such as (a) Short forms are restricted to ten characters or less, (b) At least half of the short-form characters must be upper case, and (c) The long-form must be at least twice as long as the short form, with at least two tokens. They generate candidate expansions and then score the expansions. Scoring occurs by computing features for synonym similarity, topic similarity, and surface similarity. Synonym similarity means that one term can be replaced with another while preserving the meaning of the sentence and is assessed using word embeddings using word2vec (Mikolov et al., 2013). Topical relatedness means that two terms occur in the same sorts of documents, and is assessed using Latent Semantic Analysis (Deerwester et al., 1990). Surface similarity is the overlap in the surface forms of the terms by computing the best possible alignment between a short form and a long form. The three similarity scores are combined using a logistic regression model.

The authors compare their system with a previous system developed by (Schwartz and Hearst, 2003) that extracts abbreviations using parentheses based patterns. The metric used to compare systems is Area Under the Precision/Recall curve. Without the scoring extensions, the 2 systems are comparable: the Schwartz and Hearst system has an AUC of 0.359 and the Candidate System has an AUC of 0.324. However, by adding the alignment and embedding scoring extensions, the Candidate Systems performance improves to an AUC of 0.480.

2.2 Clinical Abbreviation Expansion

(Liu et al., 2015) describe a system for identifying clinical abbreviation expansions. They note that abbreviations are heavily used in medical literature and documentation. In notes written by physicians, high workloads and time pressure intensify the need for using abbreviations. This is especially true within intensive care medicine, where its crucial that information is expressed in the most time efficient manner to provide time-sensitive care to critically ill patients. Within the arena of medical research, abbreviation expansion using NLP can enable knowledge discovery and has the potential to improve quality of care.

The author's system works as follows. Word embeddings are trained using word2vec (Mikolov et al., 2013). The material used to train embeddings consists of medical texts such as articles, journals, and books, in addition to hand-written Intensive Care notes. To generate expansions for abbreviations in the hand-written notes, abbreviations are extracted from the notes, and then matched against a domain-specific abbreviation knowledge base. From this list of expansions, embedding vectors are retrieved for the abbreviation and candidate expansion. A similarity score is computed for each (abbreviation, expansion) pair, producing a ranked list of candidates expansions.

To test the performance of the system, a ground-truth dataset is produced by having physicians manually expand and normalize the handwritten notes. The authors compare their model against several baselines. For example, one baseline chooses the highest rated candidate expansion in the domain specific knowledge base. Comparing accuracy of the authors system against the baselines results in a 50%+ increase. For example the rating baseline has an accuracy of 21% and the authors system has an accuracy of 83%.

2.3 Social Media Text Normalization

(Lourentzou et al., 2019) present a Sequence to Sequence (Seq2Seq) model for normalizing social media text. They observe that social media texts have an enormous amount of variation, and that text normalization systems that rely on surface or phonetic representations may be ill-equipped to handle such variability. To rectify this situation, they propose a hybrid word-character Seq2Seq model with attention. This type of model has been successfully applied to tasks such as machine translation, and has promise for text normalization.

The authors frame the task of text normalization as mapping an out-of-vocabulary (OOV) non-standard word to an in-vocabulary (IV) standard word that preserves the meaning of the sentence. The non standard forms in user generated content include misspellings (defenitely → definitely), phonetic substitutions (2morrow → tomorrow), shortening (convo → conversation), acronyms (idk → i dont know), slang ("low key", "woke"), emphasis (coooool → cool), and punctuation (doesnt → doesn't).

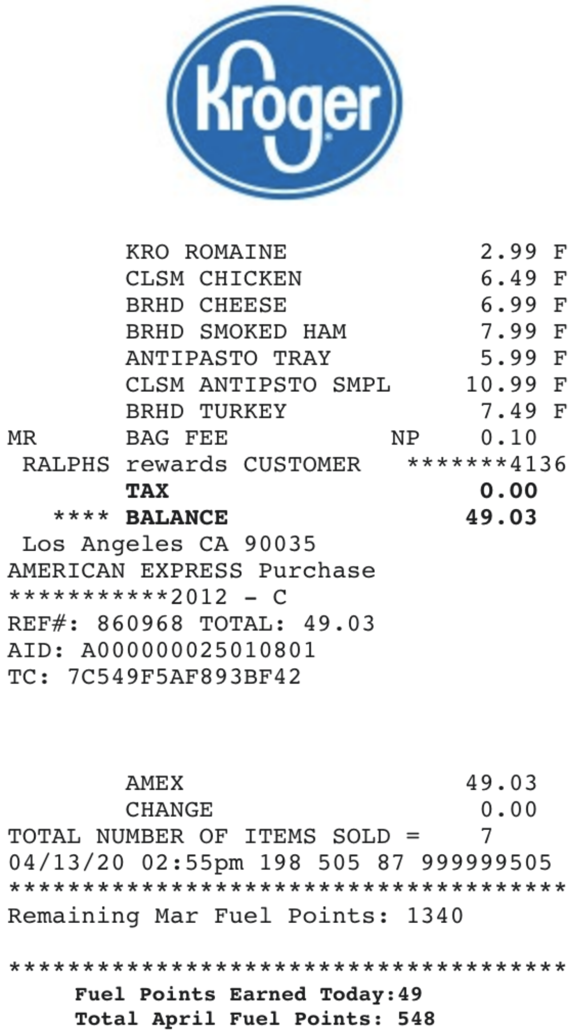
The authors note that lexicon-based approaches are not able to handle social media text properly. String similarity, such as edit distance, does not work on non-standard words where the number of edits is large, for example abbreviation. Additionally, systems that rely on candidate generation and scoring are limited in that they are not able to handle multiple normalization errors at once, e.g., spelling errors on an acronym. The authors suggest that using end-to-end neural models, particularly Seq2Seq models can deal with these shortcomings.

The authors train a bidirectional word-based Seq2Seq model to translate unnormalized text to normalized texts. OOV words are trained using a character-based Seq2Seq model. The dataset is enhanced by synthetically generating negative examples based on common normalization transformations. The network is trained on source sequences and target sequences. An example source is "got exo to share, u interested? Concert in hk !", with a corresponding target of "got extra to share, are you interested? Concert in hong kong !".

The authors present results for several variations of the model, including a word-level Seq2Seq model and the hybrid word-char Seq2Seq model. The best score is an F_1 score of 83.94 on the hybrid word-char Seq2Seq model.

3 Data

For this task, we need a dataset which includes scanned receipt product mentions (e.g, "BRHD CHEESE") and the corresponding product entities (e.g., "Boar's Head Monterey Jack with Jalapeno Pre-Sliced Cheese"). A brief web search revealed that no such publicly available dataset exists. To obtain a dataset, we built our own by scraping a grocery store website that contains purchase data. Specifically, we use our personal loyalty account with Ralph's (a subsidiary of Kroger) to obtain representations of scanned receipts along with corresponding web pages that contain fully-resolved entities. As an example, Figure 1 shows an instance of a receipt.



```
        KRO ROMAINE                2.99 F
        CLSM CHICKEN               6.49 F
        BRHD CHEESE                6.99 F
        BRHD SMOKED HAM            7.99 F
        ANTIPASTO TRAY             5.99 F
        CLSM ANTIPSTO SMPL        10.99 F
        BRHD TURKEY                7.49 F
MR      BAG FEE             NP     0.10
 RALPHS rewards CUSTOMER    *******4136
        TAX                        0.00
   **** BALANCE                   49.03
 Los Angeles CA 90035
AMERICAN EXPRESS Purchase
***********2012 - C
REF#: 860968 TOTAL: 49.03
AID: A000000025010801
TC: 7C549F5AF893BF42


        AMEX                      49.03
        CHANGE                     0.00
TOTAL NUMBER OF ITEMS SOLD =      7
04/13/20 02:55pm 198 505 87 999999505
**************************************
Remaining Mar Fuel Points: 1340

**************************************
     Fuel Points Earned Today:49
     Total April Fuel Points: 548
```

Figure 1: Scanned Receipt

Figure 2 shows part of the corresponding web page which contains linked representations of the purchased items.

We scrape both the text content of the raw receipts and the user-friendly web rendering, then join the raw receipt data with the corresponding web data. This produces a JSON structure per receipt. A sample of the JSON is shown in Figure 3. The "raw" field represents the product mention, and the "web" field represents the label associated with the entity. The "id" field is scraped from the

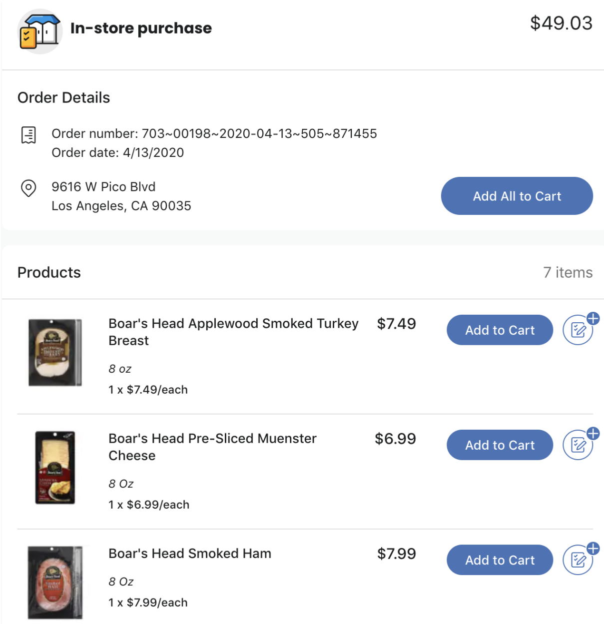


Figure 2: Web Receipt

web page and can be used as a succinct identifier for the entity.


```
[
  {
    "web": "Avocado - Extra Large",
    "raw": "AVOCADO HASS",
    "id": "0000000004770"
  },
  {
    "web": "Beyond Meat Hot Italian Plant-Based Sausage",
    "raw": "BYND SSG HT ITLN",
    "id": "0085262900475"
  },
  {
    "web": "Eggland's Best Organic Grade A Large Brown Eggs",
    "raw": "EGGLANDS BEST EGGS",
    "id": "0071514171682"
  },
  {
    "web": "Fancy Feast Grilled Beef Feast in Gravy Wet Cat Food",
    "raw": "FFST CAT FOOD",
    "id": "0005000004070"
  },
  {
    "web": "Fancy Feast Minced Turkey Feast in Sauce Wet Cat Food",
    "raw": "FFST CAT FOOD",
    "id": "0005000043494"
  },
```


Figure 3: JSON representation of joined "raw" and "web" data

The dataset consists of 65 scraped receipts, producing 711 non-unique line line items, and 296 unique line items. All data and code for these experiments are available on Github (Eric Melz).

4 Methodology

To evaluate model performance, we gather unique mentions and measure the accuracy of predicting entities. This can be conceived as a multi-class classification task where the entities to be predicted are the classes. An alternative metric would be to use a macro-average F_1 score, but this is overkill for this specific experiment setup since there is a uniform distribution across classes: each class is represented by exactly one test instance. To be concrete, heres an example. Suppose we have the following 2 unique mentions:

- BRHD CHEESE
- AVOCADO

Further suppose that the entity for BRHD CHEESE was correctly predicted as Boar's Head Monterey Jack with Jalapeno Pre-Sliced Cheese, and the prediction for AVOCADO yielded nothing. The first prediction is a "hit" and the second is a "miss". Dividing the total hits by the total number of predictions, we obtain an accuracy of $(1 + 0) / 2 = 0.5$.

Note that the mention representations contain much less information than the entity labels. In the above example, BRHD CHEESE is matched with "Boar's Head Monterey Jack with Jalapeno Pre-Sliced Cheese", but also could have been matched to "Boar's Head Spicy Cheddar Cheese", or a number of other types of Boars Head cheese. To account for this ambiguity, a prediction is counted as a hit if it is any of the possible resolutions of the product mention. In the previous example, both of the long descriptions would be considered hits for BRHD CHEESE.

The dominant modeling paradigm for the entity linking system we use is Information Retrieval. A baseline model indexes entity labels using the Lucene (Apache Software Foundation) IR engine. Lucene provides a toolkit of tokenizers and token analyzers, enabling many strategies for matching text. The most basic setup uses strict matching on tokens, providing a good baseline model. Subsequent experiments tune of the by selecting more sophisticated IR and NLP techniques such as using wildcard queries, phrase detection, etc.

5 Experiments

Lucene's default scoring formula is BM25 (Robertson, 2009). BM25 is based on a bag-of-words approach. The score of a document D given a query Q which contains the words

$q_1, ..., q_n$ is given by:

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \quad (1)$$

where $f(q_i, D)$ is q_i's term frequency in the document D, $|D|$ is the length of the document D in words, and avgdl is the average document length in the text collection from which documents are drawn. k_1 and b are free parameters.

5.1 Baseline

Blah blah blah

5.2 Wildcards

Blah blah blah

5.3 Mashed Wildcards

Blah blah blah

5.4 Phrases

Blah blah blah

5.5 Fuzzy Phrases

Blah blah blah

6 Results

blah

7 Future Work

blah

8 Conclusion

blah

References

Apache Software Foundation. Lucene.

S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R.A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*, pages 391–407.

Eric Melz. 2019. Eric melz github.

Michael R. Glass, Md. Faisal Mahbub Chowdhury, and Alfio Massimiliano Gliozzo. 2017. Language independent acquisition of abbreviations. *CoRR*, abs/1709.08074.

Ben Hachey, Will Radford, Joel Notham, Matthew Honnibal, and James R. Curran. 2012. Artificial intelligence, wikipedia and semi-structured resources evaluating entity linking with wikipedia. *Artificial Intelligence*, 194(11):130–150.

Z. Huang, K. Chen, J. He, X. Bai, D. Karatzas, S. Lu, and C. V. Jawahar. 2019. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520.

Guillaume Jacquet, Maud Ehrmann, and Ralf Steinberger. 2014. Clustering of multi-word named entity variants: Multilingual evaluation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Yue Liu, Tao Ge, Kusum Mathews, Heng Ji, and Deborah McGuinness. 2015. Exploiting task-oriented resources to learn word embeddings for clinical abbreviation expansion. In *Proceedings of BioNLP 15*, pages 92–97, Beijing, China. Association for Computational Linguistics.

Ismini Lourentzou, Kabir Manghnani, and ChengXiang Zhai. 2019. Adapting sequence to sequence models for text normalization in social media. *CoRR*, abs/1904.06100.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

S. Robertson. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Ariel S. Schwartz and Marti A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Symposium on Biocomputing*, pages 451–462.

A Appendices

blah

B Supplemental Material

blah blah blah