UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Johny Moreira da Silva

Augmenting Product Knowledge Graphs with Subjective Information

Recife

2023

Johny Moreira da Silva

Augmenting Product Knowledge Graphs with Subjective Information

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

Área de Concentração:
Inteligência Computacional

Orientador (a):
Luciano de Andrade Barbosa

Recife

2023

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S586a Silva, Johny Moreira da
Augmenting *product knowledge graphs with subjective information* / Johny Moreira da Silva. – 2023.
145 f.: fig., tab.

Orientador: Luciano de Andrade Barbosa.
Tese (Doutorado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2023.
Inclui referências e apêndices.

1. Inteligência computacional. 2. Aprendizagem. I. Barbosa, Luciano de Andrade (orientador). II. Título.

006.31 CDD (23. ed.) UFPE - CCEN 2023-44

Johny Moreira da Silva

“Augmenting Product Knowledge Graphs with Subjective Information”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 02/03/2023.

_____

Orientador: Prof. Dr. Luciano de Andrade Barbosa

BANCA EXAMINADORA

_____

Prof. Dr. Cleber Zanchettin
Centro de Informática / UFPE

_____

Prof. Dr. Fernando Maciano de Paula Neto
Centro de Informática / UFPE

_____

Prof. Dr. Altigran Soares da Silva
Instituto de Computação / UFAM

_____

Prof. Dr. José Maria da Silva Monteiro Filho
Departamento de Computação / UFC

_____

Prof. Dr. Leandro Balby Marinho
Departamento de Sistemas e Computação / UFCG

To my family for all support and love.

ACKNOWLEDGEMENTS

I want to thank my advisor Prof. Luciano Barbosa for all these years of guidance and knowledge sharing. Also, I would like to thank my professors at PPGCC (Programa de Pós-Graduação em Ciência da Computação) and all my former professors from middle school to graduation, who have guided me through this journey in the search for knowledge. Sharing knowledge is one of the noblest and most admirable attitudes I know. You'll always be remembered.

I sincerely thank the members of my examination board who have contributed to improving this thesis work with insightful comments and feedback. Thank you Prof. Dr. Fernando Maciano, Prof. Dr. José Maria Monteiro, and Prof. Dr. Leandro Balby Marinho. A special thank you to Prof. Dr. Altigran Soares da Silva and Prof. Dr. Cleber Zanchettin for all the valuable contributions developed throughout this Doctorate, some of which have taken the form of published papers.

Thank you to Tiago de Melo for all the contributions and knowledge sharing and for providing the data necessary to test and validate the PGOpi pipeline.

Thank you to Michael Cruz and Levy de Souza Silva for evaluating the synthetic triples generated by the SYNCOPATE approach. A special thanks to Everaldo Neto for all the discussions, insights, and contributions.

Special gratitude goes out to CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for providing the funding for the work. These past four years have not been easy. Right?!

Last but not least, I express my very profound gratitude to my family and friends for providing unconditional support and continuous encouragement throughout my years of study and life in general. Thanks for understanding my absence in important moments for the last few years. This accomplishment would not have been possible without you.

To all my friends, walking this path with you was way more fun.

Thank you.

"Even the interpretation and use of words involves a process of free creation" (CHOMSKY, 2008).

ABSTRACT

Product Graphs (PGs), are knowledge graphs on consumer product data. They have become popular lately due to their potential to enable AI-related tasks in e-commerce. PGs contain facts on products (e.g., mobile phones) and their characteristics (e.g., brand, dimensions, and processor) automatically gathered from several sources. Enriching these structures with dynamic and subjective information, such as users' opinions, is essential for improving recommendations, searching, comparison, and pricing. However, this is a novel task, and works trying to handle this are based on supervised approaches. In this thesis, we address this task by exploring two complementary stages: (1) We build a weak-supervised pipeline called **P**roduct **G**raph enriched with **Opi**nions (PGOpi) which augments PGs with users' opinions extracted from product reviews. For that, we explore a traditional method for opinion mining, Distant Supervision based on word embeddings to alleviate manual labor dependency for training, and Deep Learning approaches to map extracted opinions to targets in the PG; (2) We devised **SYN**theti**C** **OP**inion**A**ted **T**ripl**E**s (SYNCOPATE), a generator that autonomously builds opinionated triples and can replace traditional methods for extracting aspect-opinion pairs from opinionated reviews. We build it by exploring In-Context Learning on an adapted pretrained Language Model. Finally, we apply post-processing to clean up and label the autonomously generated text. We perform the experimental evaluation of both frameworks. We evaluated PGOpi on five product categories of two representative real-world datasets. The proposed weak-supervised approach achieves a superior micro F1 score over more complex weak-supervised models. It also presents comparable results to a fully-supervised state-of-the-art (SOTA) model. We evaluated SYNCOPATE by augmenting existing benchmark datasets with the generated data and comparing the performance of four SOTA models on aspect-opinion pair extraction. The results show that the models trained on the generated synthetic data outperform those trained on a small percentage of human-labeled data. Furthermore, three human raters' manual inspection of these triples attested to their quality.

Keywords: product graphs; subjective data; opinion mining; language model; zero-shot learning; distant supervision.

RESUMO

Grafos de Produto, do inglês *Product Graphs (PGs)*, são grafos de conhecimento com dados sobre produtos de consumo. Essas estruturas têm o potencial de facilitar tarefas de Inteligência Artificial no comércio eletrônico. Os *PGs* armazenam dados factuais sobre produtos (ex: *smartphones*) e suas características (ex: marca, dimensões, e processador) coletados de diversas fontes. O enriquecimento dessas estruturas com informações dinâmicas e subjetivas, como opiniões de usuários, pode contribuir para a melhoria dessas tarefas. No entanto, esta é uma nova tarefa e os trabalhos existentes são baseados em abordagens supervisionadas. Neste trabalho de tese nós abordamos essa tarefa por meio de duas etapas complementares: (1) Nós desenvolvemos uma abordagem semi-supervisionada chamada ***P****roduct* ***G****raph enriched with* ***Opi****nions (PGOpi)* para enriquecimento de *PGs* com opiniões extraídas de avaliações de clientes. Para isso, exploramos mineração de opinião, Supervisão Distante baseada em representação de palavras para mitigar a dependência na rotulagem manual de dados de treino, e utilizamos Aprendizagem Profunda para mapear as opiniões extraídas até os nós do *PG*; (2) Nós construímos um gerador de triplas opinativas chamado ***SYN****theti****C*** ***OP****inion****A****ted* ***T****ripl****E****s (SYNCOPATE)* que pode substituir métodos tradicionais para extração pareada de aspectos e opiniões em avaliações de produtos. Para construí-lo realizamos *In-Context Learning* em um Modelo de Linguagem pré-treinado e adaptado. Nós efetuamos a avaliação experimental das duas etapas. O *PGOpi* foi avaliado em cinco categorias de produtos de duas plataformas de *e-commerce*. O *PGOpi* alcançou valores de micro *F1-score* superiores a modelos semi-supervisionados mais complexos, e apresentou performance comparável a um modelo de estado-da-arte totalmente supervisionado. O *SYNCOPATE* foi avaliado aumentando bases de treino de *benchmarking* com as triplas opinativas geradas sinteticamente. Quatro modelos de estado-da-arte para extração pareada de aspectos e opiniões foram treinados com esses dados sintéticos e avaliados. Os resultados obtidos mostraram que os modelos treinados com dados sintéticos apresentaram performance superior àqueles treinados em uma pequena porcentagem de dados rotulados e curados por humanos. Três avaliadores humanos atestaram a qualidade das triplas geradas sinteticamente.

Palavras-chave: grafos de produto; dados subjetivos; mineração de opinião; modelos de linguagem; aprendizagem zero-shot; supervisão distante.

LIST OF FIGURES

LIST OF LISTINGS

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ABSA	Aspect-Based Sentiment Analysis
AE	Aspect Extraction
AFOE	Aspect-Oriented Fine-Grained Opinion Extraction
AI	Artificial Intelligence
AOPE	Aspect-Opinion Pair Extraction
ASC	Aspect Sentiment Classification
ASOE	Aspect-Specified Opinion Extraction
ATE	Aspect Term Extraction
ATSE	Aspect Terms Span Extractor
AUC	Area Under the Curve
BERT	Bidirectional Encoder Representation from Transformer
BiLSTM	Bidirectional Long Short-Term Memory
C4	Colossal Clean Crawled Corpus
CNN	Convolutional Neural Network
CRF	Conditional Random Fields
DAPT	Domain-Adaptative Preraining
DNN	Deep Neural Network
DS	Distant Supervision
GCN	Graph Convolutional Networks
GPT	Generative Pre-Training
GTS	Grid Tagging Scheme
HAST	History Attention and Selective Transformation
IOG	IO-LSTM + Global Context
JERE-MHS	Joint Entity Recognition and Relation Extraction as a Multi-Head Selection

KB	Knowledge Base
KG	Knowledge Graph
LAMA	LAnguage Model Analysis
LM	Language Model
LOTN	Latent Opinions Transfer Networks
LSTM	Long Short-Term Memory
MAMA	Match and Map
MLM	Masked Language Modeling
MLP	Multilayer Perceptron
MRC	Machine Reading Comprehension
MVT	MultiView Training
NER	Named Entity Recognition
NLP	Natural Language Processing
NSP	Next Sentence Prediction
OATE	Opinion-Related Aspect Targets Extraction
OPE	Opinion Pair Extraction
OTE	Opinion Term Extraction
OTE	Opinion Triplet Extraction
OWSE	Opinion Words Span Extractor
PAOTE	Pair-Wise Aspect and Opinion Terms Extraction
PG	Product Graph
PKG	Product Knowledge Graph
POI	Points of Interest
PRC	Target-Opinion Pair Relation Classification
RDF	Resource Description Framework
RNN	Recurrent Neural Network
SDRN	Synchronous Double-channel Recurrent Network

SOTA	State-of-the-Art
SpanMlt	Span-Based Multi-Task Framework
SVM	Support Vector Machine
T5	Text-to-Text Transfer Transform
TAPT	Task-Adaptative Pretraining
TOWE	Target-Oriented Opinion Word Extraction
TPE	Tree-Structured Parzen Estimator
TSMSA	Target-Specified Sequence Labeling with Multi-head Self-Attention
URI	Unified Resource Identifier
W3C	World Wide Web Consortium
WWW	World Wide Web

LIST OF SYMBOLS

ω	Omega
α	Alpha
Δ	Capital Delta
δ	Delta
ϵ	Epsilon
ρ	Rho
γ	Gamma

CONTENTS

1 INTRODUCTION

This Chapter provides an overview of this thesis work. We present the context and motivation for performing this research and address the problem of enhancing Product Graph (PG), with subjective information extracted from users' reviews. Then, we list some Research Questions we address and some contributions we reach with the work here presented. In the end, we give a brief overview of the approach and how it was developed.

1.1 CONTEXT AND MOTIVATION

In recent years Knowledge Graph (KG) have gained popularity due to their potential to enable several Artificial Intelligence (AI) related tasks. This knowledge structure has been adopted by large players such as Google, Facebook, and Amazon to organize their products and their related content. The derivation of the structure when applied to organize products is called PG. Extensive research has been performed to improve the construction, enhancement, and application of these structures to various tasks, such as the ones related to Natural Language Processing, Information Retrieval, and Recommender Systems, among others.

KG in general, and PG in particular, are concerned with objective and factual data automatically gathered from one or more sources. A typical PG contains facts on the products and their characteristics. For instance, for mobile phones, the *brand*, the *dimensions*, and the *processor model* would be available in such a graph. However, with the rise of social media, a large amount of dynamic, subjective, and opinionated information on products and their characteristics became readily and widely available (LUO; HUANG; ZHU, 2019; ZHANG et al., 2021c). This creates an opportunity to aggregate subjective information to graph nodes, e.g., nodes corresponding to products and their attributes, to potentially enrich the knowledge of the product and ultimately improve many applications that can be enabled by the graph. This form of enrichment is particularly important for applications related to online shopping experience, e.g., recommendation, searching, comparison, pricing, etc (ARCHAK; GHOSE; IPEIROTIS, 2007; LIU, 2015).

The importance of considering subjective information besides factual information has been verified in many e-commerce applications. Indeed, considering other people's opinions

before purchasing a product is a common practice, especially since there are plenty of opinions available on the Web. According to a representative survey[1], 82% of Americans refer to online reviews when they first purchase a product, and 40% always refer to online reviews when purchasing products.

A timely and dynamic source of opinions on products is the user's reviews, published in forums, blogs, and e-commerce Web sites. These reviews have several interesting characteristics for customers in general. First, they are abundant, especially for the most popular products; second, they are often large and detailed, with opinions on many characteristics of the products; third, they are spontaneous, with users expressing themselves freely; finally, they are fresh, reflecting the current moods of customers.

Paradoxically, the same characteristics that make reviews useful as opinion sources also make them hard for an ordinary customer to handle. Indeed, to fully exploit their potential for decision-making, a potential buyer would need to frequently and carefully examine each review in a large set, looking for useful information on certain characteristics of interest, and coping with disparate expressions that refer to these characteristics.

As an example, a consumer interested in opinions on the screen of a particular cell phone would have to engage in a tedious and time-consuming browsing process over several reviews of this product. To avoid this, the user could try issuing a query using the term "*screen*". This is hardly effective since reviewers may have commented on distinct *aspects* of the screen, e.g., *resolution* or *size*, etc. without using the actual term. Also, the opinions about the searched characteristic can be mixed and yet the query can return tons of information. Hence, finding a way to summarize and organize this information directly on product characteristics (aspects) shows up as an outstanding solution to the problem.

The field of Opinion Mining (Sentiment Analysis) has already found some solutions for summarizing and extracting opinions. The field has gained much attention for some time now, however still there are some challenges to be overcome. Most works focus on the sentiment classification of the reviews, performing extraction of aspects or both tasks. Opinion terms, which are the words or phrases representing the users' attitudes or opinions explicitly, are usually used as components to build sentiment lexicons and assist the sentiment polarity assignment. Works looking to perform opinion word extraction, also known as opinion word expansion, opinion-term extraction, or opinion identification, usually perform this task in isolation (TAI; KAO, 2013; VICENTE; AGERRI; RIGAU, 2014;

[1] <http://www.pewinternet.org/2016/12/19/online-shopping-and-e-commerce>

IRSOY; CARDIE, 2014; LIU; JOTY; MENG, 2015). Nonetheless, aligning actual opinion words with their respective opinion targets usually can be more helpful than just reducing this relation to polarities. Previous works have tried to perform this task as a coextraction (WANG et al., 2016; WANG et al., 2017; LI; LAM, 2017), but without considering the explicit relations between these two opinion components.

Trying to solve this problem, considering the interactions between aspect and opinion terms explicitly, the Target-Oriented Opinion Word Extraction (TOWE) task was first introduced by Fan et al. (2019) and consists of extracting opinion words for a given opinion target to retrieve aspect-opinion pairs. The main challenge of this approach is that the aspect must be known in advance.

Following the work of Fan et al. (2019), Chen et al. (2020) introduce the Aspect-Opinion Pair Extraction (AOPE) task to solve the same problem of aspect-opinion pair extraction, but the main difference for TOWE is that the aspect does not must be known in advance. Similar to AOPE, Zhao et al. (2020) propose the Pair-Wise Aspect and Opinion Terms Extraction (PAOTE) which also consists in retrieving aspect-opinion words as pairs. The three works cited above have paved the way to pair extraction of aspect and opinion terms. Each work has treated the problem using a different approach: as a sequence-labeling problem, as the joint learning of entity extraction and relation detection, and as a multi-task framework based on spans of text. We highlight each one of these approaches in Chapter 3. In general, these works are fully supervised and require manually built and curated training examples which are expensive and not scalable.

We conclude that investigating methods to extract and organize opinions both for people and downstream applications automatically is an essential topic for the academy and industry. This conclusion is the main motivation behind the task *aspect-based opinion summarization* (MOUSSA; MOHAMED; HAGGAG, 2018), whose goal is to generate summaries of opinions according to the product aspects they refer to. Hence, Product Knowledge Graphs appear as a great choice for organizing this type of subjective information. Furthermore, pre-trained Language Model (LM) (PETRONI et al., 2019) aligned to In-Context Learning (BROWN et al., 2020) have improved State-of-the-Art (SOTA) works for various tasks. LM are large deep neural network models trained on large corpora to learn language structures, syntax, and semantics. In-Context Learning is a paradigm to explore these large models using only a prompt with instructions or examples of a target task. The success of these models on different tasks, suggests that they could also provide a

new path to build synthetic training examples. More specifically, these synthetic examples can be used to improve supervised SOTA models for aspect-opinion tuples extraction or even replace these supervised methods by prompting extraction.

1.2 THE PROBLEM

Given the exposed in the previous Section, this thesis tries to solve the problem of enhancing PG with subjective information extracted from users' reviews. We have built a pipeline to solve this problem by mapping aspect-opinion values extracted from opinionated reviews to targets in a PG. For this, we have explored Distant Supervision for weakly-supervised learning, a traditional unsupervised method for Opinion Mining, and Deep Learning for Opinion-Target Classification. Additionally, we have built and evaluated a generator of synthetic opinionated triples using an adapted LM and In-Context Learning. These triples are composed of synthetic opinionated reviews, the aspects mentioned in them, and the opinion words related to these aspects. This second study investigates the augmentation of TOWE and AOPE tasks with the synthetically generated samples, which is an initial effort to improve the aspect-opinion pair extraction in the first pipeline, or even replace Supervised Learning approaches with In-Context Learning.

1.3 RESEARCH QUESTIONS

In the following, we list some Research Questions that have guided this study.

RQ1 Is it feasible to enhance Product Knowledge Graphs with subjective information extracted from user reviews without relying on manually labeled training data?

RQ2 Can we map the pairs of opinion words and opinion targets extracted from users' reviews to product targets in Product Knowledge Graphs?

RQ3 Can we explore data augmentation, by automatically building training examples, to improve the Aspect-Opinion Pair Extraction task?

1.4 OVERVIEW OF THE SOLUTION

We have explored two complementary stages to solve the problems and answer the questions raised previously. First, we have built a pipeline to perform semi-supervised learning on opinion extraction and map them to a Product Graph (PG), which solves the problem and answers the research questions *RQ1* and *RQ2*. Then, we performed a study to investigate the use of In-Context Learning for augmenting Opinion Mining tasks, which was an effort to investigate an improvement of the first solution and answer the research question *RQ3*.

As stated previously, the pipeline, which we call **P**roduct **G**raph enriched with **Opi**nions (PGOpi)[2], aims to enhance PG with subjective user opinions. It is composed of a traditional unsupervised approach for Opinion Mining (more details in Sections 3.1 and 4.1) and it is based on a Distant Supervision (DS) paradigm. The DS paradigm explores embedding similarity between product targets, from an existing Product Knowledge Graph, and the opinions extracted from the users' reviews (more details in Section 4.2). Depending on a given similarity threshold, the information is labeled as a training example. Hence, the product targets from the PG are given as labels to the extracted opinions. The labeled training data are used for training an Opinion-Target Classifier (details in Section 4.3), that aims to classify (map) unseen extracted opinions to targets of products. This Opinion-Target Classifier is based on a Deep Learning architecture.

According to experimental evaluation, we found that our semi-supervised pipeline shows comparable performance against a SOTA work on the same problem which is fully supervised (the experiments and evaluation are given in Chapter 5). Although the problem is rather new and only one work was found trying to solve it, we show that there is a large interest in organizing opinions around product targets and the proposed solution can be applied both for the academy and industry using Knowledge Graphs.

Since the proposed pipeline for mapping opinions to PG' targets has relied mostly on a traditional unsupervised approach for Opinion Mining, in the second part of our work we aimed at improving this Opinion Mining step. For this, we performed an investigation on using the In-Context Learning of pre-trained Language Models to autonomously build synthetic training examples. The aim of this investigation was to study the augmentation

[2] The PGOpi pipeline is already published in Moreira et al. (2022) and we have used part of the paper material in this thesis work.

of Opinion Mining tasks with synthetically generated training examples. Hence, we build **Syn**theti**C** **Op**inion**A**ted **T**ripl**E**s (SYNCOPATE)[3] generator, a framework for building opinionated triples that could be used to improve the training of supervised models for Opinion Mining. The main intuition was to eventually replace the current unsupervised module for opinion mining in the current pipeline with existing SOTA supervised models or even perform extraction using prompts on the adapted pre-trained Language Model.

More specifically, in the second part of this thesis work, we have investigated whether In-Context Learning approaches allow building good quality opinionated triples to augment SOTA models in the TOWE and AOPE tasks. However, there were some challenges while trying to generate synthetic opinionated triples. First, the generated sample should hold a strict format: the first element should be an opinionated sentence, the second element should be an aspect mentioned in it, and the third one should be an opinion about that aspect. Second, we expected for the LM to generate texts in the domain of interest. To this end, we have to adapt a pretrained Language Model to our task using Task-Adaptative Pretraining (TAPT) (see Section 6.1). Then, we used the In-Context Learning prompting strategies to build the synthetic triples (Section 6.2). Finally, it was required a post-processing step to clean and label the generated triples (Section 6.3).

We have evaluated the synthetic triples by augmenting benchmark datasets with them (Chapter 7). We have used the augmented datasets to train two SOTA models for TOWE and two for AOPE. Also, we asked three human raters to evaluate the quality of those synthetic triples. In summary, we found that training the models only with synthetic triples generated by our approach allows the performance to surpass the models trained with a small set of manually built triples in the majority of the scenarios. The augmented training sets have enhanced the performance of the models in three scenarios and they have not significantly hurt the performance in 10 (ten) out of 16 (sixteen) evaluated scenarios. Hence, these results aligned to the human evaluation of the triples show that the approach here presented can build good quality opinionated triples. Also, it points out that In-Context Learning on a pre-trained Language Model after performing TAPT with only a few labeled instances is promising for Prompting the Extraction of opinionated tuples without relying on prompting engineering.

The first and the second part of this thesis work can be easily integrated. The first

[3] Currently, part of the SYNCOPATE's material presented in this thesis work was submitted for publication.

part consists of a pipeline to build labeled data to train a classifier that maps extracted opinions to targets in a PG. The second part was developed in order to replace the opinion extraction model from the first part. The previous extraction model is based on rules that are not easily scalable and require constant adaptation to rightfully extract pairs of aspects and opinions. The second part of this thesis adapts a Language Model to build opinionated triples to augment existing models for aspect-opinion pair extraction.

1.5 WORK ORGANIZATION

In addition to this Chapter, this work is organized into four parts with the sections disposed of as follows:

- **Part I - Background & Literature**

  Chapter 2 introduces some key concepts, structures, and tasks related to the development of this thesis work. Section 2.1 presents the basics of the Opinion Mining field, its many branches of study and presents the formalization of a user's review, the source of subjective and unstructured information of this study. Section 2.2 formalizes Knowledge Graphs (KG), discusses how they are built, and introduces its derivation, the Product Graphs (PGs), largely applied in the e-commerce scenarios. Section 2.3 presents some of the main pre-trained Language Models in the literature. We use these models to analyze data augmentation in the Opinion Pair extraction task performed by the built solution. The last Section 2.4 presents Distant Supervision (DS), the approach applied in our model for weakly labeling of training examples.

  Chapter 3 gives a review of traditional and state-of-the-art approaches applied to the tasks related to this thesis work. Section 3.1 presents some traditional and recent approaches performing Aspect-Based Sentiment Analysis (ABSA), more specifically the ones focusing on Aspect and Opinion Terms Extraction. Section 3.2 introduces some recent works that have tried to structure subjective information, as well as works directly related to Product Knowledge Graphs construction and enhancement. Section 3.3 presents works exploring pre-trained Language Models as sources of supervision for other tasks, trying to retrieve information from their tuned parameters, and also exploring In-Context Learning for Data Augmentation. Additionally,

we discuss some SOTA works aiming at improving Aspect-Based Sentiment Analysis (ABSA) tasks with Data Augmentation. In the last Section 3.4, we benchmark the differences and similarities between SOTA works and our **P**roduct **G**raph enriched with **Opi**nions (PGOpi) approach, segmenting these similarities and differences by each module of the pipeline. We highlight some improved points and the contributions of our work against the one already existent. We also list the differences between our **Syn**theti**C** **Op**inion**A**ted **T**ripl**E**s (SYNCOPATE) generation framework against works performing data augmentation for ABSA tasks.

- **Part II - Pipeline for Enriching Product Graphs with User Opinions**

  Chapter 4 presents our effort to build the model for enriching Product Knowledge Graphs with user opinions, which we call **P**roduct **G**raph enriched with **Opi**nions (PGOpi). Section 4.1 presents the unsupervised approch applied for opinion mining. Section 4.2 presents our Distant Supervision Strategy applied to automatically label training instances. Section 4.3 presents the Deep Learning architecture build to classify (map) unseen extracted opinions to product targets in the PG.

  Chapter 5 presents the experiments and evaluation performed to validate our **P**roduct **G**raph enriched with **Opi**nions (PGOpi) pipeline. Section 5.1 gives the experiments setup configuration as training and ground truth datasets collection, construction, and labeling used for validation. It also presents the benchmarking approaches, hyper-parameters optimization, word embeddings setup and evaluation metrics. Finally, the experimental results are shown and discussed in Section 5.2.

- **Part III - In-Context Learning approach for Data Augmentation of Aspect-Opinion Pair Extraction methods**

  Chapter 6 presents the methodology for generating synthetic opinionated triples, which we call **SYN**theti**C** **O**pinion**A**ted **T**ripl**E**s (SYNCOPATE) generation framework. Section 6.1 shows the concept of TAPT and how we use it to continue the pretraining of the Language Model and incorporate task and domain knowledge into it. Section 6.2 discusses the zero-shot, one-shot, and few-shot strategies of the In-Context Learning approach and how we apply them to build the synthetic triples. Section 6.3 shows the post-processing and automatic labeling steps we perform to get the generated text and format it as a labeled triple to be used by SOTA models.

Chapter 7 presents the experimental evaluation performed to validate our **SYN**theti**C** **O**pinion**A**ted **T**ripl**E**s (SYNCOPATE) generation framework. Section 7.1 introduces the benchmark datasets we augment with the synthetically generated opinionated triples, the approaches on TOWE and AOPE tasks we use to evaluate the impact on performance after augmenting the original data with synthetic ones, and the hyper-parameter settings we employ to the models; Section 7.2 presents the statistics of the generated triples we built using SYNCOPATE; and Section 7.3 discusses the experimental results we obtained while running the TOWE and AOPE approaches with augmented data. We also discuss the results of a performed human evaluation on the triples generated by our approach.

- **Part IV - Conclusion**

  Chapter 8 concludes the thesis, showing its contributions and limitations. The chapter finishes by discussing future directions for this work, the articles published during the Doctorate course, and the articles in review.

2 BACKGROUND

In this Chapter we introduce some basic concepts related to the techniques and approaches applied in this Thesis work. Section 2.1 presents definitions of elements and tasks involved in the Opinion Mining field. Section 2.2 presents Knowledge Graph (KG) and Product Graph (PG). Section 2.3 shows state-of-the-art language models that are currently largely applied to the Natural Language Processing (NLP) field of study. Additionally, we employ a Distant Supervision approach to build labels for training examples, in Section 2.4 we give a brief overview of the technique.

2.1 OPINION MINING

The field of Opinion Mining, also known as Sentiment Analysis, has been thoroughly studied and evaluated, mostly because of its assistance in incrementing decision support systems for institutions, private or public. A large amount of information available on the Web, mostly coming from social networks, blogs, forums, news, and especially e-commerce websites, is a valuable asset to be considered as a direct feedback source from users to companies providing products or services or even to know the popular opinions on trending topics. However, this information is usually found as natural language text. This type of information is hard to be processed and consumed by humans, given its large amount; and also by computers, given the challenges already known by the field of Natural Language Processing. As defined by Liu (2015), Sentiment Analysis (or opinion mining) is the field of study that analyzes people's opinions, sentiments, appraisals, attitudes, and emotions toward entities and their attributes expressed in text. These entities can be products, services, organizations, individuals, events, issues, or topics.

Earlier works on sentiment analysis have focused mainly on a coarse-grained approach which is detecting the opinion targets, the sentiment polarity of the reviews, the opinion holder, and the time the reviews were posted. While the two last tasks are more simple to be performed, most works focus on the two first tasks, working on them jointly or focusing on only one of them. Liu (2015) splits Sentiment Analysis into three levels: Document-level, Sentence level, and Aspect level. The two first levels are worried about the polarity of the review and with one single target to which the review is about. The Aspect level looks

at the reviews with a more detailed view, looking for specific aspects and the sentiments related to each one, allowing a fine-grained understanding of the review.

The task now called Aspect-Based Sentiment Analysis (ABSA) has received more attention given the detailed analysis provided. Works have developed ABSA systems for movie reviews, services, restaurants, electronics, among others, performing tasks as Aspect Term Extraction, Aspect Term polarity, Aspect Category detection, Aspect Category polarity, Sentiment Classification, and Opinion Word Extraction. Below we introduce some basic concepts and tasks of this field.

Listing 1 – Sample of a review and its components.

I purchased a **Shazam X8** one month ago. It has a **great case**, **great screen resolution** and **contrast**, a **smooth screen** response, **simple device**, but the **software is amazing**.

Polarity: positive
Target: Shazam X8
Aspect: case, screen, screen resolution, contrast, device, software
Opinion Words: great, smooth, simple, amazing
Aspect Category: Screen
> Aspects: screen, screen resolution, and contrast

Source: Created by the author

2.1.1 Basic Concepts

The Listing 1 shows an example of user review and its components. Below we define each one of these components.

- **Polarity (or Sentiment):** the sentiment towards a sentence, paragraph, or text span independently of the entities and entities' characteristics present in the text. This sentiment is usually categorized as positive, negative, neutral, or conflicting;
- **Opinion Target (or Aspect):** Also known as Aspect Terms, as defined by Wu et al. (2020b) "*...are the words or phrases in the sentence representing features or entities toward which users show attitudes*". Usually, when representing entities, these components are called Targets. When representing features or characteristics of an entity, these components are called Aspects.

- **Opinion Word:** the terms or words in the sentence used to explicitly express an attitude, sentiment or feeling on some aspect.
- **Aspect Category:** is a category of similar parts or attributes of the product, as defined by Cheng et al. (2017).

2.1.2 ABSA Tasks

Aspect-Based Sentiment Analysis tasks consist of a fine-grained summarization of a given review, where the opinions expressed in the text can be associated to different entities and features. The term was coined by the SemEval 2014 Task 4 (PONTIKI et al., 2014), but this type of analysis was first introduced by (HU; LIU, 2004; LIU, 2010) under the name of feature-based sentiment analysis. According to (PONTIKI et al., 2014) ABSA aims to identify the aspects of the entities being reviewed and to determine the sentiment the reviewers express for each aspect. Below we highlight some specific tasks there are largely investigated under this study field.

- **Aspect Term Extraction (or detection):** As defined by Pontiki et al. (2014), given a set of review sentences, the task is to identify all aspect terms present in each sentence;
- **Aspect Term Polarity:** Also introduced by Pontiki et al. (2014), assuming that the aspect terms are known in advance the task consists in determine the polarity of each aspect term;
- **Aspect Category Detection:** introduced by Pontiki et al. (2014), given a set of categories and review sentences the task consists in identify the aspect categories discussed in each sentence;
- **Aspect Category Polarity:** introduced by Pontiki et al. (2014), given the aspect categories present in each review sentence, the task consists in detect the polarity of each category;
- **Opinion Words Extraction:** the task of extracting opinion expressions (words) oriented to a specific target (aspect);

- **End-to-end ABSA:** The task of jointly detect aspect terms/categories and the corresponding aspect sentiments/polarities.

2.1.3 Reviews

In this work of thesis we are interested in performing ABSA on users' reviews on products of e-commerce websites to enrich PG with subjective information. Here we formalize the definition of review, how they are composed and the type of opinions there can be expressed in them.

A *review* is a text posted by a user on an e-commerce website, usually reporting their experience with a specific product, which we call the *target entity* of the review. Each review is composed of a set of *sentences*. Sentences that express factual information are called *objective* sentences, while sentences that express personal feelings or beliefs are called *subjective* or *opinionated* sentences. In this work, we are particularly interested in the latter because they represent the reviewer's opinions of a product. A single sentence may have multiple opinions. For example, the sentence *"The screen is bright, but I'm not satisfied with the performance"* has two different opinions: a positive opinion regarding the display and a negative opinion about the processor.

An opinionated sentence can further be classified as *comparative* or *direct*. A comparative sentence expresses a relation of similarity or difference between two or more products. The sentence *"the camera of the Cyclone is much better than Shazam"* is an example of a comparative sentence. A direct opinionated sentence expresses an opinion directly on a characteristic or part of the product, or on the product as a whole. The sentence *"The camera of the Cyclone is fantastic"* is an example of a direct opinion. As our goal is to enrich each product with the opinions of users regarding the specific product, we decided to eliminate comparative sentences. More precisely, a *direct opinionated sentence* is a sentence in which an opinion is expressed directly on one or more characteristics of a product, or on the product as a whole.

2.2 KNOWLEDGE GRAPHS

Knowledge Graphs (or Knowledge Bases) are graphs in which nodes represent real-world objects and edges represent relations between them. These objects represented

by nodes on graphs are also called Entities. According to Balog (2018), an entity is a uniquely identifiable object or thing, characterized by its name(s), type(s), attributes and relationships to other entities. Although the terms Knowledge Base and Knowledge Graph represent the same concept, Balog (2018) highlights that *"when the emphasis is on the relationships between entities, a knowledge base is often referred to as a knowledge graph."*.

Knowledge graphs are important to organize the data, allowing an intuitive exploration across its structures. Besides, they are crucial for semantically explore the data meanings and consequently enable the development of tasks oriented to knowledge, i.e., it enables a better understanding and use of the available data by machines.

The information used to build knowledge graphs can come from several sources. It can be obtained from structured sources (e.g. Wikipedia Infoboxes, tables of product specification from online stores, databases, social networks, among others), unstructured texts (e.g. news articles, Wikipedia articles, any site on the World Wide Web (WWW), posts on social media, and so on), and it can also come in multimedia form (as images and videos).

Currently, the main form to represent and exchange obtained data to knowledge graphs structure are through the use of Semantic Web standards as Resource Description Framework (RDF)[1]. RDF is a World Wide Web Consortium (W3C) specification for data modeling and interchange on the Web, is currently used for content description, modeling, and knowledge management. RDF documents are composed of three main components: resource, property, and value. With these components at hand, the RDF standard can replicate existing entities links on the Web. Using Unified Resource Identifier (URI) to represent resources, as well as the relationships between them.

In Definition 1 we formalize a KG as will be applied by this thesis work.

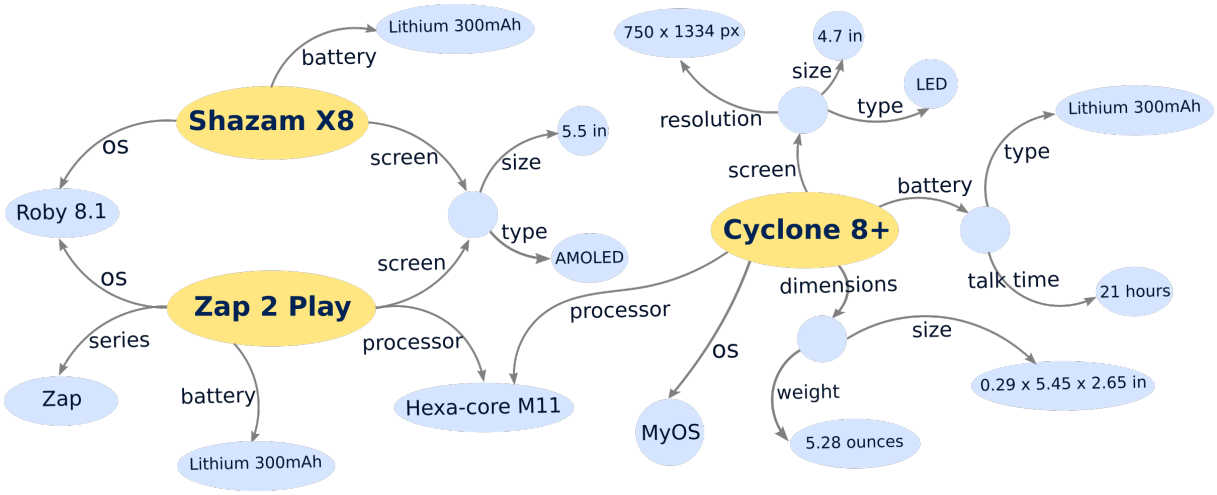
Definition 1 *A* knowledge graph *G is a directed graph $\langle V, E, L_V, L_E \rangle$, where V is a set of nodes, and $E \subseteq V \times V$ is a set of edges, L_V is a set of nodes labels and L_e is a set of edge labels. Each node $v \in V$ represents an entity with label $\ell(v) \in L_V$ and each edge $\langle v, w \rangle \in E$ represents a relationship between entities v and w with label $\ell(\langle v, w \rangle) \in L_e$.*

[1] <https://www.w3.org/RDF/>

2.2.1 Product Graphs

Currently, Product Knowledge Graph (PKG) or simply PG have been proposed to structure data on consumer products in the e-commerce scenario. In comparison to the traditional product catalogs, the PG format is more flexible. The data for building these structures are usually provided by manufacturers (KIM, 2017) or gathered and extracted from online sources. Representative examples of PG are the ones by Amazon.com (DONG, 2018), Walmart.com (XU et al., 2020), and Alibaba (LI et al., 2020). PG, as well as traditional KG, are concerned with objective and factual data gathered from one or more sources. A typical PG contain facts on products and their characteristics.

Figure 1 – Example of a Product Knowledge Graph on the Smartphone category. These are fictitious smartphone models with features similar to real ones.



Source: Adapted from Moreira et al. (2022)

In this work, for modeling PG, we adopt a data model similar to NAGA (KASNECI et al., 2008), which is also used by other authors (SONG; WU; DONG, 2017; GUO et al., 2018). A *product graph* is a knowledge graph in which semantic is assigned to its elements and constraints are imposed on them. We present a formal definition of a product graph in Definition 2, and illustrate the concept using a simple example in Figure 1.

Definition 2 *Let* $PG = \langle V, E, L_V, L_E \rangle$ *be a knowledge graph. We say that PG is a* product graph *if the following constraints apply:*

i) Any node $v \in V$ *is either a* product node*, when it represents a product p from a product catalog, or a* attribute node*, when it represents an attribute a from some*

product from the catalog also represented in PG. Attribute a can be either a simple *attribute or a* composite *attribute.*

ii) If v is a product node*, its label $\ell(v)$ is the description of the product. Otherwise, if v is a attribute node, its label $\ell(v)$ is the value of the attribute. However, if v represents a composite attribute, its label is empty.*

iii) If v is a product node, then there is no incoming edge to it. Otherwise, if v is an attribute node, there is exactly one incoming edge $e \in E$ to it originating either from a product node or from a composite attribute node. The label $\ell(e)$ is the name of the attribute in the catalog.

iv) If v is a product node, then there is at least one outgoing edge from it to an attribute node. Otherwise, if v is a composite attribute node, there are at least two outgoing edges from it, both to attribute nodes. However, if v represents a simple attribute, then there is no outgoing edge from it.

In Figure 1, we illustrate a simple product graph. The bigger ellipses depict three products: Shazam X8, Zap 2 Play, and Cyclone 8+[2], all of them from the *Cell Phone* category. Notice that, as stated in Definition 2, product nodes are source nodes and there are no incoming edges to them.

The minor ellipses with at least one incoming edge represent attribute nodes. The nodes representing simple attributes have a non-empty label, whereas composite attribute nodes have an empty label. Notice that only composite attribute nodes have outgoing edges. Simple attribute nodes, on the contrary, are sink nodes. For instance, one of the composite attributes for the product Cyclone 8+ is screen, which is composed of three single attributes: resolution, size, and type. All these are attributes of the same product Cyclone 8+.

To model the case where two products have the same attribute, we allow attributes to have more than one incoming edge. For instance, in Figure 1, both Shazam X8 and Zap 2 Play have the same screen and the same OS (operating system). For the sake of consistency, we enforce that $\ell(\langle x, a\rangle) = \ell(\langle y, a\rangle)$ for any attribute a. That is, all incoming edges to an attribute must have the same label, meaning that the same attribute has always the same name.

[2] These are fictitious smartphone models with features similar to real ones.

As can be seen in Figure 1, the three products illustrated in the figure have different sets of attributes, which means we do not impose a rigid schema for product graphs. Also, attributes with the same name, such as screen, may have a different structure in distinct products. In this work, we assume that all products in a product graph belong to the same product category without loss of generality. For instance, in the graph of Figure 1, all products are from the Cell Phone category. As a result, the attributes of the products are not expected to differ much.

2.3 LANGUAGE MODELS

Language Models are probabilistic models using self-supervised training to model the probability distribution over text. This self-supervised training is called pre-training and allows the model to learn word meaning, sentences syntax, and world-sense knowledge. These language models have shown significant results for many NLP tasks and are currently the main focus of studies in Deep Learning for NLP. In this section, we show some of these state-of-the-art models, how they are approached and built.

The essence of most state-of-the-art language models is the Transformer. Introduced by Vaswani et al. (2017), it relies on attention mechanisms to get dependencies between input and output sequences. Currently, Attention mechanisms are a crucial piece of Deep Neural Network (DNN) models presenting impressive results on various tasks (BAHDANAU; CHO; BENGIO, 2015; KIM et al., 2017). These little pieces composing the Transformer architecture help solve the information bottleneck problem introduced by non-linear transformation layers on standard neural networks. The problem consists of a standard neural layer trying to produce a fixed-dimensional hidden representation for a large input sequence. This transformation is hard to perform because of the large number of interactions between the sequence components. The Attention mechanisms perform a soft-selection over the input and allow the network to build a hidden representation scaling at the size of the source.

Unlike previous works that mostly join the attention mechanisms with recurrent networks, the Transformer architecture is composed solely of attention mechanisms. This architecture allows more parallelization and models the dependencies without the distance limitation of recurrent models. The architecture is mainly composed of self-attention mechanisms, which relate different positions of a single sequence to compute its repre-

sentation. The model presents an encoder-decoder structure, where the encoder maps the input sequence, and the decoder generates the outputs given the representations obtained by the encoder. Both modules are layers of stacked self-attention mechanisms with position-wise fully connected layers, employing residual connections and layer normalization. The decoder differs from the encoder in including attention mechanisms over the representation from the encoder output. The Transformer works with the idea of Multi-head attention, which consists of combining different linear projections of the attention function "h" times, called heads. To include sequential information into the model, the authors inject positional encodings into the input embeddings of both encoder and decoder modules. This information allows the inclusion of the relative or absolute position of the tokens in the sequence.

The Bidirectional Encoder Representation from Transformer (BERT), proposed by Devlin et al. (2018), as stated in the model's name, is made of Transformers. Working as a Language Model, the architecture of BERT is composed only by the encoder. BERT considers the bidirectional flow of the sequence to learn a deeper sense of language context. BERT is available under two model architectures: bert-base, composed of 12 Transformers blocks, 768 hidden units, and 12 self-attention heads, presenting 110M parameters in total; and BERT-large, composed of 27 Transformers blocks, 1024 hidden units, and 16 self-attention heads, totalizing 340M parameters.

Pre-trained BERT models can be applied to any other task without parameter tuning or fine-tuning. However, they must be fit to any other specific task or dataset. The training of BERT applies two strategies. One strategy bases on masking tokens from the input sequence and then trying to infer those masked tokens, this strategy is called Masked Language Modeling (MLM). The other strategy bases on Next Sentence Prediction (NSP) that aims predicting if the second sentence in the input connects to the previous one. The model introduces three markers to perform these training objectives: [MASK] to indicate masked tokens, [CLS] to indicate the start of the sentence (input), and [SEP] to mark the end of the first sentence and start of the next as well as the end of the second sentence (the end of the input).

The systematic study of Raffel et al. (2019) presents the Text-to-Text Transfer Transform (T5), which consists of a framework that studies the application of a model without any further modifications to perform various tasks. The applied model architecture does not differ too much from the one proposed by Devlin et al. (2018). Additionally, in their

experiments, the authors find out that the encoder-decoder architecture works best for the analyzed tasks. The main contributions of this work are the experiments performed over various tasks and the built pre-trained dataset. The authors scale the pre-training of the models either by increasing the number of parameters (up to 11 Billion) and with a large heuristically cleaned corpus from the Common Crawl web dump[3]. This final pre-trained dataset, called Colossal Clean Crawled Corpus (C4), contains 745 GigaBytes of cleaned unlabeled text.

Radford et al. (2018) introduces the Generative Pre-Training (GPT) framework. It also consists of a model based on Transformer. However, it differs from previous approaches, like BERT, because it transforms the input during the fine-tuning phase. These transformations force the model to be task-aware, thus requiring fewer changes to the model architecture. The input transformations are task-specific and include textual entailment, similarity, question answering, commonsense reasoning, and classification. The authors show that it is possible to learn significant world knowledge and long-range dependencies to solve discriminative tasks by pre-training a transformer architecture on a diverse corpus with long stretches of contiguous text.

In Radford et al. (2019), the authors present a new version of GPT, the GPT-2, a larger model also based on Transformer, that achieves state-of-the-art results on 7 out of 8 tested language modeling datasets. The authors introduce in this model the zero-shot setting that consists of performing a given task with the pre-trained model without any parameter or architecture modification, i.e., no supervision. The architecture of the model bases on the Transformer with 1.5B parameters, expanding the model introduced by Radford et al. (2018). The authors expanded the vocabulary to 50,257 words and increased the context size from 512 tokens to 1024.

Another extension of GPT is presented by Brown et al. (2020), the GPT-3. The model is also an autoregressive language model but larger than all the previous, with 175 billion parameters. The authors' hypotheses that since the model's parameters can learn skills and tasks, the ability to improve in-context learning lies in the increase of the number of parameters and training corpus.

In-context learning consists of conditioning the Language Model (LM) to receive some instruction or only a few examples of the task to be performed and then perform the tasks just by predicting what comes in the sequence. Hence, the learning paradigms called few-

[3] <http://commoncrawl.org/>

shot, one-shot and zero-shot learning are first coined by (BROWN et al., 2020). The first term consists of providing the language model with some few observations of some task. One-shot refers to providing the model with only one sample of the task while Zero-shot consists of providing the model with any demonstration of the task, just an instruction. The authors test the performance of their proposed model only in the few-shot setting that consists of performing different tasks without any gradient updates or fine-tuning, where a few demonstrations are given to the prompt to iterate with the model.

The authors build a pre-training unlabeled corpus of 499 Billion tokens, obtained from a filtered version of Common Crawl, WebText2, Books, and English Wikipedia. The authors evaluate the model over two dozen NLP datasets from different tasks, achieving strong performance. Some of these evaluated tasks are unscrambling words, using a novel word in a sentence, performing 3-digits arithmetic, and news article generation. Although the model shows an impressive contribution to the NLP community, currently, there are some concerns on ethical aspects of the framework, mostly coming from the training data. The training data can introduce biases to the model, generating stereotyped or prejudiced content related to gender, race, religion, among others.

2.4 DISTANT SUPERVISION

Distant Supervision (DS) is a paradigm usually applied by Relation Extraction methods to build training datasets. Instead of relying on human handcrafted features, patterns or manual annotation of training examples, DS starts from the assumption that any sentence containing a pair of entities from a known relation is likely to express that relation. Mintz et al. (2009) have used Freebase aligned with 1.2 million Wikipedia articles to generate training sets of 102 Freebase relations and entity pairs that participate in those relations.

Takamatsu, Sato and Nakagawa (2012) highlights that, although DS is an attractive approach to heuristically generate a large number of labeled data, when compared with the limitations of supervised approaches, it can generate noisy labeled data and cause poor extraction performance. They state that this can happen when the given entity pair express more than one relation on target text. As an example, in the context of Relation Extraction, the pair (Michael Jackson, Gary) expressing *place_of_birth* relation on the Knowledge Base (KB) might be matched with the sentence "*Michael Jackson moved from*

Gary." which does not rightfully represents the relation *place_of_birth*. On the other hand, the sentence "*Michael Jackson was born in Gary*" is a good representation for the relation. Roth et al. (2013) have organized DS approaches into three basic principles:

- At-least-one constraint: it considers that at least one sentence labeled positive by the DS assumption actually represents a true positive sample (RIEDEL; YAO; MCCALLUM, 2010; HOFFMANN et al., 2011; SURDEANU et al., 2010);
- Topic-based models: makes use of a generative model to discriminate between patterns that are expressing the relation and ambiguous ones (ALFONSECA et al., 2011);
- Pattern correlations: make use of a probabilistic graphic model containing hidden variables to model whether a pattern expresses a relation or not (TAKAMATSU; SATO; NAKAGAWA, 2012).

3 LITERATURE REVIEW

In this Chapter, we discuss the areas directly related to this thesis work. In Section 3.1, we overview some works performing Aspect and Opinion Terms Extraction in the Opinion Mining field. First, we introduce some traditional approaches. Then, we detail the most recent trend in the field: The Aspect-Opinion Pair Extraction (AOPE). In Section 3.2, we discuss some works trying to structure subjective information and enhance Product Graph (PG). In the end, Section 3.3 presents works exploring Language Models as source supervision for data labeling, answering questions, performing information retrieval to build knowledge graphs, exploring attention weights to find patterns, and exploring In-Context Learning for data augmentation. We finish the Chapter in Section 3.4 by performing a comparative analysis between state-of-the-art works on the tasks directly related to this work of thesis and our two pipelines: **P**roduct **G**raph enriched with **Opi**nions (PGOpi) for mapping opinions to a PG and the one for **SYN**theti**C** **O**pinion**A**ted **T**ripl**E**s (SYNCOPATE) generation.

3.1 ASPECT AND OPINION TERMS EXTRACTION

In this section, we overview some traditional and recent works on Aspect-Based Sentiment Analysis (ABSA) that performs the extraction of both Aspects and Opinion terms. First, we discuss the emergence of some classical approaches and how they address the tasks. Then, we discuss in detail the state-of-the-art works for the Aspect-Opinion Pair Extraction (AOPE) task, which has attracted too much attention in the last few years.

3.1.1 Introduction to Traditional Approaches

Hu and Liu (2004) have first studied the aspect extraction task. The authors introduce a set of rules based on statistical observations to approach the problem. Since then, tons of works have been proposed to improve the task and include the extraction of opinion terms. Some traditional works relied on building unsupervised models based on distance-rule (HU; LIU, 2004), dependency-rule (ZHUANG et al., 2006), and syntactic and semantic rules (PORIA et al., 2014; TAI; KAO, 2013; VICENTE; AGERRI; RIGAU, 2014). Recently,

with the advance of supervised machine and deep learning techniques, some works have emerged applying these techniques to the problem of aspect extraction (XU et al., 2018; LI et al., 2018), opinion terms extraction (IRSOY; CARDIE, 2014; LIU; JOTY; MENG, 2015), and the coextraction of both tasks (WANG et al., 2016; WANG et al., 2017; LI; LAM, 2017; HE et al., 2019). Below we give more details from some of these works.

Hu and Liu (2004) perform feature-based opinion summarization in three steps: i) mining product features commented by customers; ii) identifying opinion sentences in the review and deciding whether the polarity of negative or positive; and iii) summarizing the results. The authors focus only on review sentences that explicitly contain references to some product features. For that, they rely on the syntactic structure of the sentences to extract nouns and noun phrases as product features (or aspects). Hence, a product feature must explicitly appear in the sentence as a noun or noun phrase. The authors also filter all retrieved aspects by frequency, keeping the ones mentioned by most reviews. The work also performs opinion terms extraction but only to infer the polarity of the whole review sentence. For this, the authors assume that opinion words are always adjectives.

Unlike Hu and Liu (2004), Zhuang et al. (2006) retrieves explicit and implicit feature-opinion pairs. However, the authors rely on a keyword list with features and opinions and extract information from reviews using this keyword list aligned with grammatical rules. These grammatical rules are dependency relation templates defined by the authors.

Similar to Zhuang et al. (2006), Poria et al. (2014) explores external information to build an opinion lexicon for implicit and explicit aspects and opinions. Explicit aspects refer to characteristics of the product that are mentioned directly in the review, as for example in the sentence *"I love the **touchscreen** of my phone but the **battery life** is so short"* the aspects touchscreen and battery life are explicit. Implicit aspects are mentioned indirectly by using other words and can be inferred from the context, for example in the sentence *"This camera is sleek and very affordable" sleek* refers to the appearance aspect and *very affordable* refers to the price of the camera. An implicit aspect corpus proposed by Cruz, Gelbukh and Sidorov (2014) and the semantics extracted from SenticNet (CAMBRIA; OLSHER; RAJAGOPAL, 2014) compose the external information explored by the authors. This common-sense knowledge is integrated with the sentence dependency trees by various rules that allow the extraction of aspect-based opinions.

Other works were looking to extract opinion words alone, but most focused on building sentiment lexicons with polarities. Tai and Kao (2013) tries to build a sentiment

lexicon automatically by extracting opinion words and autonomously assigning polarity to them. The approach applies traditional Natural Language Processing (NLP) techniques to extract candidate opinion words from an unlabeled tweet corpus as part-of-speech and lemmatization. The authors build word graphs using WordNet, conjunction rules, and a function to compute the similarity between two target words. The approach allows the propagation of polarity labels from seed words to unlabeled words in the corpus. Vicente, Agerri and Rigau (2014) is another unsupervised approach that explores WordNet to build polarity lexicons. It selects seed words with polarities and adapts the PageRank algorithm to propagate over the word graph to assign polarities to unlabeled words.

Different from unsupervised approaches exploring distance and dependency rules, and semantic lexicons, some works have explored supervised approaches based on machine learning. Traditional works have made use of Conditional Random Fields (CRF). However, works applying deep learning neural networks for the tasks have rapidly reached state-of-the-art results. Below we give some examples and discuss those supervised approaches.

Irsoy and Cardie (2014) embraces the problem of Opinion Expression Extraction, modeling the problem as a token-level sequence-labeling task. The authors apply deep RNNs to solve the problem and compare the obtained results against shallow RNN models and the traditional CRF. The study shows the superior performance of the proposed model. Liu, Joty and Meng (2015) also employs RNNs to solve the problem by aligning the architecture with pre-trained word embeddings.

Wang et al. (2016) presents RNCRF, a joint model that integrates a Dependency-Tree Recursive Neural Network (DT-RNN) and a CRF for explicit aspect and opinion terms co-extraction. The authors add the DT-RNN architecture to extract word-level representations considering syntactic relations and semantic robustness by running a recurrent neural network over constituency and dependency trees. The CRF layer serves as a learner of the context around each word by receiving the DT-RNN learned representation as input. The absence of labeled training labels, mainly for opinion terms, required the authors to label these terms manually.

Wang et al. (2017) presents CMLA, a model based on multiple layers of attention networks, to extract aspects and opinion terms. For each sentence, the authors use a pair of attention layers to learn a prototype vector for aspect or opinion terms, a high-level feature vector for each token, as well as an attention score for each token. The authors also use the attentions to model direct and indirect relations between aspect and opinion terms.

The authors argue that while a single-layer architecture can capture direct relations, a multi-layer architecture is required to model the indirect relations.

Li and Lam (2017) introduces the Memory Interaction Network (MIN), a multi-task learning framework that jointly handles the extraction of aspect terms and opinions. The model also performs sentimental sentence classification. The framework is composed of one Long Short-Term Memory (LSTM) layer for each task. The LSTM layers are extended with memories to store interactions between the tasks. A single loss function combines the separate loss functions of each of the three LSTMs that compose the model. The LSTMs performing the extraction tasks compute a token-level cross-entropy error, while the LSTM for sentiment classification computes the sentence-level cross-entropy error. The training objective is the addition of these three loss functions.

Similar to Li and Lam (2017), He et al. (2019) introduces Interactive Multi-task learning Network (IMN), a multi-learning network that jointly learns aspect and opinion relations for co-extraction. Additionally, it performs aspect-level and document-level sentiment classification and the document-level domain classification. The model applies a message passing mechanism that propagates information from the different tasks to update the sequence representations. The authors model the extraction tasks as a sequence labeling problem, but instead of Recurrent Neural Network (RNN), layers of Convolutional Neural Network (CNN) are employed as encoders. For the classification tasks, the authors employ a self-attention layer on top of the stacked CNNs.

Xu et al. (2018) present a CNN-based architecture with two pre-trained embedding layers as input to perform the task of Aspect Term Extraction (ATE). Li et al. (2018) also try to solve the Aspect Extraction task, but its approach involves exploring the aspect detection history and opinion information. For this, the authors build a model based on LSTM layers and Attention mechanisms. Although the authors consider the association between aspect and opinion important, they focus on modeling this to improve ATE only. Both works compare their results for the ATE task against Wang et al. (2016), Wang et al. (2017), and Li and Lam (2017) joint methods, which present inferior results. Li et al. (2018) assert that these joint extraction methods sacrifice the accuracy of the aspect prediction when trying to perform Opinion Term Extraction (OTE). Additionally, it states that those joint methods do not care about the correspondences between both performed tasks.

3.1.2 TOWE

Before Fan et al. (2019) introduce the problem of Target-oriented Opinion Words Extraction (TOWE), for finding opinion words related to some aspect (or opinion target) in the review, this problem was usually seen as two separate tasks: Opinion Target Extraction (or ATE) and Opinion Words Extraction (or OTE). Some previous works, motivated by the success of neural networks, have employed multi-task architecture to perform the two tasks jointly. However, Fan et al. (2019) point out that these works do not extract aspects and opinions as pairs, which could be significant for the Aspect-based Analysis of reviews. Given the challenge of finding labeled datasets for the tasks, the authors manually build four datasets for the tasks based on the SemEval challenges (PONTIKI et al., 2014), which are largely used for other ABSA tasks but do not contain token-level labels for opinion targets and words.

The authors use the BIO scheme to label the data and the problem is then addressed as a sequence labeling problem. Then, they develop a model based on the Encoder-Decoder architecture. The Encoder consists of a target-fused approach that incorporates the left and right contexts of the target into a context representation. To model the left and right context, the authors apply an Inward-Outward LSTM and combine its outputs to a global context. The Decoder receives the Encoder output and performs the sequence labeling task. The authors highlight that the proposed Decoder can adopt two policies: A Greedy, and the other based on CRF (JOHN; ANDREW; FERNANDO, 2001). However, for experimentation, the authors consider only the Greedy one. The Greedy decoding is formulated as a three-class classification problem for each position in the output. It applies the Softmax function to compute the probability for each position given the sequential representation built by the Encoder. The authors apply the Negative Log-Likelihood (NLL) as loss function. The proposed model is called IO-LSTM + Global Context (IOG).

IOG is benchmarked against distance-rule models (HU; LIU, 2004), dependency-rule models (ZHUANG et al., 2006), and pipelines with neural architectures, using word embeddings, traditional LSTM/BiLSTM models, and a hybrid variation of neural models and distance-rules. The obtained results show that the proposed approach performs better than the other analyzed approaches, in which the distance-rule model performs worst alongside the dependency-rule model. Also, the authors find that the hybrid model composed of a pipelined neural model and distance-rule performs better than these isolated

architectures but still worst than the proposed IOG model, approximately 10% lower in F1-score.

Following the work of Fan, other recent works have tried to improve the classification results for the task by changing the approach using new architectures, new training objectives, transferring knowledge from external corpora, and even by proposing new labeling models. Below we describe these approaches.

Wu et al. (2020b) aims at transferring knowledge from external review corpus to improve the Target-Oriented Opinion Word Extraction (TOWE) task. For that, the authors propose the Latent Opinions Transfer Networks (LOTN) model, which consists of two components: The first is a simple position and word embedding-based BiLSTM network, called PE-BiLSTM, that performs the actual TOWE task; and the second component is a pre-trained sentiment classification model responsible for retrieving global and target-dependent word-level representations from the input review sentence. The proposed model works as follows: the review sentence is sent to the pre-trained sentence classification model, which outputs hidden states and attention weights relative to the sentiment classification task. This information is then concatenated to the hidden states of the PE-BiLSTM network. Hence, the representation of the TOWE module will contain task-specific context representation and external opinion knowledge.

To transform the global and target-independent opinion information coming from the sentiment classification module into target-dependent information, the authors follow the assumption that "the word that is closer to the opinion target is more likely to be the opinion word of the target". This premise is computed by a target-relevant distance weight function that considers the sentence size and the relative position of the words in the sentence to the target word. The authors perform experiments over the built datasets from TOWE, using the Amazon Review and Yelp Review corpus to train the sentiment classification model applied for latent opinion transference. Performed experiments show that the proposed method outperforms the state-of-the-art model IOG by 1.98% and 2.02% F1-score for the restaurant datasets from the SemEval 2014 and 2015.

Zhou et al. (2020) argue that most previous methods for the task have relied on the sequential representation of the sequence, ignoring the dependency structure between the target and opinion words. Hence, the authors propose a neural network architecture based on Graph Convolutional Networks (GCN) which captures the syntactic structure of the sentence and the syntactic relations between the terms. According to the authors, this

approach circumvents the problem of capturing dependencies between the sequence of words when the opinion is far from the opinion target. The authors also increment the training of the proposed model with adversarial training, by adding small perturbations to the input word embeddings, which can enhance the generalization and robustness of the model.

The model is composed of a BiLSTM encoder, which learns the contextual information of the words in the sentence. The GCN is applied over the dependency tree to compute the syntactic representation of the sentence. Both representations are integrated to predict the label of each word. The adversarial examples are created during training. These samples are added as noises to the model, hence they are built by adding worst-case perturbations into the original word embeddings, i.e. the perturbation that maximizes the loss function. This training step considers two loss functions which refer to the cross-entropy loss on the original samples and on the adversarial ones. Therefore, the training objective considers both loss functions during training. The results of the performed experiments show superior performance when compared to the traditional IOG model for the TOWE task and other basic architectures using only BiLSTM and not considering syntactic features for sequence classification. The authors also benchmark the proposed model with distance-rule and dependency-rule models which are already proven to perform worse for the task.

Zhang et al. (2021a) addresses the TOWE problem similarly to Zhou et al. (2020). The authors propose the use of GCN for capturing syntactic features between aspects and opinion words. However, to leverage the challenge of using GCNs the authors integrate a memory mechanism that updates the hidden states of each node with historical, local feature, and contextual information. The model consists of word and positional embeddings given as inputs to a BiLSTM encoder. This BiLSTM encoder process the sequence and integrates its representations with the GCN and memory cells' hidden states. The graph for the syntactic relations is split into multiple subgraphs, where each node is assigned with a memory cell. In a recurrent manner, each node is updated to build the final node representation. The cross-entropy function is used as a training criterion and a custom loss function is applied. The authors benchmark the proposed model with traditional distance-rule, dependency-rule, sequential, and pipelined models. State-of-the-art models as IOG and LOTN are also compared to the proposed work. The experiment results show that the proposed model outperforms the other analyzed works.

Zhang et al. (2021b) approach the TOWE task as a question-answering problem. Hence, they build a multiview-trained machine reading comprehension model, that consists in training a Machine Reading Comprehension (MRC) model and split the problem into three separate views: identifying opinions oriented to a given target (TOWE), Opinion-Related Aspect Targets Extraction (OATE), and performing Target-Opinion Pair Relation Classification (PRC). The authors use three question templates to automatically build questions that will help in the model training. Then, introduce the MultiView Training (MVT) strategy that captures the common knowledge obtained from those different views. To learn the contextualized representations for each token the authors use Bidirectional Encoder Representation from Transformer (BERT), which is used as the MRC model. The TOWE and OATE views receive as input the last hidden states from the BERT transformer and pass it through a softmax function, using cross-entropy as the training criterion. The PRC view receives only the last hidden states that correspond to the [CLS] token, which is also sent through a softmax function and the cross-entropy loss function. For MVT, the authors introduce a meta-learning approach, whose goal is to learn parameter initializations that could fastly adapt to all three tasks with only a few training data. The framework ends by initializing all training views with the learned parameters from the meta-learning approach, then finetunes it to the final TOWE task.

3.1.3 AOPE or PAOTE

Different from TOWE, Chen et al. (2020) introduce the AOPE which aims to explore the relationship between aspect targets and opinion terms. The authors utilize the BERT model for learning context representations for tokens and send these obtained representations through the proposed Synchronous Double-channel Recurrent Network (SDRN), which is modeled to solve the task. The model consists of an opinion entity extraction unit, a relation detection unit, and a synchronization unit. The two first units are responsible for extracting aspects, opinion expressions, and the relations between them. The latter unit is responsible for allowing the interaction between the other two units so that the extraction of both elements can be simultaneous. This unit is composed of two other submodules: The Entity Synchronization Mechanism and the Relation Synchronization Mechanism. The former captures each token's corresponding entity semantics and the latter captures the semantics of the relations between aspect and opinion. Both submodules

perform updates to the hidden representation sequence.

The recurrency is included in the model to capture high-level representations. At each recurrent step, the Opinion Entity Extraction unit builds hidden representations for the sequence based on the input token representations, coming from BERT encoder, and from the relation synchronization semantics obtained from the Synchronization unit. Simultaneously, the relation detection unit explores the relations between aspects and opinion expressions using a supervised self-attention mechanism. This unit computes the degree of correlation between tokens using a score function that receives the hidden representation sequence as a parameter. To compute this hidden representation, it uses the context sequence from the BERT encoder and the entity synchronization semantics from the synchronization unit.

The authors evaluate the model by comparing it to pipelined and joint methods. The pipelined methods consist of five advanced extraction methods applied for opinion entities recognition: HAST, DE-CNN, IMN, SPAN, and RINANTE. The output of these models is fed to the SDRN Relation Detection Unit to obtain aspect-opinion pairs. The joint models are the ones that perform joint extraction of aspect and opinion terms: IDF, CRF+ILP, and LSTM+SLL+RLL. However, the two firsts are based on shallow machine learning methods and hand-crafted features, and the last neglects the interaction between opinion entities and relations. The proposed model SDRN shows superior performance among all analyzed models and datasets. The authors find that the joint learning of aspects and opinions avoids the error propagation present in the pipelined models. Also, they find out that BERT captures rich context representations. Additionally, the co-extraction models performance demonstrates that jointly detecting aspects and opinions can benefit each other through relation detection.

The same task introduced by Chen et al. (2020) is presented by Zhao et al. (2020), under the name of Pair-Wise Aspect and Opinion Terms Extraction (PAOTE). However, unlike previous work, the authors approach the problem as a multi-task learning framework based on shared spans rather than sequence tagging. The authors argue that "sequence tagging methods suffer from a huge search space due to the compositionality of labels". They also state that these methods tend to present poor performance due to one-to-many or many-to-one relations between aspects and opinion terms, for example in the review sentence *"...this place has great* ***service*** *and* ***prices****, and a nice friendly* ***atmosphere****."* the aspects services and prices present a many-to-one relation with the

opinion term great while the aspect atmosphere and opinion terms nice friendly presents a one-to-one relation. The proposed method first learns word-level representations using a base encoder, enumerates the possible spans for aspect and opinion terms, builds representations for these spans, and finishes identifying span-span relations for label assigning. The model comprises the base encoder, the span generator, and the multi-task learning objective function. The authors propose two architectures for the base encoder: a BiLSTM-based and a BERT-based, which are responsible for learning word-level representations. The span generator builds all candidate spans for aspect and opinion terms in the review sentence and applies the base encoder to them, retrieving contextualized representations for these spans.

The objective function of the model is composed of two scores: the term scorer and the relation scorer. The first computes the probability of the span is a term label and applies a proposed span-level cross-entropy loss function; the latter computes the likelihood of a span being part of a relationship and uses a proposed pair-level cross-entropy loss function. Both scorers receive as input the span representation built previously. Finally, the objective function combines the error computed by both scorers through summation, weighing each one with a separate hyper-parameter. The authors evaluate the proposed model on Aspect Terms extraction, Opinion Terms extraction, and the joint aspect and opinion terms extraction in pairs. The work is benchmarked against sequence tagging models using different encoder structures and state-of-the-art co-extraction models. The authors find out that the performance of sequence tagging methods is not satisfactory for this problem ad that BERT-based models perform worst. Meanwhile, the performance of co-extraction models is much better. However, these co-extraction models still fail in associate aspect and opinion as pairs. Hence, the proposed method Span-Based Multi-Task Framework (SpanMlt) can better model the interactions between aspects and opinions and present better performance for Aspect Term extraction, Opinion Term extraction, and the joint extraction of aspects and opinions as pairs.

Wu et al. (2020a) propose addressing the task, which they refer to as Aspect-Oriented Fine-Grained Opinion Extraction (AFOE), by changing the tagging mechanism of training examples. The authors argue that works in Opinion Pairs Extraction can be reduced to pipeline variations which can easily suffer from error propagation and inconvenience in real-world scenarios. Hence, they propose the Grid Tagging Scheme (GTS) that aims to approach the problem as a unified grid tagging task and decrease the error propagation

problem of pipelined approaches. A proposed decoding method, applied during inference time, allows the tagging of all word-pair relations and opinion pairs at the same time.

The proposed grid tagging consists in assigning one of four tags to each token in the sentence. The labels are: “A” representing a word belonging to the same aspect term; “O” for the words belonging to a same Opinion Term; “P” to words belonging to both Aspect and Opinion terms; and “N” where no relation exists. For the Opinion Triplet Extraction (OTE), where the third term is sentiment polarity, the tag “P” is replaced by one of the tags in the set POS, NEG, NEU representing the aspect sentiment. The tagging scheme is displayed as a grid, where sentence tokens are distributed in rows and columns. However, for simplicity, the authors adopt an upper triangular grid. A decoding algorithm is proposed to form the pairs or the triples, according to the task, at inference time. For the Opinion Pair Extraction (OPE), the algorithm first looks for aspect and opinion terms, then verifies if the relation between them exists, considering at least one word pair from these terms is labeled with the tag “P”. The same is performed for the OTE task, but instead of “P”, it looks for the most predicted sentiment tag. The authors benchmark the proposed approach with OPE baselines and variation of the GTS approach with different encoding mechanisms. The results have shown that the GTS with BERT encoder performs better than the other methods.

Feng et al. (2021b) argue that the TOWE task can be applied to solve the AOPE task. Hence, the authors present two models: the Target-Specified Sequence Labeling with Multi-head Self-Attention (TSMSA) model, which is applied to the TOWE task; and the variation MT-TSMSA, with MT standing for multi-task, to perform AOPE task. The TSMSA labels the target entities in a review with the [SEP] marker and retrieves their representations context using the multi-head self-attention mechanism. A combination of projection and CRF layers receives the built representations and outputs a labeled sequence assigning each token in the review to an opinion, target word, or neither. The authors integrate aspect and opinion extraction and TOWE into a multi-task architecture. The encoder (multi-head self-attention) is the same for both models, but the projection and CRF layers are different.

The authors benchmark both proposed models with baselines for both tasks: TOWE and AOPE. The baselines include distance-rule, dependency-rule, neural models integrated with distance rule, the BERT model integrated with distance eule methods and a target-fused model. The authos also benchmark their work with state-of-the-art mod-

els as HAST, IOG (FAN et al., 2019), Joint Entity Recognition and Relation Extraction as a Multi-Head Selection (JERE-MHS) (BEKOULIS et al., 2018), SpanMlt (ZHAO et al., 2020), and SDRN (CHEN et al., 2020). The results show that rule-based methods present poor performance for TOWE, but it improves when integrated with the BERT model. The authors also find out that the performance of the neural models and the target-fused BERT model is 10% lower than IOG, SDRN, and the proposed model TSMSA. The performance of TSMSA and IOG is similar when training the first one using static word embeddings. The difference appears when training TSMSA with the pre-trained language model BERT, which also presents superior performance than SDRN. For the AOPE task, the MT-TSMSA shows competitive performance compared to SDRN. Both models outperform the other analyzed models for this task: History Attention and Selective Transformation (HAST) (LI et al., 2018) + IOG, JERE-MHS, and SpanMlt. An ablation study shows that Glove embeddings perform better for TSMSA than BERT embeddings without fine-tuning and that fine-tuned BERT performs better than both.

Gao et al. (2021) explore the AOPE task by proposing a question-driven span labeling model. The authors split the AOPE task into two subtasks: ATE and Aspect-Specified Opinion Extraction (ASOE). Here, the problem is approached as in the TOWE task, where the model first extracts all candidate aspect terms, and given the aspect, it tries to extract the corresponding opinion words. Initially, the model extracts all candidate aspects from the review sentence, then it automatically builds auxiliary questions related to each extracted aspect. To solve the ASOE task, the built questions are concatenated to the respective review sentence as sentence pairs. The task is solved as a machine-reading comprehension problem rather than as a sequence labeling problem as performed by previous works. The model is composed of an Aspect Terms Span Extractor (ATSE) and an Opinion Words Span Extractor (OWSE). The base encoder of the model is BERT. The model uses BERT as the sentence encoder to build context-related features for the sentence. BERT is also applied as a joint encoder to build features for the pair of sentences composed of the original review sentence and the automatically build auxiliary question. The ATSE develops the span-based scheme instead of the traditional BIO scheme. The objective of this module is to detect boundaries of aspect terms using two binary classifiers. ATSE detects the span (start and end positions) of each aspect item.

Similar to what Zhou et al. (2020) and Zhang et al. (2021a) propose to TOWE, Wu et al. (2021) present the use of an edge-enhanced syntactic GCN as the encoder

of syntactic features of review sentences to extract pairs of aspect-opinion terms, the AOPE task. Different from traditional GCN that models only the syntactic dependencies, the network variation here proposed models simultaneously the dependency arcs and its labels. Performed experiments suggest that the inclusion of this additional information builds better features for the studied task. The extraction pipeline is integrated with span detection, filtering, and representation, followed by a high-order pairing layer which outputs the predicted pairs. The span detection consists in applying a softmax operation over the built candidate spans, classifying them as Aspect, Opinion, or Invalid. The spans marked as invalid are then filtered out and the others are sent to the pairing layer. The pairing layer aims at deciding if a pair of terms are valid aspect-opinion terms or not and consists of two scorers, a biaffine and a triaffine. The biaffine scorer receives two terms and decides if the relation between them exists. The triaffine scorer receives combinations of three candidate terms and aims at detecting overlapping relations in the sentence. In the end, the punctuation of both scorers are combined and a unique measure is assigned to the pairs. A sigmoid layer is applied over this punctuation and a threshold is used to output valid pairs.

3.2 STRUCTURING SUBJECTIVE INFORMATION AND PRODUCT KNOWLEDGE GRAPHS

The term Product Knowledge Graph (PKG), also called Product Graph (PG), is first coined by Dong (2018) when presenting the effort of Amazon in building an authoritative knowledge graph for all products in the world. The approach investigates building the PG by exploring semi-structured web sources and using the distantly supervised tool, CERES (LOCKARD et al., 2018), building noisy training labels automatically. Sequence-based classification models are also studied to extract attribute values from these semi-structured sources. They also investigated techniques for knowledge integration, cleaning, graph mining to decide the importance of entities and relations in the graph, and even human-in-the-loop techniques.

More recently, Li et al. (2020) from Alibaba introduced AliMe KG, a domain knowledge graph in e-commerce that captures user problems, Points of Interest (POI), item information, and relations. Unlike Dong (2018), the authors aim to capture users' interest from chatbots conversations and link them to product' items in the Knowledge Graph (KG). The authors perform the KG construction by first mining POI, to extract

potential user interests, needs, and problems. The approach consists of setting heuristics to build training examples automatically and then applying a BERT-based binary classifier for POI mining. The authors use another mining approach based on a Named Entity Recognition (NER) model to extract structured information from products and their hierarchical structure. The authors focus on property values of items (IPV) and category-property-value (CPV) information, the latter consists of the hierarchical categories structure containing properties and items. The NER model is a combination of BERT embedder, Bidirectional Long Short-Term Memory (BiLSTM) encoder, and CRF decoder. Finally, the authors employ a POI relational Knowledge Mining to relate POI with user needs to the product items, which supports the explanation of recommendation tasks. The authors apply a BERT-based model to perform the Relation Extraction.

Xu et al. (2020) also explore PG by learning embeddings from raw customer activity data and product descriptions. The authors model relations as ISA, substitute, complement, co-view, describe and search to satisfy e-commerce applications. Learning these intertype of connections between user interactions and product descriptions can allow tasks such as knowledge completion, search ranking, and recommendation. Hence, the authors build a PG for some of these relations to validate the proposed approach. For example, to construct the PG, the authors build a weighted graph with edges representing the number of sessions two products have been co-viewed, co-purchased, or substituted.

OpineDB (LI et al., 2019) is a subjective database system that stores opinions extracted from user reviews. These opinions are structured according to a subjective database schema, and allows subjective queries to be processed over these opinions. However, this system depends on a predefined schema according to the focusing domain and it aims at specifically build queries to answer subjective questions on the stored information.

Kobren et al. (2019) proposed a method for constructing a knowledge base of entities and their attributes that allows tunable precision, that is, the Knowledge Base (KB) can be set to run with a particular false positive rate, even when it stores subjective attributes. Firstly, the system sends questions for users about an entity in the KB and receives "yes" votes and "no" votes in response. It uses the votes to bootstrap the training of a probabilistic "yes" rate model for each entity-attribute pair. Uncertainty in each model is explicitly represented via a distinct prior distribution. When one queries the KB, entity-attribute pairs are only included in the response if the KB is sufficiently confident that their corresponding "yes" rate exceeds a given threshold. It uses three neural networks

for estimating the “yes” rate of each entry.

Melo et al. (2019) introduced OpinionLink to organize opinions around product attributes as defined in a product catalog. The authors argue that the product attributes are the most important characteristics of the products. Their approach is divided into two phases. In the first phase, OpinionLink uses a supervised classifier to identify opinionated sentences in the reviews on a particular product. In the second phase, they use another supervised classifier to map opinions previously extracted from user reviews to the attributes of the products in the product catalog.

3.3 LANGUAGE MODELS

In this section, we overview works exploring Language Model (LM) as Knowledge Bases to extract information from their tunned parameters and solve different tasks. We also highlight works applying In-Context Learning, focusing mainly on Data Augmentation tasks. By the end of the Section, we discuss work focusing exclusively on using Language Models for Data Augmentation of ABSA tasks.

3.3.1 Language Models as Knowledge Bases

Given the growing interest and the variety of proposed LM in the literature in recent years, some works have emerged trying to explore the knowledge stored on their large number of parameters. Also, given that these models are unsupervised, these emerging works have investigated using them as sources of supervision to answer questions, perform data labeling, and even build knowledge graphs. Below we list some of these works.

Petroni et al. (2019) point out some advantages of using Language Models instead of structured knowledge bases. These advantages are: LM do not require schema engineering; their structure allows querying open class of relations; They are easily extendable and do not require human supervision during training. Beyond learning linguistic patterns, the authors state that Language Models can also store relational knowledge and answer structured queries in the format “fill-in-the-blank”. Without fine-tuning, the authors present an analysis of the knowledge contained in BERT to validate the hypothesis previously stated.

The authors introduce the LAnguage Model Analysis (LAMA) that test and evaluate

the knowledge contained in pre-trained language models. The analysis explores existing knowledge sources to build golden standard triples. The authors use these triples to query the Language Models and validate the information retrieved. For this, the authors mask the object element of the triple in the sentence. The LM receives this masked sentence and outputs the probable tokens that match the blank space. As an example, from the triple (Dante, *born-in*, Florence), the authors try to predict the masked object "Florence" by feeding the sentence "*Dante was born in* _____." as input to the model. The model evaluation considers how highly the model ranks the expected token against other words in the vocabulary. According to their studies, the authors find out that BERT can recall the knowledge stored in its weights, showing competitive performance with non-neural and supervised approaches. Hence, there is potential for applying pre-trained language models to work on unsupervised open-domain question-answering systems.

Similarly to Petroni et al. (2019), the work of Roberts, Raffel and Shazeer (2020) evaluates the capacity of LM on the open domain question answering task. However, unlike the former, the authors fine-tune the LM to the question-answering task. Then, the authors parse Natural Language Queries as input to the model that must be answered just by accessing the knowledge stored in its parameters. The authors use three different open-domain natural questions datasets and base the fine-tuning of the model on the work of Raffel et al. (2019) (i.e., T5). For experimentation, the authors employ a 90/10 split rate on datasets to perform a hold-out validation. At the evaluation phase, the authors choose the most likely token obtained from the models' predictions as output. Like Petroni et al. (2019), Roberts, Raffel and Shazeer (2020) show that large language models can indeed show competitive performance on question answering problems without accessing additional or external information.

Similar to Petroni et al. (2019), Wang, Liu and Song (2020) proposes Match and Map (MAMA), an unsupervised end-to-end approach that intends to build Knowledge Graphs using Language Models without fine-tuning. The proposed approach retrieves facts from the pre-trained language model, passing the LM over a text corpus. As the name suggests, the method consists of two main steps: match and map. The first step generates candidate facts in a triple format (head, *relation*, tail). In this step, the authors perform a beam search over the attention weights matrix retrieved from the LM passage over the input sentence. The head and tail tokens are identified as noun chunks of the sentence. The head-tail pair is given as input to the matching step, along with the attention weights and the sentence

itself. The beam search and other heuristics try to retrieve the relation between head and tail by looking for the largest score from the attention matrix. The mapping step involves performing entity linking and relation mapping between the retrieved candidate triple and the KG schema. The authors also find out that large Language Models can store richer knowledge and be further applied for the continuous improvement of knowledge graphs.

Language Models are also used by Feng et al. (2021a) as annotators for Dialog Summarization. The authors present DialoGPT, a pre-trained model based on GPT2 (RADFORD et al., 2019) that generates conversational responses in an unsupervised manner. The proposed model can also perform keywords extraction, redundancy detection, and topic segmentation.

Beyond these approaches for transferring knowledge from Language Models, the works of Clark et al. (2019) and Xu et al. (2020) investigate the learned representations and attention weights of pre-trained BERT. These works try to find patterns existent in the attention maps from the Transformers' heads. Clark et al. (2019) presents a series of analysis methods for understanding the attention heads at the word level. The study finds out, for example, that some heads can attend to direct objects of verbs, determiners of nouns, objects of prepositions, and coreference mentions. Hence, they demonstrate that BERT's attentions also capture syntactic information.

Xu et al. (2020) focuses its analysis on review knowledge from ABSA tasks. The authors try to understand the inner workings of the Masked Language Model of BERT, looking at token-level features and their connections with ABSA tasks, like end-to-end ABSA, Aspect Extraction, and Sentiment Classification. The authors pre-train the BERT model on a review corpus composed of Amazon and Yelp reviews on Laptops and Restaurants. For validating the analysis, the authors sample 150 labeled examples from each domain (Laptops and Restaurants) from the SemEval 2014 Task 4 (PONTIKI et al., 2014) and SemEval 2016 Task 5 (PONTIKI et al., 2016) datasets. The authors explore the inner workings of the Masked Language Modeling (MLM). The study's main finding is that MLM tends to learn very fine-grained features from aspects and that these representations are mostly related to the domain's semantics than to opinions. Hence, the authors point out that the pre-trained BERT is good for tasks such as Aspect Extraction (AE) and the extraction part of End-to-End ABSA. Meanwhile, tasks as Aspect Sentiment Classification (ASC) and Aspect Summarization are not well fitted by the BERT model. This analysis finds, for example, some general patterns as no-op relations on [CLS] and

[SEP] markers, offsets on previous/next tokens, broadcast over whole sentences, and context words of aspect or opinions. The authors point out future directions as using these learned representations for self-supervised learning on ABSA tasks by predicting masked aspect words.

These State-of-the-Art (SOTA) works, exploring Language Models as Knowledge Bases and trying to explain their predictions and weights or how relations are retrieved from learned representations, denote that Language Models can be explored as supervision sources. Different from these works we do not aim to analyze the learned weights or representations of these Language Models. Instead, we explore In-context Learning without prompting engineering. Given these provisions, we aim to use LM and In-Context Learning without relying on prompting engineering to generate opinionated triples to enhance the newly introduced ABSA tasks: TOWE and AOPE.

3.3.2 In-Context Learning and Data Augmentation

In-Context Learning as first introduced by (BROWN et al., 2020) consists of providing the Language Models (LM) with some samples or instructions through prompting to solve some task. As categorized by them, In-context learning is composed of three learning paradigms: zero-shot, one-shot and few-shot. The first consists of providing the model with any demonstration of the task, just an instruction. The second and the third differ in the number of task observations provided to the model.

Recently, a few works (LIU et al., 2021; WANG et al., 2021; WANG et al., 2021) have explored LM *as-is* or performing prompt engineering which consists of finding the most appropriate prompt to solve some given task. (LIU et al., 2021) surveys these methods for Natural Language Processing tasks. The paradigm has already been used for Unsupervised Data Generation (UDG)(WANG et al., 2021) and for open information extraction, relation classification, and factual probe (WANG et al., 2021).

UDG, proposed by Wang et al. (2021), consists of training the LM with a zero-label procedure to enable the LM for few-shot examples generation. In other words, the authors build a prompt with some labeled examples and a description of the task. Hence, the text generated by the LM is expected to represent the same labels assigned to the prompt examples. The generated text aligned to the label in the prompt examples is used to finetune any other model to the required task. The authors evaluate the approach to text

classification and language understanding tasks.

Although , Wang et al. (2021) deal with Open Information Extraction, it deals with the problem as a translation task where the zero-shot approach is applied giving as input a NP-chunked text and the generation must be on the triple format. For this, the authors rely on the pretraining of BERT and on a ranking step based on it. Since the authors work with BERT, the generation step is performed using beam-search instead of In-Context Learning prompting.

Other works have explored the finetuning of pre-trained LM to generate training examples for augmentation tasks (TAVOR et al., 2020; ABONIZIO; JUNIOR, 2020; BAYER et al., 2022; LIU et al., 2020; YOO et al., 2021). Tavor et al. (2020) presents the Language-Model-BAsed Data Augmentation (LAMBADA) which consists of finetuning GPT-2 to a text classification task and conditioning the generation to the expected classes. Abonizio and Junior (2020) propose PRE-trained Data AugmenTOR (PREDATOR), an improved version of LAMBADA but using DistilGPT2 (SANH et al., 2019) and applying the concept of few-shot prompt. Bayer et al. (2022) incorporates the finetuning of GPT-2 with prompting and filtering method based on document embeddings to augment tasks using long texts. Liu et al. (2020) aligns GPT-2 with reinforcement learning to predict the tokens of an instance to be generated giving as input the instance class. Yoo et al. (2021) is among the first authors to use GPT-3 for data augmentation, it uses already known labeled training instances to build prompts and condition the model to generate similar samples.

MetaICL (MIN et al., 2022) applies meta-training, i.e., it makes use of a large set of different tasks to tune a pre-trained language model. The authors show that this tuned pre-trained model learns how to do In-Context Learning on unseen tasks. The experiments run on a large collection of 142 datasets including tasks such as text classification, question answering, natural language inference, and paraphrasing. The approach is based on Few-shot learning, where the LM is meta-trained on a given task with *k+1* samples of this task emulating the in-context learning approach. At inference time an unseen task is given as input to the model along with k training instances, a test example, and a set of candidate labels. The model must predict the label with the maximum conditional probability.

Our work differs in some aspects from the ones mentioned previously: i) the tasks embraced by these works are usually text classification and mostly applicable to short texts and instances while in our work we comprise a more complex problem, the generation of

opinionated triples; ii) We perform Task-Adaptation Pretraining (TAPT) instead of fine-tuning, which is applied for supervised tasks; iii) we do not perform prompt engineering, we get the expected generation format with Task-Adaptative Pretraining (TAPT); iv) some approaches train a classifier to filter the generated samples, we only apply a post-processing step.

3.3.3 Data Augmentation in ABSA

Recent works have already explored LMs for Data augmentation, but in other tasks related to ABSA (DING et al., 2020; LI et al., 2020; HSU et al., 2021; LI; YU; XIA, 2022).

Ding et al. (2020) unifies the process of sentence generation and labeling using a LM. Dealing with sequence tagging problems, such as named Entity Recognition (NER), Part of Speech (POS) and End-to-End Target-based Sentiment Analysis (E2E-TBSA), the authors apply a sentence linearization. The sentence linearization consists of pairing up each word label right before the word which results in one single sequence. With these linearized sequences at hand the authors train a Language Model based on one-layer recurrent neural network. The resulting generative LM can then be used to generate fine-grained synthectic data from scratch.

Li et al. (2020) applies the MAsked Sequence-to-Sequence method of MASS to conditionally build new sentences while preserving the original aspects and labels. The authors apply their approach to solve the Aspect Term Extraction (ATE) task.

Hsu et al. (2021) proposes a method, called Selective Perturbed Masking (SPM), to measure the importance of each word in a textual sentence. With the masking approach of BERT the authors mask the unimportant words and replace them by words generated by the LM. The method is applied to augment the Aspect Category Sentiment Classification (ACSC), Aspect Term Sentiment Classification (ATSC), Aspect Term Extraction (ATC), and Sentiment Classification (SC) tasks.

Li, Yu and Xia (2022) propose a Generative Cross-Domain Data Augmentation Framework using BART (LEWIS et al., 2020) to enhance the task of aspect and opinion co-extraction. In other words, they focus on transferring knowledge from one source domain to augment the performance in another.

To the best of our knowledge, no work has yet investigated the automatic generation of complete opinionated triples to augment TOWE (FAN et al., 2019) or AOPE. The work

closest to ours is the one from Ding et al. (2020). However, their work is applied to End-to-End ABSA, and our pretraining and generation strategies do not rely on sequence tagging. Also, we use a pretrained large LM instead of building one from scratch.

3.4 COMPARATIVE ANALYSIS

In this section, we present a comparative analysis between state-of-the-art works (presented in the previous chapter) and our proposed method **P**roduct **G**raph enriched with **Opi**nions (PGOpi). Here we highlight the similarities and differences between them and how we contribute to the state-of-art with the proposed pipeline. Also, we end this section by discussing the similarities and differences between our **SYN**theti**C** **Op**inion**A**ted **T**ripl**E** (SYNCOPATE) generator approach and other works trying to improve the ABSA study field with data augmentation.

3.4.1 Entire Pipeline

Table 1 – Comparison between state-of-the-art work on PG enhancement with opinion information (OpinionLink) and our proposed pipeline.

	PGOpi	**OpinionLink (MELO et al., 2019)**
Approach	Weak-supervised	Supervised
Opinion Extraction	Unsupervised rule-based approach: dependency trees and common-sense knowledge	Supervised direct opinionated sentences classifier (SVM) + rule-based approach: dependency trees and common-sense knowledge
Labeling of training examples	Distant Supervision	Manual
Opinion Mapping	DNN	SVM with cross-validation

Source: Created by the author

OpinionLink (MELO et al., 2019) is directly related to our work. Similar to ours, it uses as input a set of reviews on specific products, and groups the opinions extracted from these reviews around the targets of this product. However, in OpinionLink, the targets come from a product catalog. Hence, the authors assume that all products from the same category in the catalog have the same attributes. Unlike them, our method is

more flexible since it allows products of the same category to present different attributes. Table 1 highlights the differences between both models.

Furthermore, OpinionLink requires manual labeling of training data for achieving accurate predictions. **P**roduct **G**raph enriched with **Opi**nions (PGOpi) employs a semi-supervised approach, mitigating the problem of manually labeling training datasets. Additionally, the automatic building of training examples allows applying state-of-the-art models, like Deep Neural Networks that require a large amount of training data, instead of employing traditional models, like Support Vector Machine (SVM), using just a few manually labeled examples.

Given these reasons, although OpinionLink effectively organizes opinions around product attributes, it can hardly scale or adapt to flexible scenarios. Since PG can show flexible schemas, it requires too much effort to enrich it depending on manually labeled training data. Since OpinionLink is closer to ours, we decided to adapt it to use as the baseline in our experiments.

3.4.2 Organizing Subjective Information

Table 2 – State-of-the-art works organizing subjective information and/or building/enhancing Product Knowledge Graphs.

		INFORMATION TYPE			
	DATA STRUCT.	Semi-structured/ Structured information	Unstructured information		
			Opinions	chatbot conversations	activity data
Dong (2018)	KG	X			
Li et al. (2019)	Rel		X		
Kobren et al. (2019)	KG				
Melo et al. (2019)	KG		X		
Li et al. (2020)	KG			X	
Xu et al. (2020)	KG				X
PGOpi	KG		X		

Source: Created by the author

In the previous subsection, we discuss OpinionLink (MELO et al., 2019), the work directly related to **P**roduct **G**raph enriched with **Opi**nions (PGOpi). Here, we discuss other works that explore subjective information from the Web but are not necessarily

directly related to our pipeline. Table 2 highlights some differences and below we give details on these differences between these approaches to ours.

Dong (2018) presents a product knowledge graph to structure factual information from semi-structured sources on the Web. Also, different from our work, Xu et al. (2020) models relations between customer activity and the product descriptions, trying to identify related products through the users' navigation. Unlike them, our work explores the addition of subjective information (unstructured), like opinions, to existing PG. Hence, these works are not directly related to ours.

Similar to OpineDB (LI et al., 2019), our work structures, and stores opinions to be further processed. However, while OpineDB only deals with subjective information, our work integrates subjective information with objective data in the PG. Also, OpineDB organizes the obtained information on a relational database schema, different from PGOpi that works with a flexible schema. Thus, instead of developing a whole new model for query processing as in OpineDB, one can leverage methods that already exist for knowledge graphs (ZHENG et al., 2018; ZHANG et al., 2018) for querying both forms of information in an integrated way.

Although our work is somewhat similar to Kobren et al. (2019) in the sense that we use a set of user opinions as input, and both methods associate factual with subjective data. Our approach stores numerous user opinions about product attributes, not just whether an opinion is true or false. Our method works solely by learning from the opinions available on the Web and does not require submitting surveys to people. Therefore, the method by Kobren et al. (2019) would not be appropriate to the task we address, i.e., autonomously organizing opinions around product attributes.

Li et al. (2020) build a PG from structured information on the Web: information from products and its hierarchical structure. Additionally, the authors explore chatbot conversations with users to capture users' interest. Although this work looks for subjective information and structures them to PG, it looks to solve the problem of recommendation task explanation, specifically, relating product items in the PG to users' needs obtained from chatbots. Our work is straightforward: unsupervisedly extract opinions and aspects terms from users' reviews then map them to product items in the PG.

3.4.3 Opinion Extraction

Table 3 – Some traditional and state-of-the-art works performing Opinion Extraction. UNSUP. - Unsupervised Approaches, RULE - Rule-based Approaches, ATE - Aspect Term Extraction, OTE - Opinion Term Extraction, CO - Co-extraction of aspect and opinion terms, TOWE - Target-oriented Opinion Words Extraction, AOPE - Aspect-Opinion Pair Extraction (the same as PAOTE - Pair-wise Aspect and Opinion Terms Extraction), CL - Sentiment Classification. The highlighted row corresponds to the approach currently applied in our pipeline.

	UNSUP.	**SUPERVISED**					
	RULE	**ATE**	**OTE**	**CO**	**TOWE**	**AOPE**	**CL**
Hu and Liu (2004)	X						
Zhuang et al. (2006)	X						
Tai and Kao (2013)	X						
Poria et al. (2014)	X						
Vicente, Agerri and Rigau (2014)	X						
Irsoy and Cardie (2014)			X				
Liu, Joty and Meng (2015)			X				
Wang et al. (2016)				X			
Wang et al. (2017)				X			
Li and Lam (2017)				X			X
Xu et al. (2018)		X					
Li et al. (2018)		X					
He et al. (2019)				X			
Fan et al. (2019)					X		
Zhou et al. (2020)					X		
Chen et al. (2020)						X	
Zhao et al. (2020)						X	
Wu et al. (2020a)						X	X
Wu et al. (2020b)					X		
Zhang et al. (2021a)					X		
Zhang et al. (2021b)					X		
Feng et al. (2021b)					X	X	
Gao et al. (2021)					X	X	
Wu et al. (2021)						X	

Source: Created by the author

In this subsection, we discuss state-of-the-art works related to the Opinion Extraction field. Here, we present the main differences between them and the approach applied by our **P**roduct **G**raph enriched with **Opi**nions (PGOpi) pipeline (See Table 3). Since our PGOpi pipeline was mainly focused on investigating the organization of subjective information into Product Knowledge Graphs, initially, we were looking for a direct way to extract this information (opinions or aspects). Hence, we decided by using an already established approach. Also, considering the lack of labeled training samples for this task

(extract opinion information), we opted to apply an unsupervised method. Given these observations, we employ the method proposed by Poria et al. (2014).

As shown before, Poria et al. (2014) explores external information to build an opinion lexicon and rules that allow its extraction alongside aspect terms. A highlight of this approach different from previous ones (HU; LIU, 2004; ZHUANG et al., 2006) is that it looks for aspect and implicit aspect terms in the review, and beyond syntactic rules, it explores a semantic lexicon. Additionally, it differs from other works as Tai and Kao (2013) and Vicente, Agerri and Rigau (2014) because these latter works focus only on opinion words and their respective polarities.

As shown in Table 3, most works existing in the literature are supervised, and in the last few years, the pair-wise extraction of aspect and opinion terms (TOWE and AOPE) has gained much attention. Due to this, we devised the **SYN**theti**C** **OP**inion**A**ted **T**ripl**E** (SYNCOPATE) to improve the TOWE and AOPE tasks. Our goal with SYNCOPATE was to explore pre-trained language models with In-Context Learning (see Section 3.3.1) to surpass the problem of labeled training data.

3.4.4 Language Models and In-Context Learning for Data Augmentation

Here, we discuss and compare the differences and similarities of our **SYN**theti**C** **OP**inion**A**ted **T**ripl**E** (SYNCOPATE) generation pipeline to some recent approaches exploring Language Models and In-Context Learning for the ABSA tasks. We focus mainly on SOTA approaches for Data Augmentation and ABSA. However, we also discuss approaches applied to other contexts than this but that is somehow similar to our SYNCOPATE approach. These works are organized in Table 4.

All of these works listed in Table 4 aim at Data Augmentation on the evaluated tasks, except the work of Wang et al. (2021) which aims at translating text to triples for Open Information Extraction. Due to the similarities, one might think it has a direct relation with our work, here we discuss their differences and why they are not strongly related to our problem. Wang et al. (2021) use a pre-trained LM to translate an NP-chunked input text to a triple format. The triple format generated by them is focused on relations between entities, such as the evaluated tasks. Hence the works and their applications are different. Our data augmentation pipeline does not look only for extracting aspect-opinion pairs and the relation between them in a given opinionated sentence, instead, we try to

Table 4 – Some recent works exploring Language Models and In-Context Learning for Data Augmentation. We list the NLP tasks evaluated by each of them and the strategy performed to generate augmentation data.

	STRATEGY	**EVALUATED TASKS**
Wang et al. (2021)	Zero-shot translation and Ranking	Open Information Extraction Relation Classification Factual Probe
Wang et al. (2021)	Few-shot Prompt Generation	Sentiment Classification Textual Entailment Question Answering Common Sense Reasoning Word Sense Disambiguation Coreference Resolution
Tavor et al. (2020)	Fine-tuning	Text Classification
Abonizio and Junior (2020)	Fine-tuning	Text Classification
Bayer et al. (2022)	Fine-tuning	Sentiment Classification Topic Classification
Liu et al. (2020)	Reinforcement Learning on a Conditional Generation LM	Offense Detection Irony Classification Sentiment Classification
Yoo et al. (2021)	Prompt Engineering	Sentiment Classification Text Classification
Min et al. (2022)	Meta-Learning	text classification question answering natural language inference paraphrase detection
Ding et al. (2020)	Language Model Inference	Named Entity Recognition Part of Speech Tagging End-to-End Target based Sentiment Analysis
Li et al. (2020)	Masking Words	Aspect Term Extraction
Hsu et al. (2021)	Masking Words	Aspect Category Sentiment Classification Aspect Term Sentiment Classification Aspect Term Extraction Sentiment Classification
Li, Yu and Xia (2022)	Transferring Knowledge	Aspect and Opinion Co-Extraction

Source: Created by the author

generate opinionated sentences and at generation time also extract aspects and opinion words mentioned in this synthetic sentence. Hence, these are different kinds of triples.

Another work is Wang et al. (2021) which is also different from our SYNCOPATE generator. Wang et al. (2021) is applied to tasks that do not require well-formatted examples with various restrictions as the opinionated triples generation. Also, it requires a task description, initial prompt examples, and a label description that will be assigned to the generated text. On the other hand, our work does the pretraining of the LM on the expected format to be generated, requiring only a few labeled examples for this. Addi-

tionally, the generated text is at the same time, the training sentence and the labels used to train models for aspect-opinion pair extraction.

SYNCOPATE deals with the complex task of generating opinionated triples different from Tavor et al. (2020), which focuses the work on text classification conditioning the generated text to the expected class. As stated previously, our work generates an opinionated sentence and at generation time tries to extract pairs of mentioned aspects and opinion expressions.

Beyond the differences in the application of our SYNCOPATE generator and the work of Liu et al. (2020), the latter applies a more complex approach while the former takes a more straightforward strategy. The authors of (LIU et al., 2020) convert the LM into a conditional generator, and for the text classification task, they guide the generation to the desired class with reinforcement learning. Our work continues the pretraining of an LM and using In-Context Learning it receives the well-formatted generation text.

Different from Yoo et al. (2021) we do not rely on prompt engineering. The authors also made use of a larger LM: GPT-3, which does not require any finetuning or TAPT, mostly because of its large number of parameters and training set, and also the low complexity of the evaluated tasks (text and sentiment classification).

The work of MetaICL by Min et al. (2022) requires a lot of effort since it performs meta-training using few-shot learning on the LM. For experiments, the authors report using 142 datasets in different tasks for this strategy and 52 tasks as targets. Additionally, the authors report that tasks requiring retrieval are a limitation of the proposed approach.

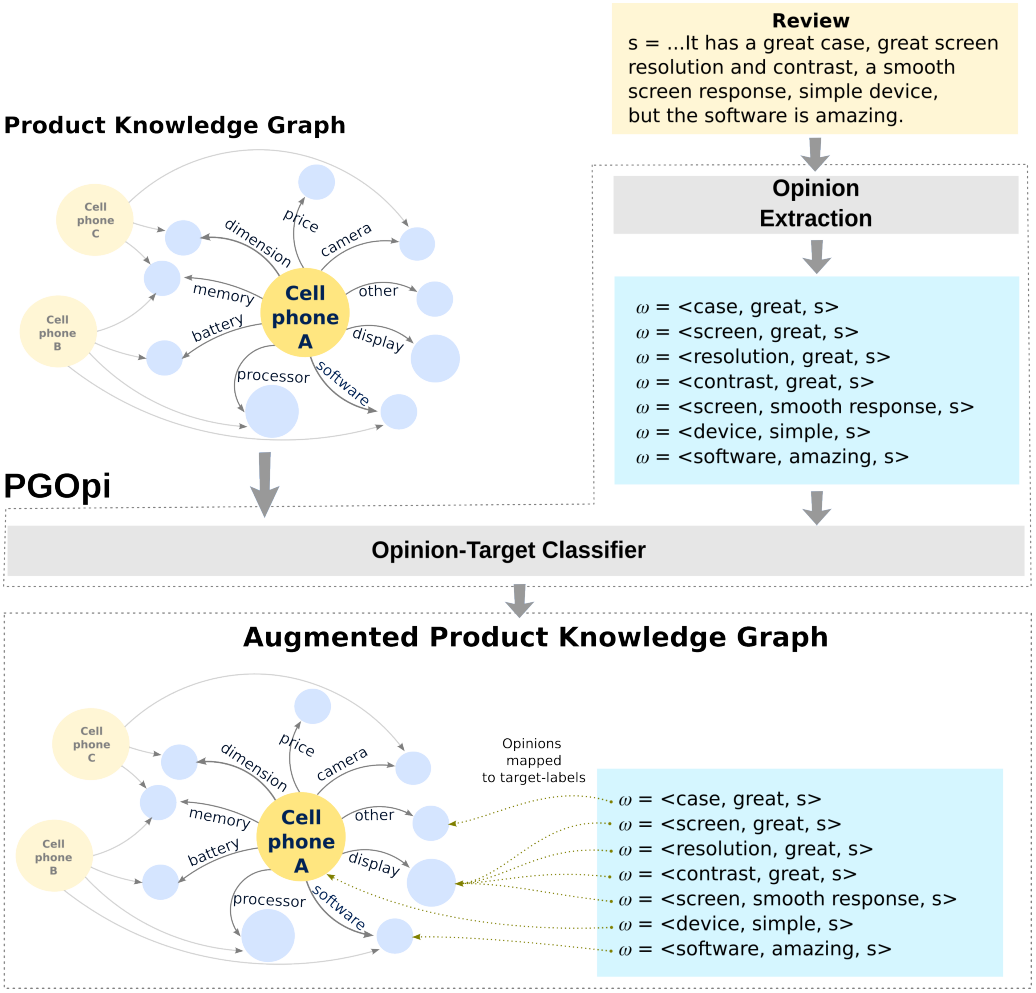
Different from our work, Ding et al. (2020) relies on training an LM from scratch with linearized labeled sequences in order to generate new labeled instances. The End-to-End Target-Based Sentiment Analysis (E2E-TBSA) task, although similar, is different from TOWE and AOPE tasks since the former seeks to extract the aspects with their respective sentiment polarities instead of sentiment words.

Although the work of Li, Yu and Xia (2022) focuses on the Aspect and Opinion Co-Extraction, this task is different from TOWE and AOPE since the former does not consider the relationship between the two elements. Hence, the co-extraction of aspects and opinion words are independent for each element. The authors train an LM using the Masked Sequence-to-Sequence approach and the model can either be used to predict word labels or generate new opinionated labeled sentences.

4 METHODOLOGY FOR ENRICHING PRODUCT GRAPHS WITH USER OPINIONS

This chapter presents **P**roduct **G**raph enriched with **Opi**nions (PGOpi)[1], our proposed model for enriching Product Graph (PG) with User Opinions. Additionally, beyond solving the addressed task, the proposed approach tries to circumvent missing labeled training data. This approach embraces some already consolidated Aspect-Based Sentiment Analysis (ABSA), Natural Language Processing (NLP), and Machine Learning techniques to build the model. Although the proposed model already presents some contribution to the Database community, we intend to improve this pipeline by using state-of-the-art techniques like the ones shown in Chapter 3.

Figure 2 – PGOpi pipeline at prediction time using an opinion-target classifier to map unseen opinions to the Product Knowledge Graph.

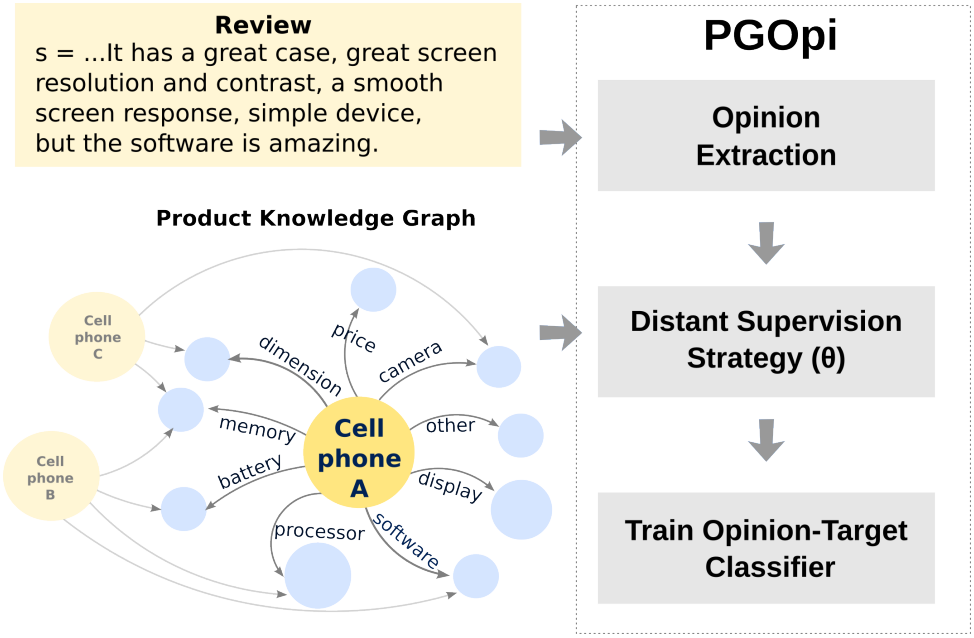


Source: Created by the author

[1] This chapter presents part of our work published in Moreira et al. (2022)

As shown in Figure 2, the proposed method PGOpi aims at mapping extracted opinion triples to product targets in the Product Graph. It receives as input an already known Product Knowledge Graph to be enriched and a set of reviews. PGOpi extracts opinionated triples from the reviews and the opinion-target classifier performs the mapping between these extracted opinions and the targets in the PG. Since our proposed approach does not rely on labeled data, we automatically assign labels to the extracted opinionated triples using a Distant Supervision Strategy as shown in Figure 3

Figure 3 – PGOpi pipeline for building training examples for the opinion-target classifier to map opinions to product targets in the knowledge graph.



Source: Created by the author

Hence, as shown in Figure 3, PGOpi is composed of an Opinion Extraction module, a Distant Supervision Strategy for labeling training data, and an opinion-target classifier that maps extracted opinions to targets in the PG. The Opinion Extraction module is detailed in Section 4.1. Section 4.2 presents the Distant Supervision Strategy. Section 4.3 discusses the architecture and training of the Opinion-Target Classifier. Additionally, we present some attempts to include different deep learning mechanisms that have not shown significant results when considering their computational complexity in the final architecture.

4.1 OPINION EXTRACTION

Given a set of reviews on a product, we aim to extract the opinions about aspects of this product. we apply the unsupervised method proposed by Poria et al. (2014) to extract aspect expressions and the sentiment words in the sentence review. We have chosen it mainly because it works specifically in the domain of product reviews, and it is a fully unsupervised model that does not require labeled data. Other unsupervised works we found under the ABSA task focus only on aspect words and sentiment polarities. In this work, we need to extract the opinion expression directly related to some aspect, which is done by Poria. Hence, given a review on a product, using Poria's approach we segment the review into sentences and apply the rules of Poria et al. (2014) to extract the opinion information in an unsupervised manner.

This approach explores external information from common-sense knowledge, as the implicit aspect corpus developed by Cruz, Gelbukh and Sidorov (2014) and SenticNet (CAMBRIA; OLSHER; RAJAGOPAL, 2014). Additionally, the approach also explores dependency trees to build the rules and extract the opinions. The rules defined by Poria et al. (2014) are based on two directions: i) rules for sentences with subject verb; and ii) rules for sentences which do not have subject verb. Given the amount of rules defined by Poria et al. (2014) we refer the original work for major details on each rule and below we give some examples of how the rules are applied to extract the aspects and build our opinion triples.

- Example of opinion extraction on sentences with subject verb:

  Review Sentence: *"The **camera** is nice."*

  According to the *copular relation rules*, the token camera is extracted as an explicit aspect because it is a *Noun* in a sentence with a copular relation. A copular relation consist of the relation between the *complement of a copular verb* (in this case the word nice) and the *copular verb* (is). The rule checks the implicit aspect lexicon(CRUZ; GELBUKH; SIDOROV, 2014) and once the copular verb exists in the lexicon, the token nice is also extracted. The original work treats the token nice as an aspect, but for our case we adapt the rule and treat the token as an opinion word.

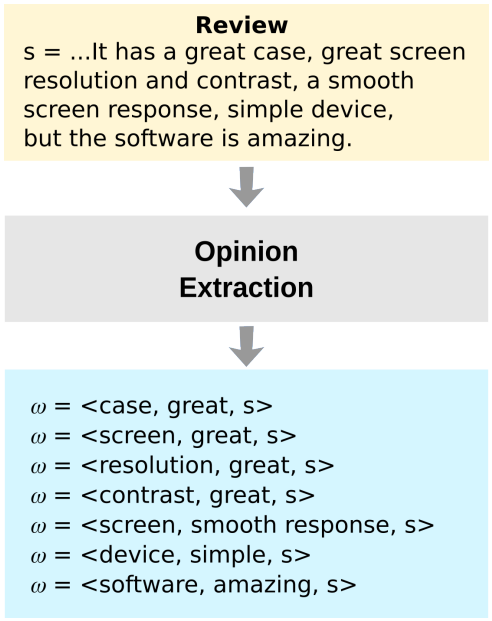
- Example of opinion extraction on sentences without subject noun relation in their parse tree:

  Review Sentence: *"Love the* ***sleekness of the player****"*

  According to the *prepositional relation rule*, the tokens sleekness of the player are retrieved as aspects from the sentence because of their relation through the proposition of. Since Love is present as a concept in SenticNet (CAMBRIA; OLSHER; RAJAGOPAL, 2014) and it is in *direct object relation* with sleekness which is connected to player in a *prepositional relation*, we extend Poria et al. (2014) rules by extracting the token Love as an opinion word.

We represent opinions as triples $\omega = \langle \alpha, w, s \rangle$, where α is the aspect of the target entity on which the opinion has been given, w is the sentiment word of the opinion, and s is the sentence from which the opinion was extracted, see Figure 4. Notice that, as in previous work (MELO et al., 2019; LIU, 2015), additional components, such as the sentiment polarity of the opinion towards aspect α, the opinion holder, and the opinion posting time, can also be considered to represent an opinion. However, and without loss of generality, these three components are enough in our setting.

Figure 4 – Opinion Extraction Module.



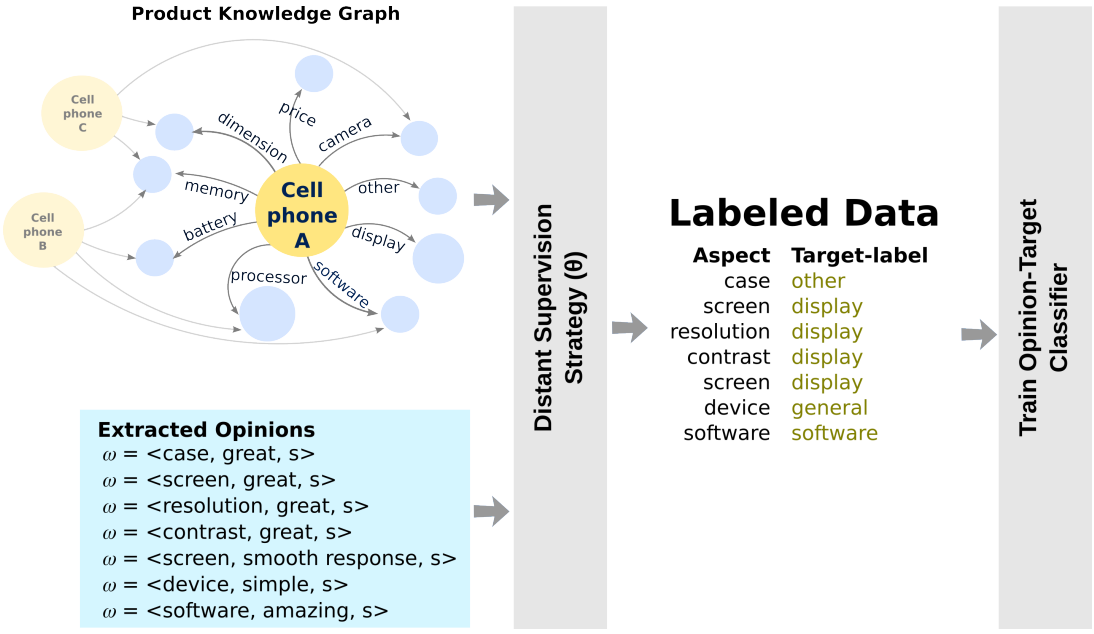
Source: Created by the author

4.2 THE DISTANT SUPERVISION STRATEGY

We have to map the extracted target-opinion pairs to products in the Knowledge Graph (KG) to solve our problem. However, we suffer from the absence of labeled training data to perform supervised learning, and unsupervised approaches to perform this type of mapping are usually less effective and require specialized intervention to set parameters. Additionally, since different products under different product categories can present many attributes/targets, manually building a training dataset for each target requires even more effort. Hence, automatically build these training datasets is crucial for this task.

To surpass these problems, we apply the weak-supervision approach called Distant Supervision, see Figure 5. As shown in Section 2.4, Distant Supervision allows building training datasets for supervised tasks by using simple heuristics. Mintz et al. (2009) introduce the approach by aligning natural language sentences with knowledge bases tuples to build training labels for the relation extraction task. Inspired by this approach, we build a similarity function that aligns the extracted opinion to product attributes in the PG and outputs representative training instances. The output of this approach is the labeled training dataset for each product target, which we use to train the Opinion-Target Classifier that maps the extracted opinions to their respective product target in the PG.

Figure 5 – Distant Supervision module for assigning labels to instances.



Source: Created by the author

Exploring the attributes (properties) and values of the products in the PG, we use them to label the extracted opinions. For that, we take the cosine similarity between the embedding vector representations from the product attribute-value pair and the opinion aspect. In other words, we build a set with the terms (words) belonging to the attribute name and the attribute value. To this set, we give the name target descriptor ($\Delta(tg)$). These target descriptors are related to target-labels (t), which are the names of the product's attributes and other two additional attributes: general and other, that will be detailed below. Similarly, we build a set with the terms from the aspect expression (α) that builds the opinion (ω).

The intuition behind this matching function is that aspect expressions (α) and the target descriptions ($\Delta(tg)$) containing terms that frequently occur in similar contexts are likely to be related. Hence, we measure the cosine similarity between these terms, one term from each set (target descriptor and aspect expression). When applying the match function 4.1, if the value is higher than a predefined threshold value ϵ, we consider a match between the opinion (ω) and the product target (t). If extracting an opinion on some product aspect, but a match does not occur between this opinion and the known product's attributes in the PG, we assume that the opinion can refer to another attribute of the product that is currently unknown by the PG. Hence, we introduce the target (t) "other" to include these opinions in the training set. The target "general" is added to the target-labels (t) representing the product itself, hence all opinions mentioning directly the product are organized into this target-label.

$$Match(\alpha, \Delta(tg)) = \begin{cases} 1 & \text{if} \underset{\substack{w \in \alpha \\ \delta \in \Delta(tg)}}{max} [\cos(\vec{w}, \vec{\delta})] \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

where $\vec{w}$ and $\vec{\delta}$ are, respectively, the word embeddings representation of each term $w \in \alpha$ and each term $\delta \in \Delta(tg)$. $\Delta(tg)$ is the target descriptors set, and α the aspect expression set.

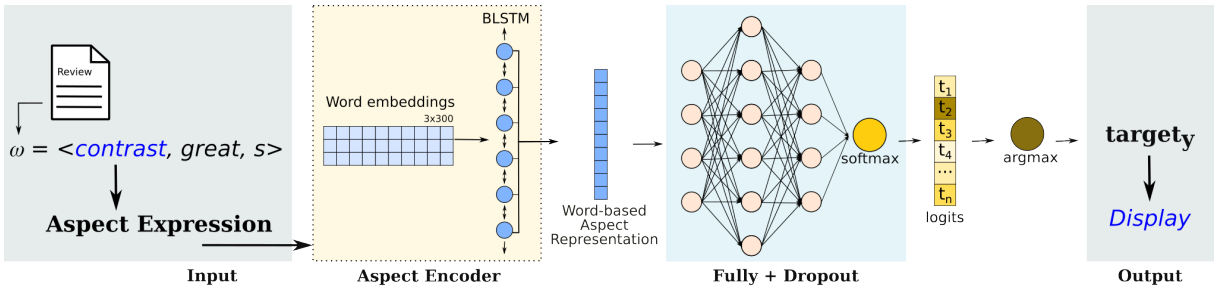
4.3 THE OPINION-TARGET CLASSIFIER

As aforementioned, the goal of the opinion-target classifier is to map extracted opinions to targets in the PG. One could argue that the inclusion of this classifier is unnecessary

since the extracted opinions already contain a target aspect, being necessary just the direct mapping using the similarity function. However, the target words present in users' reviews can suffer from many inconsistencies such as misspellings, the user can use different names to refer to the same product feature, and different aspects of the product can refer to the same product feature. Also, relying just on the similarity mapping would require constant human intervention to set the similarity threshold.

Based on that, we argue that a classifier is a better fit to perform the mapping. The mapping with a classifier keeps the product's structure in the PG without adding too much granularity if performing the direct mapping only by the opinion aspect. Also, learning the mapping patterns with classification allows better generalization for unseen examples. For example, without our opinion-target classifier it would be difficult to map aspects as "screen", "screen contrast", and "screen resolution" to the product target/feature "display". This could be performed by using our semantic similarity function, however it would still require a similarity threshold and yet some mappings could be missed. Additionally, training a classifier and fitting thresholds just once to build some automatically labeled instances allows capturing the patterns of opinions for the domain and, once trained, can be applied to unseen reviews in the same domain without setting thresholds each time. Hence, it allows the constant inclusion of information in the PG. Otherwise, it would be necessary to keep fitting the similarity threshold for each new instance, or much noisy information would be included in the PG.

Figure 6 – Deep Neural Network architecture of the Opinion-Target Classifier for performing the mapping between extracted opinions and product targets.



Source: Adapted from Moreira et al. (2022)

Our solution uses a deep neural model to predict a target-label (product attribute) to a given extracted opinion from product reviews. Figure 6 shows the network architecture. The model receives as input the extracted aspect α of the opinion ω and outputs the predicted target label t. We use Word2Vec (MIKOLOV et al., 2013) to pre-train word

embeddings fed to the aspect encoder. The aspect encoder, composed of a Bidirectional Long Short-Term Memory (BiLSTM) (GRAVES; SCHMIDHUBER, 2005), learns aspect representation from the aspect word sequence. We add a Multilayer Perceptron (MLP) layer aligned with dropout regularization to the model to mitigate overfitting. The MLP receives the encoder output and sends it to a Softmax activation function, which outputs probabilities of the given input aspect referring to a product's target t. An *argmax* function gives the final prediction returning the target label with the highest probability.

Word Embedding Layer

Word embeddings capture syntactic and semantic features of words in a given context. This dense representation embeds each word into a d dimensional space and can be learned during the model's training or initialized from vectors pre-trained on a corpus. For our task we use the Word2Vec model to build word embeddings based on the *Amazon Review Corpus*, which is better described in Chapter 5. We have opted for the Word2Vec model because it is already a well-established model to this end, and it is less computationally costly than fine-tuning a BERT model to get the embedding representation. Using the *Amazon Review Corpus* to train the Word2Vec model we could better capture the semantics of the words written in product reviews on the internet without too much effort. The embeddings dimension d was fixed in 300. Since not all aspects have the same length, before converting the raw input into embeddings we defined a maximum number of words (ρ) and performed padding. Given the words in each input aspect, the model retrieves from the word embedding layer the pre-trained word embedding vector for each token in these sequences. Hence, the embedding layer produces an aspect matrix $l_{asp} \in \mathbb{R}^{\rho \times d}$.

The Aspect Encoder

The representation of the words from the word embedding layer that compose the aspect is passed to capture contextual patterns within this sequence. Specifically in our solution, the encoder is implemented using sequence-based neural models since they have been successfully applied to sequence-based problems such as speech recognition, language modeling and language translation. In particular, we use Long Short-Term Memory

(LSTM) (HOCHREITER; SCHMIDHUBER, 1997). LSTM is composed of memory cells and gating mechanisms which allow the storage of information about items in the sequence for long sequences. The BiLSTM network is a modification of the LSTM model which looks at the previous and next contexts of a memory cell in order to predict the current state. This strategy enriches the representation of the words from their context. The aspect encoder presented in the proposed model architecture in Figure 6 is composed of a BiLSTM network. The output of the encoder is the resulting vector from forward and backward lookup over input sequences. Hence, taking u as the number of units (cells) in each LSTM network, the output vector of the encoder presents a dimension of $2 \times u$.

MLP and Output Layer

The aspect representation vector produced by the aspect encoder is passed to the next layer of our architecture: a MLP Network. In Figure 6, the MLP component is shown as one fully connected layer, followed by dropout. However, the number of fully connected layers can be optimized and defined for each dataset (more details on Chapter 5). The output layer of our model receives the resulting tensor coming from the last hidden MLP layer. The number of units in the output layer is given by the number of target-labels $t_n = |T|$ to be learned by the network. The softmax activation function is used to obtain the class-membership probability for each input, as given by

$$P(Y = t|x, W, b) = \frac{e^{W_t x+b}}{\sum_i e^{W_i x+b}} \tag{4.2}$$

where, x is the input vector, t is the label index, W the matrix of weights and b the bias vector.

Finally, we apply the *argmax* function:

$$t = \underset{t \in T}{\operatorname{argmax}}\, P(t) \tag{4.3}$$

to obtain the target-label t with the highest class-membership probability $P(t)$.

4.3.1 Training

For training, the network applies backpropagation to minimize its loss function. Given the imbalanced nature of our data, we applied a multi-class modification of the Focal Loss (LIN et al., 2020):

$$FL(y) = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} -(1-\hat{y}_{nt})^{\gamma} * y_{nt} * log(\hat{y}_{nt}) \tag{4.4}$$

where T is the number of target classes, N is the number of samples used to calculate the loss to update the network parameters, y_n and $\hat{y}_n$ with n=1...N are the one-hot vectors of the true labels and the corresponding predicted softmax output, respectively. $\gamma \geq 0$ is a tunable focusing parameter that is used to downweight easy examples and, as a result, to pay more attention on the hard ones.

4.3.2 Failed Attempts

Throughout the development of this work, we attempt different model architectures using some deep learning state-of-the-art mechanisms. They result in more complex models, and consequently higher execution time, without any significant gain in the results' quality. These attempts are listed below:

Sentence Encoder

In addition to the aspect encoder, we investigated adding a sentence encoder to the network, with the same layers of the aspect encoder: a BiLSTM on top of a word embedding layer. By doing that, we aimed to allow the network to capture more context from the aspect's sentence to perform the classification that might not be present in the aspect itself. Our evaluation showed, though, that adding this encoder does not significantly improve the classifier's effectiveness while increases the model complexity.

Character-based CNN

We tried a character-based Convolutional Neural Network (CNN) to generate a representation of the aspect based on its characters. For that, we added an input layer for aspect characters embeddings $l_{char} \in \mathbb{R}^{c \times a}$ where c is the maximum number of characters

and a is the maximum number of characters in each word, which we define as 12. This input is sent by a CNN layer, maxpooling, flatten, dropout, and a final fully connected layer. The output of these operations is a dense vector of dimension $char\_d$ given by the number of neuron units at the fully connected layer, which is automatically defined during parameters optimization. The intuition is that this representation would capture the local composition of characters in the aspect, possibly allowing a better separation of some attributes, as zoom, exposure_control, imaging, storage, memory and so on, which sometimes can contain character chains over the aspect feature, e.g. f/1.7, m4/3 lens, 80mm, 12mp, 18x, 256B, and 1600x900. Preliminary experiments show, however, no significant improvement of adding the character-based CNN to the aspect encoder in comparison to our proposed network. Hence, given the increase in the model's complexity and no significant improvement, we do not consider it for final experimentation.

Self-Attention

Another attempt was to use the self-attention mechanism to learn a representation for a sequence of words. In our context, we would expect that the sentence representation would carry additional information about the aspect. For that, we assign a random vector as query and the BiLSTM hidden states as key/value. This mechanism allows the capture of the most significant hidden states coming from the BiLSTM, assigning higher weights to the more representative hidden states and lower weights to the less representative ones. The intuition was to get additional representation both for aspect and for sentence. It turns out that adding these representations did not bring any improvement, so we keep it out from final evaluation.

Co-Attention

We also tried to apply co-attention to produce joint representations of the aspect and sentence. For the sentence representation, we consider the hidden states of the sentence's BiLSTM as the query on the attention mechanism and the hidden states from the aspect's BiLSTM as key/value. We perform a similar strategy to build the aspect representation. Similarly to the previous attempts, these co-attention representations did not improve the results. However, in order to report the performance of attention mechanisms on this task we include this model in our final experiments.

5 PGOPI EVALUATION

In this Chapter we present the experiments performed to evaluate our PGOpi[1] approach. First, in Section 5.1 we present the experimental setup describing training and test datasets, the word embeddings setup, the metrics and baseline works used for benchmarking our work. Section 5.2 presents the experimental results and the analysis benchmark.

5.1 SETUP

5.1.1 Datasets

We collected product data from two distinct e-commerce websites: Amazon[2] and BestBuy[3]. Both collections are composed of reviews and specifications of products in five categories: photographic cameras (CAMERAS), cellphones (CELLPHONES), DVD players (DVDS), laptop computers (LAPTOPS), and internet routers (ROUTERS). These are notoriously popular categories among consumers of electronic products, and they have been previously explored on opinion mining research (LIU et al., 2017; MCAULEY; YANG, 2016; MCAULEY et al., 2015). Table 5 presents the set of target-labels for the products of each of the categories we used. Notice that target-labels in all categories include other and product.

Table 6 presents the number of user reviews and sentences for each data collection along with the number of products referred in the reviews for each category. Although the two collections are essentially similar in terms of structure and content, we used each dataset for different purposes in our work. Since the Amazon dataset is much larger than the BestBuy one, we used it to train the models and for evaluation we manually annotated a hold-out set of each dataset. We present more details about those datasets later in this section.

[1] This chapter presents part of our work published in Moreira et al. (2022)
[2] https://www.amazon.com/
[3] http://www.bestbuy.com

Table 5 – Set of target-labels for each product category in our datasets.

CATEGORY	TARGET-LABELS
CAMERAS	dimension, exposure control, imaging, performance, power, price, zoom, other, product
CELLPHONES	battery, camera, dimension, display, memory, price, processor, software, other, product
DVDS	audio, dimension, price, video, other, product
LAPTOPS	battery, connectivity, dimension, graphics, memory, price, processor, screen, software, storage, other, product
ROUTERS	accessory, coverage area, dimension, port, price, security, software, speed, other, product

Source: Our work published in Moreira et al. (2022)

Table 6 – Overview of the datasets.

CATEGORIES	*Amazon*			*BestBuy*		
	Products	**Reviews**	**Sentences**	**Products**	**Reviews**	**Sentences**
CAMERAS	12K	1M	3M	12	246	606
CELLPHONES	15K	1M	1M	20	372	1K
DVDS	1K	45K	45K	8	159	372
LAPTOPS	7K	240K	240K	20	376	1K
ROUTERS	9K	1M	1M	10	237	607
Total	**44K**	**3M**	**10M**	**70**	**1K**	**3K**

Source: Our work published in Moreira et al. (2022)

5.1.2 Test Sets

To serve as a golden standard, the opinionated sentences in *Amazon* and *BestBuy* collections were identified manually and each opinion ω in a sentence was annotated as follows: (i) if ω is an opinion on an attribute from the product graph, ω is annotated with the *attribute name*; (ii) if ω is an opinion on the product as a whole, ω is annotated with general; (iii) if ω is an opinion on a product characteristic that is not represented as an attribute in the product graph, ω is annotated with the label other.

For the BestBuy collection, we have annotated 405 opinions in CAMERAS, 621 opinions in CELLPHONES, 279 opinions in DVDS, 680 opinions in LAPTOPS, and 362 opinions in ROUTERS. In the case of the Amazon Collection, it would be infeasible to

Table 7 – Summary of the training datasets built from *Amazon Collection*. The numbers refer to the size of the dataset in number of training samples.

	PRODUCT CATEGORIES				
ϵ	**CAMERAS**	**CELLPHONES**	**DVDS**	**LAPTOPS**	**ROUTERS**
0.5	812.685	389.039	321.394	521.087	352.942
0.6	719.351	375.518	270.168	414.113	271.937
0.7	667.747	341.366	213.972	360.357	243.278
0.8	556.661	330.045	154.428	318.452	133.241
0.9	551.581	328.387	102.302	312.294	128.65
TOTAL	3.308.025	1.764.355	1.062.264	1.926.303	1.130.048

Source: Our work published in Moreira et al. (2022)

manually annotate all the opinions identified due to the volume of sentences. Thus, we created a random sample of 400 opinionated sentences for each product category from the entire set of reviews. This was sufficient to allow a confidence level of 95% in the results of our experiments. We invited two annotators to manually label the opinions identified in this sample for each of the five product categories. The average inter-annotator agreement on classifier prediction annotation was $k = 0.676$ (standard error = 0.0179) according to Cohen's Kappa statistic.

5.1.3 Training Sets

As aforementioned, we built the training sets with data from the Amazon collection using the Distant Supervision (DS) process described in Section 4.2. We applied DS strategy varying the matching threshold from 0.5 to 0.9. This step was required in order to investigate the best fitting threshold for the similarity function. Table 7 shows the number of instances for each threshold/category dataset.

5.1.4 Word Embeddings Setup

We used *Word2Vec* to generate the word embeddings trained over the set of user reviews from the Amazon collection for both the DS process and the opinion-target classifier. We performed the following steps to clean the reviews and prepare them for building the embeddings. First, we split the reviews into sentences. Second, we performed a term-

based tokenization, discarding punctuation marks and symbols. In addition, we removed sentences with less than 5 terms and transformed each term to lowercase. To obtain the *Word2Vec* representation of terms, we used the skip-gram model (MIKOLOV et al., 2013) with the following parameters: embeddings' dimension = 300, epochs = 25, learning rate = 0.25, and word count threshold = 5. The training and test sets, and pre-trained word embeddings are public available[4], as well as the code[5] for all experiments here performed.

5.1.5 Approaches

To validate our approach, we use OpinionLink (MELO et al., 2019) as baseline and also report the results of the other attempts discussed in Section 4.3.2. Specifically, we executed the following approaches:

- **OpinionLink**: The method uses a Support Vector Machine (SVM) classifier to map the opinions extracted from user reviews to the attributes of the products in the product catalog. The authors of *OpinionLink* assume that all the products from the same category have the same set of attributes, while PGOpi is more flexible and allows the products of the same category may have different sets of attributes. For this evaluation, we adapted *OpinionLink* to work over a product graph. Notice also that, *OpinionLink* is fully supervised and requires a large amount of manual labeled data, while PGOpi uses a distant supervision strategy to reduce the dependency of manual training data. For futher details on *OpinionLink*, please refer to Section 3.2.
- **PGOpi**: The proposed approach described in Chapter 4. It receives as input only the aspect expression α.
- **PGOpi**$_{sent}$: The model receives as input only the sentence s from an opinion ω.
- **PGOpi**$_{asp+sent}$: The model receives the sentence s and the aspect expression α from opinion ω as input of the model.
- **PGOpi**$_{co-att}$: The sentence s containing the opinion ω is given as additional input to the network. The representations coming from sentence and aspect encoders are sent through a co-attention mechanism to build joint representations for words in

[4] <http://tiny.cc/rk0wtz>
[5] <https://github.com/guardiaum/PGOpi>

Table 8 – Search space applied for hyper-parameters optimization. FC: Fully Connected.

	Hyper-parameter	**Search Space**
Aspect and Sentence encoders	LSTM units	[50, 100, 150, 200]
FCs before output layer	Number of FC Layers	[1, 2, 3]
	Units (first layer)	[100, 200, 400, 600, 800]
	Units (second layer)	[50, 100, 200, 400, 600]
	Units (third layer)	[25, 50, 100, 200, 400]
Focal Loss	Gamma	[1.0, 2.0, 3.0, 4.0, 5.0]

Source: Our work published in Moreira et al. (2022)

the sentence and word in the aspect. We use this model to compare a network based on attention mechanism with our proposed approach PGOpi.

5.1.6 Hyper-parameter optimization and training

Since we build a model for each threshold of each product category, we have optimized the model hyper-parameters regarding all datasets using a validation set, which comprises 30% of the training set. The tuned hyper-parameters and their respective search space are shown in Table 8. We fit the model parameters for each dataset built for each threshold ϵ for each product category. The number of epochs was defined by an early stopping strategy after 5 iterations with no significant improvement in Area Under the Curve (AUC) over the development set. The model was trained with the Adam optimizer (KINGMA; BA, 2015) with 0.001 of learning rate. Dropout rates are set in 0.5. We also vary the number of fully connected layers before the output layer, after concatenating the representations obtained. The hyper-parameter optimization was performed over 20 trials with the HyperOpt (BERGSTRA; YAMINS; COX, 2013) implementation of the Tree-Structured Parzen Estimator (TPE) (BERGSTRA et al., 2011). Since the results of Deep Learning models are susceptible to random seed noise, we perform 10 training runs of each model architecture using the selected best hyper-parameters. The results are reported as the average of these runs.

5.1.7 Evaluation Metrics

We used *precision*, *recall*, and F_1 evaluation metrics. These metrics are calculated as follows. Let A be the set of correct answers, according to a reference set, and let B be the set of answers generated by the method being evaluated. We define precision (P), recall (R) and F_1 as: $P = |A \cap B|/|B|; R = |A \cap B|/|A|; F_1 = 2 \times (P \times R)/(P + R)$. Since we are dealing with a multi-class problem and unbalanced datasets we calculate the micro measures. However, to save space and given the multi-class aspect of the problem we summarize the results reporting the Micro F-score measure. Also, the reported results are based on the model with the highest F_1 value on the validation data of evaluated training datasets.

5.2 EXPERIMENTAL RESULTS

Table 9 shows the results obtained from the evaluated models. Our proposed approach *PGOpi* shows superior performance in all scenarios when compared to its variation with co-attention ($PGOpi_{co-att}$). This indicates that using the joint attention-based representations of the sentence and aspect does not give any contribution for this task. In fact, using a simple concatenation of sentence and aspect as input does not improve the model's performance as well, as the numbers of $PGOpi_{sent+asp}$ confirm. Regarding $PGOpi_{sent}$, the results in Table 9 show that the sentence alone does not add significant context information to help the classification model.

To verify whether the difference between the models' performance is statistically significant, we ran the Friedman Test over each product category from each corpus using 10-run executions after fitting parameters. We consider a significance level of 0.05 and the null hypothesis H_0 the following: *There is no significant difference between the models with different inputs*. Excepting for DVDS on the Amazon dataset, the Friedman Test rejected H_0 for all the other scenarios (each product category from each corpus). We also executed the Nemenyi test post-hoc test to find which models differ in those cases. H_0 could not be rejected only for the pair *PGOpi* and $PGOpi_{sent+asp}$ for all cases. Even though the performance of these two models are equivalent, the $PGOpi_{sent+asp}$ variation is more complex in terms of input size which can imply on more computation time and power. Hence, we can conclude from these results that our proposed model *PGOpi*, using

Table 9 – Micro F-score results for each benchmark model and the proposed model PGOpi. Results for PGOpi$_{co-att}$ and PGOpi are the average of 10-run executions of the best trained model over test set. The bullets (•) indicate scenarios where the baseline fully-supervised model achieves the best result for the task. Here, the reported results for the PGOpi models are selected based on the best threshold ϵ. Full results across thresholds (ϵ) are shown in Table 10.

	MODEL	**AMAZON**	**BESTBUY**
CAMERAS	OpinionLink	0.90•	0.85•
	PGOpi	0.85	**0.82**
	PGOpi$_{sent}$	0.82	0.60
	PGOpi$_{sent+asp}$	**0.85**	0.82
	PGOpi$_{co-att}$	0.45	0.66
CELLPHONES	OpinionLink	0.87•	0.84
	PGOpi	**0.85**	**0.87**
	PGOpi$_{sent}$	0.84	0.59
	PGOpi$_{sent+asp}$	**0.85**	0.87
	PGOpi$_{co-att}$	0.77	0.65
DVDS	OpinionLink	0.90•	0.85•
	PGOpi	**0.75**	**0.85**
	PGOpi$_{sent}$	0.74	0.56
	PGOpi$_{sent+asp}$	0.75	0.84
	PGOpi$_{co-att}$	0.34	0.64
LAPTOPS	OpinionLink	0.81	0.78
	PGOpi	**0.83**	**0.80**
	PGOpi$_{sent}$	0.78	0.56
	PGOpi$_{sent+asp}$	0.82	0.80
	PGOpi$_{co-att}$	0.30	0.56
ROUTERS	OpinionLink	0.92•	0.87
	PGOpi	0.88	0.88
	PGOpi$_{sent}$	0.85	0.63
	PGOpi$_{sent+asp}$	**0.88**	**0.88**
	PGOpi$_{co-att}$	0.44	0.71

Source: Our work published in Moreira et al. (2022)

only the aspect as input, performs better or equivalent to more complex models.

The *PGOpi* model also surpasses a full-supervised model, *OpinionLink*, in 5 out of 10 scenarios: LAPTOPS on the Amazon corpus; LAPTOPS, CELLPHONES, and ROUTERS on Bestbuy. In the other 5 domains, the biggest difference between the two models is in the DVDS on Amazon where *OpinionLink* obtained micro F-score equals to

0.9 whereas *PGOpi* 0.749. In the remaining scenarios, the results of those models were much closer. Furthermore, our distant-supervised strategy, obtained compared results with the sota fully-supervised approach, with the advantage of requiring no label effort, which makes it easily to build for new attributes or products.

Looking at these *OpinionLink* results, we conclude that the difference between the two methods (OpinionLink and ours) is small, indicating that PGOpi is as effective as the method considered as state-of-the-art. Additionally, PGOpi has the advantage of being semi-supervised and flexible when dealing with different attributes of products in the same category.

Table 10 presents the distribution of micro F1-score results on Amazon Golden Standard dataset over all evaluated thresholds ϵ. Looking at these distributions there is no evident pattern indicating a common best threshold across categories nor an agreement between the different models within the same category. Excepting for the category DVDS, where all models agree that the best threshold is 0.9, the other categories seem to diverge. The more dissonant category is LAPTOPS where amongst the four analyzed models three distinct thresholds are selected (0.5, 0.7 and 0.8). The three remaining product categories present closely selected thresholds: ROUTERS (0.6 and 0.7), CELLPHONES (0.5 and 0.6), and CAMERAS (0.5 to 0.7).

Table 10 also presents the results obtained for the BestBuy dataset, which was used as test set to validate the generalization of our model to different sources of review data. As in the case of Amazon, it is difficult to find a pattern of best threshold to select for each category and model. We can conclude from these numbers that the threshold selection is an important step since the results have shown a significant variation in the evaluated scenarios even in models under the same product category.

We present confusion matrices and classification reports in Appendix A for details on the performance of the PGOpi pipeline on each target of each product category. We select the best threshold from Table 10 and used the best model (combination of hyperparameters) to report these values. From those reports, we can notice that specific targets are often mislabeled as `other` and sometimes as `general`. Hence, because these are broader targets containing opinions about the product itself (`general`) and all other aspects not yet known by the Product Knowledge Graph (PKG) (`other`) it is expected that these targets will make the classification harder.

An important issue that *PGOpi* deals with is class imbalance. Figure 7 presents the

Table 10 – Micro F1-scores obtained over Amazon and BestBuy Golden Standard datasets for each threshold ϵ used for building training examples. *sent*, *sent+asp*, and *co-att* are the variations of the PGOpi model as described in **Approaches**. The values are the average of 10-runs. Boldface values indicate the selected best value for each model and product category. Ties were solved by considering three decimal places.

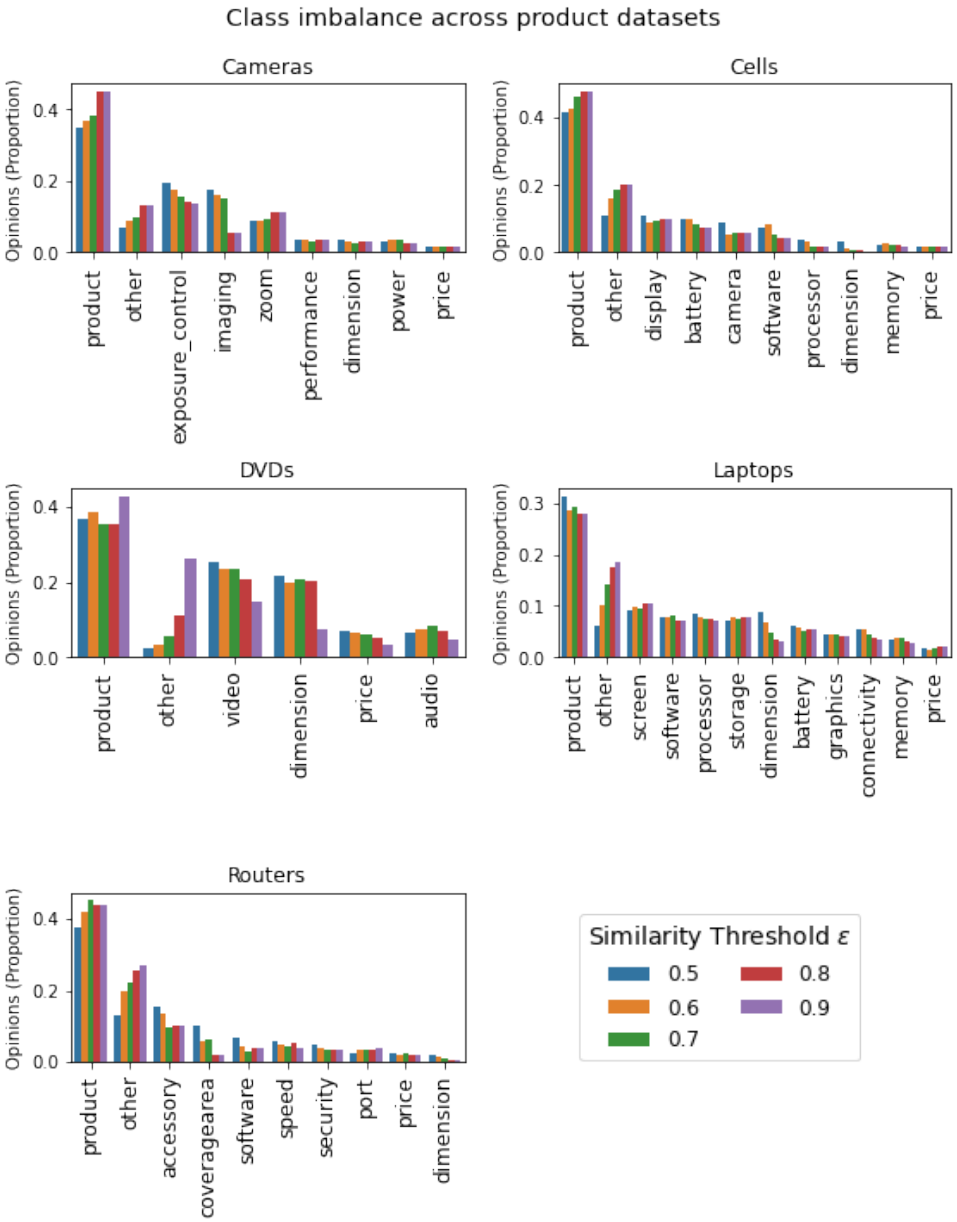
		AMAZON				**BESTBUY**			
	ϵ	PGOpi	*sent*	*sent+asp*	*co-att*	PGOpi	*sent*	*sent+asp*	*co-att*
CAMERAS	0.5	0.80	0.78	0.81	**0.45**	**0.82**	0.57	0.82	0.63
	0.6	**0.85**	**0.82**	0.85	0.35	0.82	0.59	0.82	0.63
	0.7	0.85	0.82	**0.85**	0.41	0.82	**0.60**	**0.82**	**0.66**
	0.8	0.73	0.69	0.73	0.33	0.73	0.55	0.72	0.53
	0.9	0.73	0.70	0.73	0.34	0.72	0.55	0.72	0.53
CELLS	0.5	0.85	**0.84**	0.84	**0.77**	**0.87**	0.51	**0.87**	0.58
	0.6	**0.85**	0.83	**0.85**	0.37	0.86	**0.59**	0.87	**0.65**
	0.7	0.85	0.83	0.85	0.39	0.86	0.59	0.86	0.62
	0.8	0.85	0.82	0.85	0.35	0.85	0.59	0.86	0.63
	0.9	0.84	0.83	0.85	0.39	0.84	0.58	0.85	0.63
DVDS	0.5	0.56	0.47	0.56	0.22	0.62	0.33	0.61	0.37
	0.6	0.60	0.49	0.60	0.27	0.68	0.34	0.68	0.46
	0.7	0.67	0.51	0.67	0.28	0.74	0.39	0.74	0.48
	0.8	0.68	0.56	0.67	0.18	0.84	0.47	0.83	0.52
	0.9	**0.75**	**0.74**	**0.75**	**0.34**	**0.85**	**0.56**	**0.84**	**0.64**
LAPTOPS	0.5	**0.83**	0.75	**0.82**	0.22	**0.80**	0.54	**0.80**	0.55
	0.6	0.81	0.78	0.81	0.27	0.78	**0.56**	0.78	0.51
	0.7	0.82	**0.78**	0.82	0.25	0.78	0.56	0.78	**0.56**
	0.8	0.77	0.74	0.77	**0.30**	0.70	0.53	0.70	0.49
	0.9	0.77	0.74	0.77	0.24	0.70	0.53	0.70	0.48
ROUTERS	0.5	0.78	0.73	0.78	0.34	0.85	0.57	0.85	0.62
	0.6	0.84	0.81	0.85	**0.44**	0.87	0.60	0.87	**0.71**
	0.7	**0.88**	**0.85**	**0.88**	0.41	**0.88**	**0.63**	**0.88**	0.63
	0.8	0.81	0.79	0.81	0.38	0.82	0.62	0.82	0.59
	0.9	0.80	0.78	0.80	0.36	0.80	0.61	0.80	0.65

Source: Our work published in Moreira et al. (2022)

class imbalance across target-labels for each ϵ applied to build the training sets. As expected, many of the reviews on the Amazon dataset contain opinionated sentences referring to the product itself (`general`). We can also notice that, excepting for `general` and `other`, the higher the similarity threshold ϵ, the smaller the number of training examples of the attributes.

As previously pointed out, the proportion of opinionated sentences related to attribute `other` impacts the threshold similarity selection and consequently the classification per-

Figure 7 – Proportion of the class unbalancing for the five product categories applied in this study. The proportion is obtained from each thresholds ϵ used for building training examples.



Source: Our work published in Moreira et al. (2022)

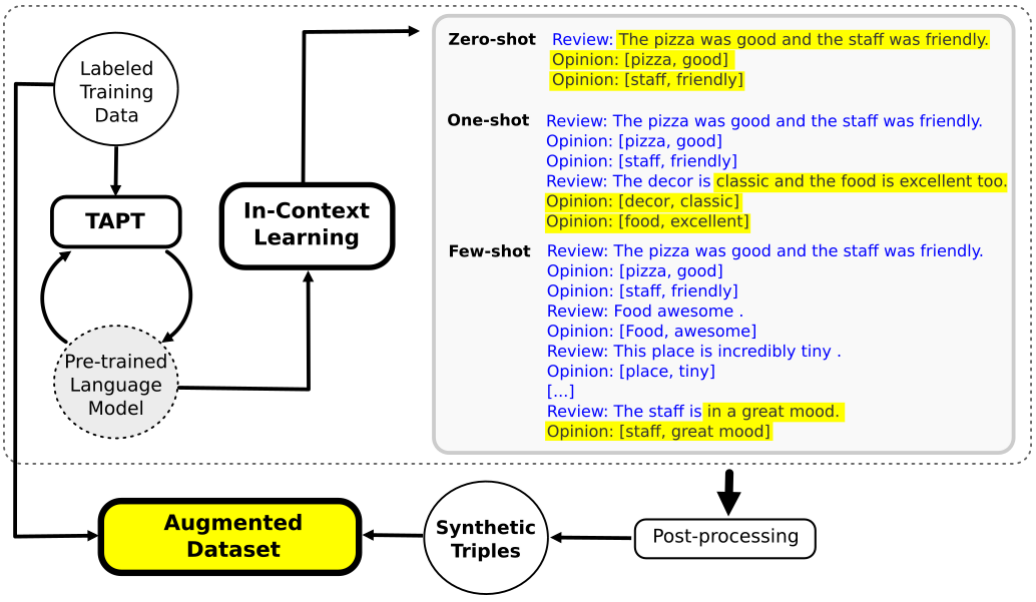
formance. As an example on DVDS category, the attribute `other` presents a very low proportion of instances in relation to other attributes. When this class is underrepresented, the models tend to predict false negatives for this class and false positives for the other ones, hurting the models' quality.

6 METHODOLOGY FOR GENERATING SYNTHETIC OPINIONATED TRIPLES

This chapter shows our solution to build synthetic opinionated triples to augment the performance of state-of-the-art models for Target-Oriented Opinion Word Extraction (TOWE) and Aspect-Opinion Pair Extraction (AOPE). We explore In-Context Learning on a pre-trained Language Model (LM) to investigate the possibility of mitigating the dependency on manually labeled data. We call this framework **Syn**thetic **Op**inion**a**ted **T**riple**s** (SYNCOPATE) generator.

The main intuition is to analyze the performance of the well-established models when trained with these synthetic triples. The manual labeling of training sets is expensive, time-consuming, and not scalable. Hence, removing the dependency on these types of data would be of great contribution to the exploration and organization of real data present on the Web.

Figure 8 – Pipeline of our framework SYNCOPATE for building opinionated triples composed of a synthetic sentence and the tuples of aspect and opinion words mentioned in it. Highlighted words are autonomously generated by the LM after Task-Adaptative Pretraining (TAPT).



Source: Created by the author

As shown in Figure 8, our solution first performs TAPT (GURURANGAN et al., 2020) on a pretrained Language Model. This is needed to indicate the generation format that the LM must follow. It then applies In-Context Learning to build synthetic examples and performs a post-processing step to filter out duplicates, out-of-vocabulary aspects, and opinions. The generated samples are finally merged into the original data to augment

existing Machine Learning solutions for TOWE and AOPE. Below we detail each of these steps.

6.1 TASK-ADAPTATIVE PRETRAINING (TAPT)

As previously stated, there are some challenges while trying to generate opinion triples using In-Context Learning: First, it requires a formatted text. For our problem, it is the triple format: <*sentence*, *aspect*, *opinion words*>. Second, the text generation must be in a particular domain. Third, the sentence must carry an opinion, and the words representing the aspect and opinion elements must coexist in the same generated sentence.

Large pretrained LM as they are made publicly available are general, i.e. trained over large corpora of unstructured text on a variety of topics. Although showing superior performances for a variety of Natural Language Processing (NLP) tasks, the generation and in-context learning approaches performed by these models without previous fine-tuning or TAPT are general and prompt-engineering dependent. Hence, they work well for direct simple tasks such as slot-filling, question answering, and building labels for classification and sometimes for sequence-labeling. Meanwhile, the generation required for our problem comprises some conditionals, which require a specific generation regarding both the information domain and the generation format. TAPT has already been shown effective to include domain and task specificity (GURURANGAN et al., 2020).

The continuous pretraining of the LM requires feeding the model with task data. This step is needed since using a pretrained LM *as-is* demands more effort in prompt engineering to create high-quality sentences, aspects, and opinions. To illustrate this assertion, we perform In-Context Learning with the pretrained Language Model GPT-2 *as-is*, i.e. as it has been made available without performing any additional task as fine-tuning, TAPT or Domain-Adaptative Preraining (DAPT). We tried to generate opinionated tuples from an opinionated review using prompt engineering.

The prompt examples shown in Listing 2 demonstrate that some effort is required to find a prompt configuration that returns the correct opinionated tuples. In the three prompting examples none of them produced what was expected. Notice that the problem of generating opinionated triples is even more complicated than the examples in Listing 2 since, beyond the tuples, the LM must generate a new opinionated sentence.

The TAPT step presets the information domain and the text format we expect the LM

Listing 2 – Prompting examples for exploring a pretrained LM *as-is* to generate opinionated tuples. Boldface highlighted text is fed as a prompt to the LM while the following text is automatically generated.

ZERO-SHOT

PROMPT: Extract opinion from the review.
Review: The bread is amazing.
Opinion: No reason why all our bread lovers should try this recipe. The bread is delicious. It has a nice crunchy texture and taste. Order the recipe Here I get: The bread has a nice crunchy texture and taste. [...]

ONE-SHOT

PROMPT: Extract the opinion of the review.
Review: The bread is amazing.
Opinion: [bread, amazing]
Review: The best food in the city.
Opinion: [food not for the faint of heart.
Opinion: [food for the faint of heart.
Great service, super easy service, fast service, delicious stuff, great quality
Opinion: [food, awesome food, good service
Review: Not bad, not bad, not bad

FEW-SHOT

PROMPT: Extract the aspect and the opinion words of the review.
Review: The bread is amazing.
Aspect: bread
Opinion Words: amazing
Review: I love the croissants.
Aspect: croissants
Opinion Words: love
Review: I really dislike the cake they sell.
Aspect: cake
Opinion Words: dislike
Review: Their coffee is so tasty and strong.
Aspect: coffee
Opinion Words: tasty
Aspect: coffee
Opinion Words: strong
Review: The best coffee in town.
Aspect: coffee
Opinion Words: strong
Review: I love this place!
Aspect: coffee
Opinion Words: great

Source: Created by the author

Listing 3 – Fragment of text fed to the LM for TAPT

```
...
<startoftext>
Review:  But the staff was so horrible to us.
Opinion: [staff, horrible]
<endoftext>
<startoftext>
Review:  The design and atmosphere is just as good.
Opinion: [design, good]
Opinion: [atmosphere, good]
<endoftext>
...
```

Source: Created by the author

to generate. As previously stated, the fine-tuning strategy adapts the LM to supervised tasks while TAPT is the continuous pretraining of the LM using task-specific unlabeled data. Since we are dealing with a generative LM, we use a few labeled examples to build the unlabeled text required to pretrain the LM.

To generate opinionated triples for TOWE and AOPE tasks, we condition the LM to generate a new opinionated sentence and the tuples of aspects and opinion words mentioned in the sentence. The foundation of this step is existing opinionated triples on the required target domain, e.g. *restaurants* or *electronics*. While performing TAPT, as shown in Figure 8, our pipeline receives this small set of triples and formats them as shown in Listing 3. This formatted data is fed to the pretrained LM to continue the pretraining. The result is a Task-adapted Pretrained LM with embedded specific information about generation format and information domain.

6.2 IN-CONTEXT LEARNING

After building a task-specific LM, we apply in-context learning approaches to generate new triples. Building the zero-shot prompt is straightforward since it is required only a brief introduction to the task. In our case, the introduction to triple generation is just a "*Review:*" indicator, as shown in Figure 8. Since we perform TAPT on the LM, one expects that the model already understands the generation format: an opinionated review followed by tuples of aspect and opinion words present in the generated review, see Figure 8.

For One-shot and Few-shot, we present hints to the model as examples. For this, we randomly select labeled triples from other datasets on the same domain as the text that should be generated. Another solution might be using triples already generated by the zero-shot prompting. However, we decided to use a more conservative approach since the triples synthetically generated can introduce noise to the other prompt formats.

In the last step, we process all the triples generated by the In-Context Learning approaches. Mainly, we check if the generated tuple is true, i.e., the generated words for aspect and opinion elements are present in the generated review. After the number of required triples is achieved, we remove the duplicates. Finally, the synthetic triples can be used to augment an existing training dataset or create a new one.

6.3 POST-PROCESSING

Listing 4 – Heuristics for automatic labeling of generated triples.

```
1.  Get the tokens positions for the aspect in the sentence;
2.  Get the tokens positions for the opinion expression in the sentence;
3.  In case the tokens for any element (aspect or opinion expression)
    appear more than once in the sentence, check if their positions follow
    a sequence:
    If positive, group them;
    If negative, return them separate.
4.  Tag the sentence according to the approach.
```

Source: Created by the author

During text generation, we save only those obtained from the text generated with the expected format as actual triples. For this, we use regex to detect the sentence generated after the field *"Review:"* and extract the tuples inside brackets after the field *"Opinion:"*. At last, we verify whether the words assigned as aspect or opinion are present in the generated review. After filtering the synthetic triples, the final post-processing step assigns labels for each token in the generated sentence. It performs this labeling step considering the particularities of each TOWE and AOPE algorithm data format used for evaluation.

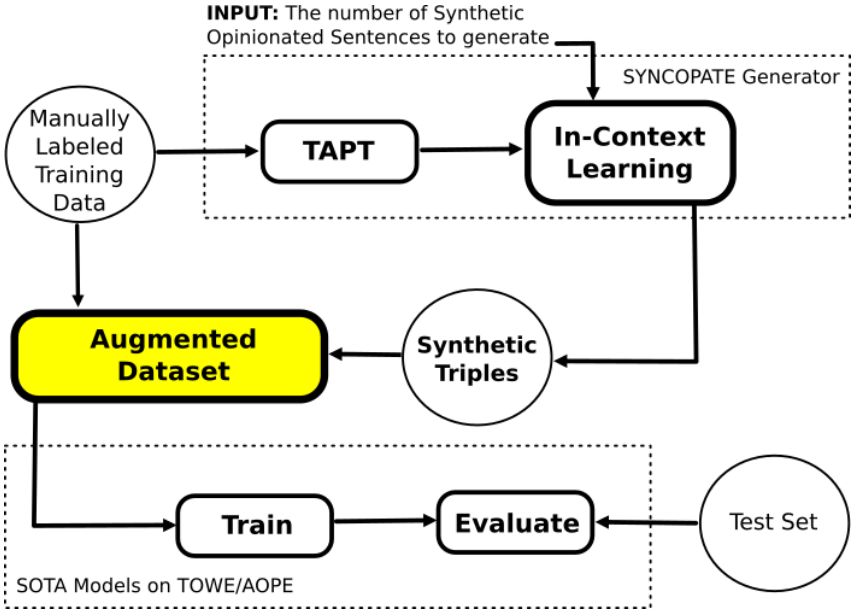
Since the generated triples do not contain any labeling information, we have to accordingly assign them before feeding the augmented data to each evaluated model. Due to the large number of generated samples required for evaluation and the effort of labeling them

manually, we have defined some heuristics to do it automatically. With these heuristics, we have tried to faithfully follow the original labeling of the evaluated methods mentioned in section 7.1.2, which was done manually. Also, for fair comparisons, we have assigned the same automatic labeling to the original sets as well as the generated samples. The heuristics are given in Listing 4.

7 SYNCOPATE EVALUATION

This chapter presents the experiments performed to evaluate our SYNCOPATE approach. Figure 9 depicts the steps required to build the augmented datasets, train the State-of-the-Art (SOTA) models on both evaluated tasks, and evaluate the performance of these models. Below we describe the setups to perform each step and finish the chapter with an analysis of the models' performance when trained with data augmented with synthetic opinionated triples. First, in Section 7.1 we give the experimental setup describing the benchmark datasets we use to adapt the Language Model, the state-of-the-art approaches we use to verify enhancement in the training performance, and a hyperparameter settings statement. Section 7.2 presents the synthetic triples we generate to augment the benchmark datasets, and at last in Section 7.3 we discuss the results of our experimental evaluations.

Figure 9 – The steps performed to generate the synthetic triples using the SYNCOPATE framework and evaluate them on SOTA models for the TOWE and AOPE tasks.



Source: Created by the author

7.1 SETUP

7.1.1 Datasets

We have opted to evaluate the SYNCOPATE pipeline for opinionated triples generation using benchmark datasets, and SOTA works on both Target-Oriented Opinion Word Extraction (TOWE) and Aspect-Opinion Pair Extraction (AOPE) tasks. Mostly because benchmark datasets are well-known for the tasks and because they are manually curated. Hence, they will not introduce noise during Language Model adaptation, which is not the case for the automatically labeled data we built using PGOpi for *Amazon* and *BestBuy* opinion-target mapping. Therefore, to evaluate the quality of SYNCOPATE, we conduct experiments with benchmark datasets initially created for Aspect-Based Sentiment Analysis (ABSA) tasks. These datasets were collected, labeled, and curated by (PONTIKI et al., 2014; PONTIKI et al., 2015; PONTIKI et al., 2016) with data related to *Laptops* and *Restaurants*. Since these datasets originally do not present labels to the aspect-opinion relation, Fan et al. (2019) have included this labeling to perform TOWE. Table 11 shows some statistics of the datasets.

Table 11 – Statistics of the original Datasets. The number of unique triples, aspects, opinions and pairs of aspect-opinion. The same as (FAN et al., 2019). Triples: sentence, aspect, and opinion. Pairs: aspect and opinion.

	ELEMENTS	**14lap**	**14res**	**15res**	**16res**
TRAIN	Triplas	1634	2643	1076	1512
	Aspects	759	954	470	641
	Opinions	802	1073	534	719
	Pairs	1473	2248	952	1327
TEST	Triplas	482	864	436	457
	Aspects	312	425	236	237
	Opinions	297	410	265	237
	Pairs	464	768	418	429

Source: Created by the author

7.1.2 Approaches

We investigate the impact of the synthetically generated triples in four state-of-the-art approaches, two for the TOWE task and two for the AOPE task. The investigated approaches are the ones listed below:

IOG IO-LSTM + Global Context, proposed by (FAN et al., 2019) addresses TOWE as a sequence labeling task. It is a model based on an Encoder-Decoder architecture. The Encoder consists of a target-fused approach that incorporates the left and right contexts of the target into a context representation. To model the left and right context, the authors apply an Inward-Outward Long Short-Term Memory (LSTM) (IO-LSTM) and combine its outputs to a global context. The Decoder receives the Encoder output and performs the sequence labeling task. The authors formulate a greedy decoding as a three-class classification problem to assign a label for each position in the output. The authors assign the BIO schema for tagging the labels in the sentence separately for aspect and opinion expression.

TSMSA Target-Specified sequence labeling with Multi-head Self-Attention is developed by (FENG et al., 2021b) and addresses the TOWE task by identifying the aspect with the [SEP] marker and retrieving its representation from the multi-head self-attention mechanism of the pretrained Language Model Bidirectional Encoder Representation from Transformer (BERT). The BIO scheme is applied for labeling opinions. A projection and a Conditional Random Fields (CRF) layers are responsible for the sequence labeling in the output.

MT-TSMSA It is the Multi-Task version of Target-Specified Sequence Labeling with Multi-head Self-Attention (TSMSA) (FENG et al., 2021b). It merges both the TOWE and AOPE tasks into a Multi-task learning approach to evaluate the joint extraction of aspect and opinions. The labeling for the TOWE task follows the same strategy previously described, while for AOPE the BIO-ASP and BIO-OP are used for sequence labeling.

SDRN Synchronous Double-channel Recurrent Network, proposed by (CHEN et al., 2020) consists of an encoding layer based on BERT to learn context representations, an opinion entity extraction unit and a relation detection unit built as double channels to extract aspects, opinion expressions and relations, simultaneously. The authors apply a BIO-P

tagging scheme for labeling opinions and a BIO-T for aspects. The model also requires a separate label with the indexes of the tokens holding a relation in the sentence, this labeling is identified by the term *#Relation*.

7.1.3 Hyper-parameter Settings

In this study, we have opted to use the GPT-2 Language Model (LM) (BROWN et al., 2020). To perform Task-Adaptative Pretraining (TAPT) we have defined the number of *epochs* in 3, and $max\_length = 400$ for the text fed to the LM. We do not perform any hyper-parameter tunning for training TOWE and AOPE approaches. Since we evaluated state-of-the-art models running over benchmark datasets, and the autonomously generated samples are in the same domain as the benchmark datasets, we have applied the same hyper-parameters found by the methods mentioned in the previous section.

7.2 GENERATED SYNTHETIC TRIPLES

We set the number of unique opinionated synthetic sentences to 10 (ten) thousand for each scenario. Since a unique opinionated sentence can present more than one opinion about one or more aspects mentioned, the number of triples is expected to be larger than unique sentences. We set the few-shot setting to build the prompt with 5 (five) random sentences selected from other datasets in the same domain, as described in Section 6.2. For example, when generating triples for the `14res` domain, we randomly select triples from `15res` or `16res` to build the prompt. Notice that the test sets can not be used for this step to avoid bias towards it. Since only one dataset relative to the *Laptops* domain (`14lap`) is labeled and largely used for TOWE and AOPE tasks, we manually built a new set of 201 triples by labeling aspect-opinion relations in 128 sentences selected from other SemEval datasets (`15lap` and `16lap`) (PONTIKI et al., 2015; PONTIKI et al., 2016) in the Laptops domain that was not yet used for TOWE and AOPE.

Table 12 shows the number of synthetic triples built using the pre-trained LM aligned to TAPT. IOG and TSMSA refer to TOWE methods while SDRN and MT-TSMSA refer to AOPE methods. For comparison purposes, we also show the statistics for the original datasets after running the same heuristics for automatic labeling applied to the synthetic samples.

Table 12 – Statistics of the generated synthetic raw triples and the final number of labeled samples (triples and pairs) for each state-of-the-arte model, prompt setup and original datasets after removing duplicates and performing post-processing (automatic labeling). IOG and TSMSA are approaches for TOWE. SDRN and MT-TSMSA are applied to the AOPE task.

			TRIPLES			
		PROMPT TYPE	**14lap**	**14res**	**15res**	**16res**
SYNTHETIC RAW TRIPLES		ZERO	12682	14655	13473	13450
		ONE	10382	11286	11276	10706
		FEW	10497	10673	10532	11043
AFTER POST-PROCESSING	IOG	ORIG.	1634	2643	1076	1512
		ZERO	11675	13978	12633	12657
		ONE	8706	10200	10515	9973
		FEW	9150	9828	9949	10293
	SDRN	ORIG.	1634	2643	1076	1512
		ZERO	11675	13978	12633	12657
		ONE	8706	10200	10515	9973
		FEW	9150	9828	9949	10293
	(MT) -TSMSA	ORIG.	1634	2643	1076	1512
		ZERO	11619	13936	12599	12618
		ONE	8674	10159	10493	9954
		FEW	9142	9818	9930	10275

Source: Created by the author

Regarding the automatic annotation (Listing 4), there was a small difference between the annotation for TSMSA and MT-TSMSA, where sentences without any aspect-opinion pair were not labeled. Hence, the annotation error $\left(AE = \frac{annotation\ errors}{filtered\ triples}\right)$ for these models was around 3.84% while for the other two models was 3.59%. This small difference between the labeled data does not harm the analysis since our aim is not to benchmark models but verify their performance when fed autonomously generated training samples.

As expected, the number of triples generated by each prompt type was different. The zero-shot prompting has created more triples than One-shot and Few-shot prompts. This happened because the two last prompt formats passed to the LM have too much specificity. As shown by (ZHAO et al., 2021), the given prompt, especially by the Few-shot strategy, can bias the text generation to the words fed to the prompt. Hence, a prompt formed by opinionated sentences with only one opinionated tuple (aspect and opinion expression) might bias the LM to generate an opinionated sentence with only one tuple.

After the raw generation of triples in the expected format, the post-processing step removed possible duplicated triples and built the labeled samples to feed the TOWE and AOPE methods. On average, the Duplicity Ratio $\left(DR = \frac{duplicates}{generated}\right)$ of all generated datasets is around 4.66%. The average Annotation Error $\left(AE = \frac{annotation\ errors}{filtered\ triples}\right)$, computed after the automatic labeling, of all datasets for all evaluated methods is around 3.67%.

7.3 RESULTS

We evaluate the augmented datasets by experimenting with state-of-the-art methods for TOWE and AOPE in three different scenarios. For this, we split all datasets: the original SemEval datasets, referenced in this section only by "`original`", and the datasets built with synthetic samples. We split these datasets into five subsets: 10%, 25%, 50%, 75%, and 100%. Notice that the percentages relative to the augmented (`AUG`) datasets are related to the percentage of synthetic samples added to the complete original set.

The first scenario evaluates the models' performance while trained only with a few manually built triples. The second scenario analyses the models' performance only with synthetic data. Finally, the third scenario analyses the performance while training the models with augmented data (the complete original datasets augmented with the synthetic triples). The results of this analysis are shown in Table 24 in Appendix B for the TOWE task and in Table 25 in Appendix D for the AOPE task. We comment on each of these results in the next sections.

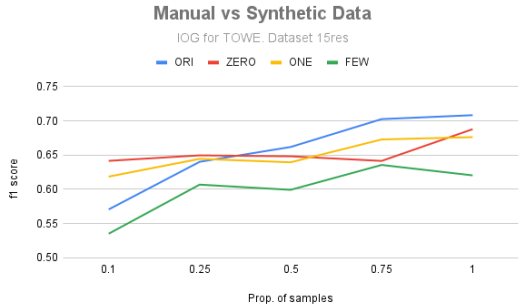
7.3.1 TOWE

Table 24 in Appendix B presents all the results for the performance of the SOTA models in the TOWE task when trained with the analyzed datasets. Here we highlight the main insights for each model and dataset due to the large number of scenarios to evaluate.

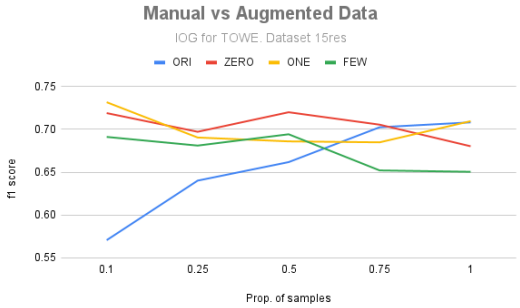
Although feeding the models only with synthetic samples (`GEN`) does not surpass the results obtained by the original datasets (`ORG`), we notice in the zero-shot setting that using at least 10% of the synthetic samples alone is equivalent to using 25% of manually built triples, see in Table B the performance of `IOG` in the `15res` dataset - `ORG`: $f1(25\%) = 0.64$,

Figure 10 – Evaluation scenarios for the performance of the IOG model when trained on variations of the `15res` dataset: Manually built data (ORI), and the Synthetic Triples built by SYNCOPATE's Zero-shot paradigm (ZERO), One-shot paradigm (ONE), and Few-shot paradigm (FEW).

(a) Manual *versus* Synthetic



(b) Manual *versus* Augmented



Source: Created by the author

`GEN`: $f1(10\%) = 0.64$. This insight is also illustrated in Figure 10(a). The same behavior is observed when training `IOG` with the other three synthetic datasets (`14lap`, `14res`, and `16res`) and training `TSMSA` with `14lap` and `14res`. However, this is not observed for `TSMSA` in the `16res` dataset - `ORG`: $f1(10\%) = 0.78$, `GEN`: $f1(50\%) = 0.77$ - where the best performance for `GEN` using 50% of synthetic triples does not even reach the performance of using 10% of manually built triples. All the other comparison scenarios on training the models with Manually built data, with only Synthetic data, and with Augmented data are illustrated in Appendix C for all the evaluated models in the TOWE task.

Nevertheless, the results show that augmenting (`AUG`) the original data with synthetic triples generated by our approach in the zero-shot setting can improve the performance of the models for the task, see Figure 10(b) and the other scenarios in Appendix C. The poor performance with synthetic triples generated by the one-shot and few-shot settings is probably due to the bias inherited from the random examples selected to build the prompts. Furthermore, the results indicate that the model's performance decreases when the percentage of synthetic triples increases, which indicates that more noise is introduced, hurting the model's performance.

We perform hypothesis testing to verify whether there is a statistically significant difference in the performance of the models when trained with these different datasets. We restrict the evaluation scenarios to the augmented data with 10% of synthetic samples generated by the zero-shot setting against full (100%) original data.

We performed the Student's t-test one-tailed over ten runs for each model trained with

Table 13 – F1 score average of ten runs of the performance of the models in the TOWE task. Comparison between training the models only with manually labeled data *versus* training the model with the same dataset augmented with 10% of autonomously generated opinionated triples.

AVERAGE OF 10 RUNS (F1-SCORE)	IOG		TSMSA	
	ORI	**AUG**	**ORI**	**AUG**
14lap	0.6895	**0.7011**	**0.7922**	0.7884
14res	**0.7722**	0.7648	**0.8610**	0.8506
15res	0.7077	**0.7141**	**0.8045**	0.7988
16res	0.8056	0.8051	**0.8821**	0.8787

Source: Created by the author

the original dataset against the dataset augmented with 10% of synthetic samples. The results for the ten runs average are shown in Table 13. The null hypothesis is "*H0: There is no difference in the f1-scores mean of the models trained with the different training sets.*" and the alternative hypothesis is "*H1: The f1-scores mean of the model trained only with manually built triples is smaller than training the model with augmented training set.* The results of the one-tailed t-test are shown in Table 14. The one-tailed t-test showed that at a significance level of 0.05 ($\alpha = 0.05$) we reject the null hypothesis to the model `IOG` trained with the `14lap` datasets ($t - value = -2.1234$ and $p - value = 0.023921$). Hence, training `IOG` with the augmented `14lap` dataset (mean F1 score = 0.701) has surpassed the performance of the same model trained with the original training set (mean F1 score = 0.690). The test also rejected the null hypothesis for training the `IOG` model with datasets from the `14res` domain ($t - value = 1.8506$; $p - value = 0.040355$; mean F1 score on $original = 0.772$ and $augmented = 0.765$); and the test also rejected *H0* for `TSMSA` trained with `14res` ($t - value = 5.4074$; $p - value = 0.000019$; mean F1 score on $original = 0.861$ and $augmented = 0.851$). This indicates that in these two last scenarios, the performance of the models was decreased by including synthetic triples to the original dataset. For all the other cases, we could not reject the null hypothesis, which indicates that the synthetic triples do not improve the results for TOWE. However, it also does not significantly hurt the performance of the model. Comparing the results of the zero-shot to the one-shot and few-shot settings, in most cases, the performance of the models seems to drop as the number of hints given to the prompt increases. Additionally, the `IOG` model seems more sensitive to noisy data than `TSMSA`.

Table 14 – Results of the one-tailed t-test hypothesis testing of the models' performance for the TOWE task. The blue boldface result is the scenario where the synthetically built opinionated triples have improved the models' performance. Red boldface otherwise.

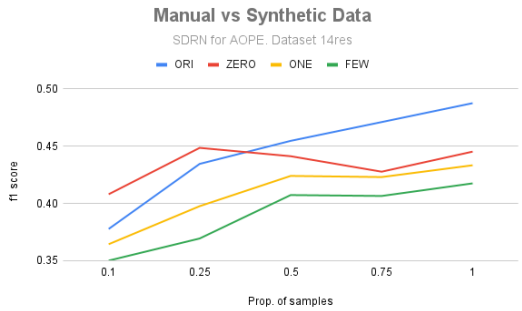
$\alpha = 0.05$	IOG		TSMSA	
	p-value	**t-value**	**p-value**	**t-value**
14lap	**0.023921**	**-2.1234**	0.209007	0.82891
14res	**0.040355**	**1.8506**	**0.000019**	**5.4074**
15res	0.142833	-1.10038	0.057638	1.6549
16res	0.460534	0.10049	0.203906	0.84756

Source: Created by the author

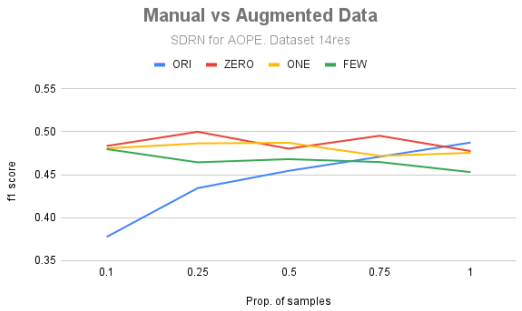
7.3.2 AOPE

Figure 11 – Evaluation scenarios for the performance of the SDRN model when trained on variations of the `14res` dataset: Manually built data (ORI), and the Synthetic Triples built by SYNCOPATE's Zero-shot paradigm (ZERO), One-shot paradigm (ONE), and Few-shot paradigm (FEW).

(a) Manual *versus* Synthetic



(b) Manual *versus* Augmented



Source: Created by the author

Table 25 in Appendix D presents all the results for the performance of the SOTA models in the AOPE task when trained with the analyzed datasets. The performance results for the models in AOPE reinforce the previous analysis that the zero-shot setting allows for building better triples, see Figure 11. The figure also shows that training the `SDRN` model only with 10% of synthetic triples, Figure 11(a), surpasses the performance of training the model with 10% of the manually built triples, which is also observed while using 25% of both datasets during training. The same does not occur to the `MT-TSMSA` model (The visualization for this and other scenarios are shown in Appendix E). When training both models with the augmented training data, it surpasses the performance of

Table 15 – F1 score average of ten runs of the performance of the models in the AOPE task. Comparison between training the models only with manually labeled data versus training the model with the same dataset augmented with 10% of autonomously generated opinionated triples.

AVERAGE OF 10 RUNS	SDRN		MT-TSMSA	
	ORI	AUG	ORI	AUG
14lap	0.5126	**0.5195**	**0.5974**	0.5903
14res	0.4887	**0.4903**	**0.5703**	0.5653
15res	0.5254	**0.5255**	0.5940	**0.6042**
16res	**0.5892**	0.5868	0.6450	**0.6630**

Source: Created by the author

Table 16 – Results of the one-tailed t-test hypothesis testing of the models' performance for the TOWE task. The blue boldface results are the scenarios where the synthetically built opinionated triples have improved the models' performance. Red boldface otherwise.

$\alpha = 0.05$	SDRN		MT-TSMSA	
	p-value	t-value	p-value	t-value
14lap	**0.040038**	**-1.85483**	**0.038018**	**1.88249**
14res	0.358673	-0.36775	0.366577	0.34627
15res	0.490419	-0.02435	0.13698	-1.12842
16res	0.29378	0.55227	**0.031654**	**-1.97911**

Source: Created by the author

the original training data only in the `14res`, verify in Figure 11(b), and `16res` scenarios (Appendix E). The same is true for the `15res` scenario of the `MT-TSMSA` model.

In general, the augmented training data performs better when training the `MT-TSMSA` model. This is probably due to the Multi-Task approach, which considers different labeling strategies for each task and merges them to get the relationship between aspect and opinion expressions. Hence, the noisy labels of synthetic triples have a significant influence while training the models only with the generated samples. However, their influence is reduced when augmenting the original training set. Regarding the `SDRN` method, it seems more sensible to the noise in training data than `MT-TSMSA`.

We also perform the Student's t-test one-tailed over ten runs for each model trained with original datasets against the training set augmented with 10% of synthetic triples. The results for the ten runs average performance in these two case scenarios are shown in Table 15. We considered the same null and alternative hypotheses previously stated. The results of the one-tailed t-test are shown in Table 16. Also, at a significance level of

0.05 ($\alpha = 0.05$) the test rejected the null hypothesis to the model MT-TSMSA trained with the 16res datasets ($t - value = -1.97911$ and $p - value = 0.031654$). This indicates that the model trained with the augmented dataset (mean F1 score =0.663) shows superior performance over the model trained with the original training set (mean F1 score =0.645). The test also rejected the null hypothesis for MT-TSMSA in the 14lap scenario where the augmented training set decreased the performance of the model (mean F1 score on the original set = 0.597 and the augmented one = 0.590). Regarding the SDRN model, the augmented dataset significantly improved the performance of the model for 14lap (mean F1 score on the original set = 0.513 and the augmented one = 0.520). The other evaluated scenarios do not show a significant difference in performance when training the model with the augmented datasets.

Table 17 – Samples of sentences rated as non-opinionated by manual raters. The first block contains all sentences classified as non-opinionated with a total agreement between raters. The second block presents some sentences with evaluation divergence. The agreement proportion is given between parentheses.

FULL AGREEMENT
1. You can see the monitor and keyboard on it, right above the keyboard, and below the keyboard.
2. After ordering and had the computer replaced with an inoperable thermal pad, the computer completely failed to power on.
3. Then I realized that it's not exactly a power cord.
DISAGREEMENT BETWEEN RATERS
1. The price is even lower. (1/3)
2. I just had to have it replaced and it runs flawlessly now! (2/3)
3. I have had it for about 3 months now and have been very happy with it. (1/3)
4. After eating this sandwich, I ordered the entree and enjoyed all the portions. (2/3)
5. The service was prompt too, and the portions were pretty much the same. (1/3)
6. There is a large area for food, drinks, and service. (1/3)
7. There is the most standard operating system we can get from my computer. (2/3)
8. A few things I highly recommend to anyone, including those visiting the restaurant. (2/3)
9. The battery life in this laptop is long after it last in a notebook. (1/3)

Source: Created by the author

7.3.3 Human Evaluation of Generated Triples

We asked three manual raters to analyze the quality of the triples generated by the zero-shot setting. Due to a large number of generated triples, we randomly selected 25 unique opinionated sentences from each dataset domain (`14lap`, `14res`, `15res`, and `16res`), resulting in 161 triples. All three manual rates evaluated each triple. Details on the recruitment and instructions given to the raters are given in Appendix F.

We asked four questions about each generated triple: *Q1*) Is the sentence opinionated? *Q2*) Is the extracted aspect an aspect of a service or a product? *Q3*) Do the extracted opinion terms represent an opinion or sentiment? *Q4*) Are the extracted aspect and opinion terms related?

Since the answers to those questions are in the "yes/no" type, we use Fleiss' Kappa for assessing the reliability of agreement between the raters (k) (FLEISS; LEVIN; PAIK, 2003). In general, there was a moderate level of agreement between the raters for all asked questions: *Q1*) $k = 0.41$; *Q2*) $k = 0.32$; *Q3*) $k = 0.40$; *Q4*) $k = 0.47$.

According to the evaluation, 92%, on average, of the generated sentences are opinionated. Although at least 21 sentences were rated as non-opinionated, there was disagreement between the raters. Only three sentences have presented full agreement. Table 17 presents some samples of these sentences evaluated as non-opinionated. Additionally, the raters agreed that around 81% of extracted aspects from opinionated sentences are real aspects, 80% of the extracted opinion terms are opinion or sentiment words, and around 69% of extracted aspect-opinion pairs are related.

Given the difficulty of the task, the results obtained from the human evaluation, and the moderate agreement between the three raters, we notice that a considerable number of generated triples have good quality. However, the quality decreases when trying to mimic the relations between aspects and opinion expressions, mainly because of the difficulty in handling opinionated sentences with multiple opinion expressions and aspects, as shown in Table 18.

Table 18 – Samples of triples evaluated by manual raters as containing an opinionated sentence. Examples of triples with disagreement between raters are also presented. The agreement proportion is given between parentheses. ♣ indicates disagreement regarding the generated aspect. ♠ indicates disagreement on the generated opinion words. ♢ indicates disagreement regarding the relationship between aspect and opinion words.

Review:	the price (in my opinion) looks good for what I need it, the design is stunning, and the netbook just looks awesome.
Opinion:	[price, good]
Opinion:	[design, stunning]
Review:	The only regret is the delivery.
Opinion:	[delivery, regret]
Review:	The service was poor, we ate at a second-hand bistro with decent food except for some pizza.
Opinion:	[service, poor]
Opinion:	[food, decent]
Opinion:	[pizza, decent] ♢
Review:	I liked the wait for food and service but their service is not as impressive.
Opinion:	[wait, liked] ♣ (2/3) ♢ (2/3)
Opinion:	[service, not as impressive]
Review:	The service was very nice as well and did a great disservice to the hostess and staff.
Opinion:	[service, nice]
Opinion:	[staff, did great disservice] ♣ (1/3) ♠ ♢
Review:	The ambience is average, it's not that different from Brooklyn... but the staff is so overtrained that it is difficult for reasonable people to maintain good relationships with the music, music, service, and decor.
Opinion:	[ambience, average] ♠(1/3)
Opinion:	[staff, overtrained]
Opinion:	[music, difficult] ♠ (1/3) ♢

Opinion:	[decor, difficult] ♠ (1/3) ♢
Review:	They offer some very attractive features like easy to use, built-in video player.
Opinion:	[features, attractive] ♣ (1/3) ♢ (1/3)
Opinion:	[built-in video player, attractive]
Opinion:	[video player, attractive] ♣ (1/3) ♢ (1/3)
Review:	The service is great, prices are extremely reasonable.
Opinion:	[service, great]
Opinion:	[prices, reasonable]
Review:	The pizza was good and the staff was friendly.
Opinion:	[pizza, good]
Opinion:	[staff, friendly]
Review:	The ambience in this nice little town has a nice flair to it, the service is prompt, it is great.
Opinion:	[ambience, nice] ♢ (1/3)
Opinion:	[service, prompt]

Source: Created by the author

8 CONCLUSIONS AND FUTURE WORK

In this thesis, we propose a method devised to enrich Product Graph (PG) with opinions extracted from user reviews. Currently, our work is based on a traditional unsupervised method for opinion extraction and on a deep learning architecture that generates models to help in the task of mapping opinions to nodes in the PG. Nodes that correspond to the targets of these opinions.

To answer the first research question *RQ1: Is it feasible to enhance Product Knowledge Graphs with subjective information extracted from user reviews without relying on manually labeled training data?* we reviewed State-of-the-Art (SOTA) works on PG, Aspect-Based Sentiment Analysis (ABSA), and also listed works that already embrace the problem of organizing subjective information. We also formalized the PG structure to include this subjective information and explored an unsupervised approach to extract pairs of aspects and opinion words from customers' written product reviews.

We answer the second research question *RQ2: Can we map the pairs of opinion words and opinion targets extracted from users' reviews to product targets in Product Knowledge Graphs?* by building the weak-supervised pipeline PGOpi that relies on a distant supervision strategy based on word embeddings to map opinions extracted from opinionated reviews to targets in a PG. We have validated the pipeline performance with ground-truth manually labeled data from two real-world datasets obtained from large-scale e-commerce platforms: Amazon, and BestBuy. We also compared the performance of our weak-supervised approach against a fully supervised SOTA work.

The PGOpi pipeline compared to a SOTA fully supervised pipeline has shown superior performance in 4 out of 10 scenarios and equal performance in another one. Hence, the results have attested that the PGOpi pipeline presents a competitive performance to supervised approaches without relying on the manual labeling of training data. Also, our Opinion-Target Classifier for mapping extracted opinions to the PG has surpassed the performance of more complex deep learning architectures. These differences were attested by hypothesis testing using the Friedman Test followed by a Nemenyi post-hoc test at a significance level of 0.05. Additionally, the Distant Supervision approach for assigning automatic labels depends on selecting a suitable similarity threshold in order to build training datasets without too much noise.

Regarding research question number three *RQ3: Can we explore data augmentation, by automatically building training examples, to improve the Aspect-Opinion Pair Extraction task?* we investigate an approach based on the adaptation of a pre-trained Language Model. We perform In-Context Learning to build synthetic opinionated triples and to extract tuples of aspect and opinion words. This investigation has shown that our generation's approach is promising for building good quality opinionated triples, and consequently can be applied to extract the opinionated tuples from a given opinionated sentence, working as an aspect-opinion extraction tool depending only on a few manually built samples without prompting engineering.

To build good quality opinionated triples, we only rely on a few manually built triples and the In-Context Learning of a pre-trained Language Model. We use these few initial seeds to continue the pre-training of the Language Model so it can specialize in the required information domain and expected format. Using zero-shot, one-shot, and few-shot approaches, we could build synthetic opinionated triples that can be used to enhance existing models or even use the pre-trained Language Model to perform prompting extraction of aspects and opinion words from a given opinionated sentence.

We evaluate the autonomously generated synthetic opinionated triples by using them to train four SOTA works on two aspect-opinion pair extraction tasks. The obtained results have shown that using only 25% of the built synthetic triples to train the models has improved the generalization capacity of the models in most cases when compared to using the same proportion of manually labeled data. The results also show that using the manually built data augmented with synthetic triples has enhanced the performance of the models in 3 out of 16 scenarios according to the hypothesis test Student's t-test at a significance level of 0.05. Also, the performed the hypothesis testing has shown that the performance of the models has not been hurt in 10 out of 16 scenarios which indicate that the generated opinionated triples are well-formed.

Additionally, we asked three human raters to attest to the quality of these Synthetic Triples by asking them four questions related to the elements of the triple. 92% of the answers have attested that the generated sentences were opinionated, 81% agreed that the generated aspect terms were really characteristics of a product or service, 80% attested that the opinion expressions really express sentiment or opinion, and 69% agreed that the generated aspect and opinion terms are related in the generated opinionated sentence.

Hence, we can conclude that the work here presented has solved all the research

questions previously raised. Also, both pipelines here presented are complementary and have contributed to the Knowledge Graphs field by allowing the augmentation of these structures with subjective information without relying on a lot of manual effort to build labeled data.

8.1 CONTRIBUTIONS

Despite the broad interest in product graphs (DONG, 2018; XU et al., 2020; KIM, 2017) and in aggregating subjective information to structured information (HALEVY, 2019; KOBREN et al., 2019; LI et al., 2019; MELO et al., 2019), to the best of our knowledge, the problem of enriching PG with subjective information, has not been proposed before. This may yield the development of a series of new methods that can take advantage of this new kind of graph to improve e-commerce-related methods such as recommendation, searching, comparison, pricing, etc. In addition, it may also influence the proposal of knowledge graphs in other domains (e.g., health or law) that also include subjective information.

From the practical point of view, our proposals for graph representation, deep learning model and architecture, and distant supervision strategy are fairly adequate for industrial implementation. Importantly, they rely on resources that are readily available in most e-commerce websites and services, that is, a product graph, or at least a product catalog, and user-written reviews. Indeed, all of our experiments used data collected from real-world e-commerce websites, which evidences the applicability of our proposals. The code and data for **P**roduct **G**raph enriched with **Opi**nions (PGOpi) is available at <https://github.com/guardiaum/PGOpi>.

Additionally, we could not find any other work trying to build synthetic opinionated triples to improve the pair extraction of aspects and opinion expressions. The proposed Synthetic Opinionated Triples generation framework shows promising results augmenting the performance of Target-Oriented Opinion Word Extraction (TOWE) and Aspect-Opinion Pair Extraction (AOPE) tasks, and can be further integrated into our PGOpi pipeline to extract aspect-opinion pairs. The experimental evaluation results show that the generation framework is effective in generating good-quality opinionated triples. Although it has not improved the tasks in all evaluated scenarios, the generated triplets have not significantly hurt the models' performance. It is crucial to notice that our pipeline is straightforward and does not focus on complex layers of models or filters.

The code and data for **SYN**theti**C** **OP**inion**A**ted **T**ripl**E**s (SYNCOPATE) is available at <https://github.com/guardiaum/SYNCOPATE>.

Given the points before, we list below our current contributions:

1. We introduce the problem of enriching product graphs with user opinions from product reviews. We proposed a new representation of product graphs so that they can be enriched over time with subjective information taken from user opinions;

2. We propose a method named PGOpi that is very effective for the task of mapping opinions extracted from a set of reviews to the nodes of a product graph, creating an enriched product graph;

3. We describe a distant supervision strategy to automatically generate a representative set of training instances in order to ease the labor of manually annotating sentences for generating training data;

4. We present experimental results on real-world datasets that demonstrate the effectiveness of PGOpi by beating a competitive baseline and other attempts in micro F-score on two representative datasets collected from real online retail stores.

5. We devise a framework called **Syn**theti**c** **Op**inion**a**ted **T**ripl**e**s (SYNCOPATE) that has proven to be effective in build good opinionated triples and can be further used as an aspect-opinion pair extractor;

6. We explore In-Context Learning to generate Synthetic Opinionated Triples without relying on prompt engineering or parameters tunning;

7. We present experimental results on benchmark datasets to demonstrate the quality of the generated opinionated triples on SOTA models for two tasks under the ABSA technique.

8.2 LIMITATIONS OF THE CURRENT APPROACH

Currently, the proposed pipeline relies mainly on rules for unsupervised opinion extraction and semantic similarity for distantly supervised opinion-target mapping. Although Poria et al. (2014) is one of the most consolidated approaches for unsupervised opinion mining, the external information used for building opinion lexicons and semantics is

not sufficient at scale and while dealing with an inconsistent, open, and vast corpus as real users' reviews. Real users' reviews on products are marked by colloquialisms and misspellings, which can generate too much noise over the extracted opinions or even decrease their performance. Although we have also proposed a Synthetic Opinionated Triples (SYNCOPATE) generator that can replace the Poria et al. (2014)'s approach, we could not manage to replace it and perform new experiments to validate the modification.

Another point of discussion in the present PGOpi pipeline is the similarity function used for distantly label training instances for the opinion-target classifier. Even though it is an important step for building labels and allowing the training of the mapping classifier, fitting the parameter ϵ can be exhausting. Although the performed experiments show that most times the differences in performance in the analyzed thresholds can be small, there are cases where a more strict (or soft) threshold is recommended.

Also, regarding the SYNCOPATE generator, the framework is limited in the sense that it still requires a small set of labeled examples to continue the pretraining of the model. Furthermore, the automatic labeling step performed by heuristics can present inconsistencies and introduce noise to the evaluated models. Additionally, we notice difficulties in the approach to handling large opinionated sentences containing multiple opinion expressions and aspects. At last, the proposed pipeline does not filter noisy triples, which can be done by using another classifier as a generation supervisor.

8.3 FUTURE WORK

The main modification to be performed on future work is to replace the traditional method of Poria et al. (2014) for opinion extraction by the SYNCOPATE approach. Additionally, due to this modification new experiments must be employed to investigate the performance of the pipeline as a whole and the aspect-opinion pair extraction specifically. Beyond that, we intend to investigate the use of Transformers or even In-Context Learning to perform the mapping between extracted Opinion Tuples and Targets in the PG. This improvement of the PGOpi pipeline will allow the removal of the dependency on the distant label approach and consequently on the threshold selection. Regarding the tokens labeling and filtering of noisy triples in SYNCOPATE, we intend to improve the pipeline by including Deep Learning approaches for this. A new evaluation must be performed with the SYNCOPATE approach to validate its performance only on the aspect-opinion

pairs extraction task, ignoring the generation of synthetic sentences. It is also necessary to evaluate the integration of the PGOpi pipeline into end applications such as the ones for recommendation tasks, decision-making and searching. This evaluation is required to validate the solution's applicability and the usefulness of the augmented subjective information.

8.4 PUBLICATIONS

We list below the works directly or indirectly related to this thesis. We also list the contributions we made to other authors. The works were all published within the last four years during the development of the doctorate.

Journals

1. **Moreira, J.**, de Melo, T., **Barbosa, L**, da Silva, A. "A distantly supervised approach for enriching product graphs with user opinions". J. of Intelligent Information Systems. 59, 435–454 (2022).
   https://doi.org/10.1007/s10844-022-00717-5

2. **Moreira, J., Barbosa, L.** "DeepEx: A Robust Weak Supervision System for Knowledge Base Augmentation". Journal on Data Semantics (2021).
   https://doi.org/10.1007/s13740-021-00134-x

3. **Moreira, J.**, Costa Neto, E. and **Barbosa, L.** "Analysis of structured data on Wikipedia". Int. J. Metadata Semantics and Ontologies, Vol. 15, No. 1, pp.71–86. (2021) DOI: http://dx.doi.org/10.1504/IJMSO.2021.117108

Conferences

1. Costa Neto, E., **Moreira, J., Barbosa, L.**, Salgado, A. "CoFFee: A Co-occurrence and Frequency-Based Approach to Schema Mining". In Anais do XXXVII Simpósio Brasileiro de Bancos de Dados, (pp. 52-64). Porto Alegre: SBC. (2022)
   doi:10.5753/sbbd.2022.224190

2. **Moreira, J.**, Oliveira, C., Macêdo, D., Zanchettin C. and **Barbosa, L.** "Distantly-Supervised Neural Relation Extraction with Side Information using BERT". 2020 International Joint Conference on Neural Networks (IJCNN), 2020, pp. 1-7, doi: 10.1109/IJCNN48605.2020.9206648.

Workshop

1. **Moreira, J.**, Silva Neto, E. C., **Barbosa, L. A.** "Índices de Infoboxes para Recuperação de Informação Estruturada de Entidades da Wikipédia". In: Brazilian Symposium on Databases, 2019, Fortaleza. 34th Brazilian Symposium on Databases - Dataset Showcase Workshop, 2019.

In Review Articles

Journals

1. Silva Neto, E. C., **Moreira, J.**, **Barbosa, L. A.**, Salgado, A. "Domain-Specific Schema Discovery from General-Purpose Knowledge Base". Int. J. Metadata Semantics and Ontologies.

2. Silva Neto, E. C., **Moreira, J.**, **Barbosa, L. A.**, Salgado A. C. "Toward a Class Schema Discovery for Semi-Structured Data". Journal of Information and Data Management.

Conferences

1. **Moreira, J.**, **Barbosa, L. A.** "In-Context Learning for Data Augmentation in Aspect-Based Opinion Extraction". The 61st Annual Meeting of the Association for Computational Linguistics (2023).

2. (SHORT) Barbosa, J. M., **Moreira, J.**, **Barbosa, L. A.** "Improving Binary Text Classifiers on Imbalanced Data Using Prompt-based Learning". The 61st Annual Meeting of the Association for Computational Linguistics (2023).

REFERENCES

ABONIZIO, H. Q.; JUNIOR, S. B. *Pre-trained Data Augmentation for Text Classification*. Springer International Publishing, 2020. 551–565 p. ISSN 16113349. ISBN 9783030613761. Available at: <http://dx.doi.org/10.1007/978-3-030-61377-8_38>.

ALFONSECA, E.; FILIPPOVA, K.; DELORT, J.-Y.; GARRIDO, G. Pattern Learning for Relation Extraction with a Hierarchical Topic Model. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, n. July, p. 54–59, 2011.

ARCHAK, N.; GHOSE, A.; IPEIROTIS, P. G. Show me the money!: Deriving the pricing power of product features by mining consumer reviews. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 56–65, 2007.

BAHDANAU, D.; CHO, K. H.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, p. 1–15, 2015.

BALOG, K. *Entity-Oriented Search*. Cham: Springer International Publishing, 2018. (The Information Retrieval Series, v. 39). ISBN 978-3-319-93933-9. Available at: <http://link.springer.com/10.1007/978-3-319-93935-3>.

BAYER, M.; KAUFHOLD, M. A.; BUCHHOLD, B.; KELLER, M.; DALLMEYER, J.; REUTER, C. Data augmentation in natural language processing: a novel text generation approach for long and short text classifiers. *International Journal of Machine Learning and Cybernetics*, Springer Berlin Heidelberg, n. 0123456789, 2022. ISSN 1868808X. Available at: <https://doi.org/10.1007/s13042-022-01553-3>.

BEKOULIS, G.; DELEU, J.; DEMEESTER, T.; DEVELDER, C. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, Elsevier Ltd, v. 114, p. 34–45, 2018. ISSN 09574174. Available at: <https://doi.org/10.1016/j.eswa.2018.07.032>.

BERGSTRA, J.; BARDENET, R.; BENGIO, Y.; KÉGL, B. Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, p. 1–9, 2011.

BERGSTRA, J.; YAMINS, D.; COX, D. D. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. *Proceedings of the 30th International Conference on Machine Learning*, v. 28, n. 1, p. 115–123, 2013. ISSN 1545-5882.

BROWN, T. B.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A.; AGARWAL, S.; HERBERT-VOSS, A.; KRUEGER, G.; HENIGHAN, T.; CHILD, R.; RAMESH, A.; ZIEGLER, D. M.; WU, J.; WINTER, C.; HESSE, C.; CHEN, M.; SIGLER, E.; LITWIN, M.; GRAY, S.; CHESS, B.; CLARK, J.; BERNER, C.; MCCANDLISH, S.; RADFORD, A.; SUTSKEVER, I.; AMODEI, D. Language models are few-shot learners. *arXiv*, 2020. ISSN 23318422.

CAMBRIA, E.; OLSHER, D.; RAJAGOPAL, D. SenticNet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis. *Proceedings of the National Conference on Artificial Intelligence*, v. 2, p. 1515–1521, 2014.

CHEN, S.; LIU, J.; WANG, Y.; ZHANG, W.; CHI, Z. Synchronous Double-channel Recurrent Network for Aspect-Opinion Pair Extraction. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020. p. 6515–6524. Available at: <https://www.aclweb.org/anthology/2020.acl-main.582>.

CHENG, J.; ZHAO, S.; ZHANG, J.; KING, I.; ZHANG, X.; WANG, H. Aspect-level Sentiment Classification with HEAT (HiErarchical ATtention) Network. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. New York, NY, USA: ACM, 2017. Part F1318, p. 97–106. ISBN 9781450349185. Available at: <https://dl.acm.org/doi/10.1145/3132847.3133037>.

CHOMSKY, N. Language and Freedom. In: ARNOVE, A. (Ed.). *The Essential Noam Chomsky*. New York, NY, USA: The New Press, 2008. chap. 6, p. 75–91. ISBN 9781595581891.

CLARK, K.; KHANDELWAL, U.; LEVY, O.; MANNING, C. D. What does BERT look at? An analysis of BERT's attention. *arXiv*, 2019. ISSN 23318422.

CRUZ, I.; GELBUKH, A.; SIDOROV, G. Implicit Aspect Indicator Extraction for Aspect-based Opinion Mining. *International Journal of Computational Linguistics and Applications*, v. 5, n. 2, p. 135–152, 2014. ISSN 0976-0962. Available at: <http://www.ijcla.org/2014-2/IJCLA-2014-2-pp-135-152-Implicit.pdf>.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. n. Mlm, 10 2018. Available at: <http://arxiv.org/abs/1810.04805>.

DING, B.; LIU, L.; BING, L.; KRUENGKRAI, C.; NGUYEN, T. H.; JOTY, S.; SI, L.; MIAO, C. DAGA: Data augmentation with a generation approach for low-resource tagging tasks. *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, p. 6045–6057, 2020.

DONG, X. L. Challenges and Innovations in Building a Product Knowledge Graph. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. New York, NY, USA: ACM, 2018. p. 2869–2869. ISBN 9781450355520. Available at: <https://dl.acm.org/doi/10.1145/3219819.3219938>.

FAN, Z.; WU, Z.; DAI, X. Y.; HUANG, S.; CHEN, J. Target-oriented opinion words extraction with target-fused neural sequence labeling. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, v. 1, p. 2509–2518, 2019.

FENG, X.; FENG, X.; QIN, L.; QIN, B.; LIU, T. Language Model as an Annotator: Exploring DialoGPT for Dialogue Summarization. v. 1, 2021. Available at: <http://arxiv.org/abs/2105.12544>.

FENG, Y.; RAO, Y.; TANG, Y.; WANG, N.; LIU, H. Target-specified Sequence Labeling with Multi-head Self-attention for Target-oriented Opinion Words Extraction. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021. p. 1805–1815. Available at: <https://www.aclweb.org/anthology/2021.naacl-main.145>.

FLEISS, J. L.; LEVIN, B.; PAIK, M. C. *Statistical Methods for Rates and Proportions*. Wiley, 2003. (Wiley Series in Probability and Statistics). ISBN 9780471526292. Available at: <https://onlinelibrary.wiley.com/doi/book/10.1002/0471445428>.

GAO, L.; WANG, Y.; LIU, T.; WANG, J.; LIAO, J. Question-Driven Span Labeling Model for Aspect – Opinion Pair Extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, n. 1, 2021.

GRAVES, A.; SCHMIDHUBER, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, v. 18, n. 5-6, p. 602–610, 2005. ISSN 08936080. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S0893608005001206>.

GUO, S.; WANG, Q.; WANG, L.; WANG, B.; GUO, L. Knowledge graph embedding with iterative guidance from soft rules. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, p. 4816–4823, 2018.

GURURANGAN, S.; MARASOVIĆ, A.; SWAYAMDIPTA, S.; LO, K.; BELTAGY, I.; DOWNEY, D.; SMITH, N. A. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In: . [S.l.: s.n.], 2020. p. 8342–8360.

HALEVY, A. The Ubiquity of Subjectivity Subjective data Subjective presentation. *EEE Data Engineering Bulletin*, v. 42, n. 1, p. 6–9, 2019.

HE, R.; LEE, W. S.; NG, H. T.; DAHLMEIER, D. An Interactive Multi-Task Learning Network for End-to-End Aspect-Based Sentiment Analysis. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2019. p. 504–515. ISBN 9781950737482. Available at: <https://www.aclweb.org/anthology/P19-1048>.

HOCHREITER, S.; SCHMIDHUBER, J. Long Short-Term Memory. *Neural Computation*, v. 9, n. 8, p. 1735–1780, 11 1997. ISSN 0899-7667.

HOFFMANN, R.; ZHANG, C.; LING, X.; ZETTLEMOYER, L.; WELD, D. S. Knowledge-based weak supervision for information extraction of overlapping relations. *ACL-HLT 2011 - Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, v. 1, p. 541–550, 2011.

HSU, T. W.; CHEN, C. C.; HUANG, H. H.; CHEN, H. H. Semantics-Preserved Data Augmentation for Aspect-Based Sentiment Analysis. *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, p. 4417–4422, 2021.

HU, M.; LIU, B. Mining and summarizing customer reviews. *KDD-2004 - Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 168–177, 2004.

IRSOY, O.; CARDIE, C. Opinion Mining with Deep Recurrent Neural Networks. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014. p. 720–728. ISBN 9781937284961. Available at: <http://aclweb.org/anthology/D14-1080>.

JOHN, l.; ANDREW, M.; FERNANDO, C. P. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, v. 2001, n. June, p. 282–289, 2001. ISSN 1410-3680. Available at: <https://repository.upenn.edu/cgi/viewcontent.cgi?article=1162&context=cis_papers>.

KASNECI, G.; SUCHANEK, F. M.; IFRIM, G.; ELBASSUONI, S.; RAMANATH, M.; WEIKUM, G. NAGA: Harvesting, searching and ranking knowledge. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, v. 00, p. 1285–1287, 2008. ISSN 07308078.

KIM, H. Towards a sales assistant using a product knowledge graph. *Journal of Web Semantics*, Elsevier B.V., v. 46-47, p. 14–19, 2017. ISSN 15708268. Available at: <http://dx.doi.org/10.1016/j.websem.2017.03.001>.

KIM, Y.; DENTON, C.; HOANG, L.; RUSH, A. M. Structured attention networks. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, p. 1–21, 2017.

KINGMA, D. P.; BA, J. L. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, p. 1–15, 2015.

KOBREN, A.; BARRIO, P.; YAKHNENKO, O.; HIBSCHMAN, J.; LANGMORE, I. Constructing High Precision Knowledge Bases with Subjective and Factual Attributes. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. New York, NY, USA: ACM, 2019. p. 2050–2058. ISBN 9781450362016.

LEWIS, M.; LIU, Y.; GOYAL, N.; GHAZVININEJAD, M.; MOHAMED, A.; LEVY, O.; STOYANOV, V.; ZETTLEMOYER, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. p. 7871–7880, 2020.

LI, F. L.; CHEN, H.; XU, G.; QIU, T.; JI, F.; ZHANG, J.; CHEN, H. AliMeKG: Domain Knowledge Graph Construction and Application in E-commerce. *International Conference on Information and Knowledge Management, Proceedings*, p. 2581–2588, 2020.

LI, J.; YU, J.; XIA, R. Generative Cross-Domain Data Augmentation for Aspect and Opinion Co-Extraction. *NAACL 2022 - 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, p. 4219–4229, 2022.

LI, X.; BING, L.; LI, P.; LAM, W.; YANG, Z. Aspect term extraction with history attention and selective transformation. *IJCAI International Joint Conference on Artificial Intelligence*, v. 2018-July, p. 4194–4200, 2018. ISSN 10450823.

LI, X.; LAM, W. Deep Multi-Task Learning for Aspect Term Extraction with Memory Interaction. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2017. p. 2886–2892. Available at: <http://aclweb.org/anthology/D17-1310>.

LI, Y.; FENG, A.; LI, J.; MUMICK, S.; HALEVY, A.; LI, V.; TAN, W.-C. Subjective databases. *Proceedings of the VLDB Endowment*, v. 12, n. 11, p. 1330–1343, 7 2019. ISSN 2150-8097.

LIN, T. Y.; GOYAL, P.; GIRSHICK, R.; HE, K.; DOLLAR, P. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 42, n. 2, p. 318–327, 2020. ISSN 19393539.

LIU, B. Sentiment analysis and subjectivity. *Handbook of Natural Language Processing, Second Edition*, p. 627–666, 2010.

LIU, B. *Sentiment Analysis*. Cambridge: Cambridge University Press, 2015. ISBN 9781139084789. Available at: <http://ebooks.cambridge.org/ref/id/CBO9781139084789>.

LIU, M.; FANG, Y.; CHOULOS, A. G.; PARK, D. H.; HU, X. Product review summarization through question retrieval and diversification. *Information Retrieval Journal*, Springer Netherlands, v. 20, n. 6, p. 575–605, 12 2017. ISSN 1386-4564. Available at: <http://link.springer.com/10.1007/s10791-017-9311-0>.

LIU, P.; JOTY, S.; MENG, H. Fine-grained Opinion Mining with Recurrent Neural Networks and Word Embeddings. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2015. p. 1433–1443. ISBN 9781941643327. Available at: <http://aclweb.org/anthology/D15-1168>.

LIU, P.; YUAN, W.; FU, J.; JIANG, Z.; HAYASHI, H.; NEUBIG, G. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. p. 1–46, 2021. Available at: <http://arxiv.org/abs/2107.13586>.

LIU, R.; XU, G.; JIA, C.; MA, W.; WANG, L.; VOSOUGHI, S. Data boost: Text data augmentation through reinforcement learning guided conditional generation. *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, p. 9031–9041, 2020.

LOCKARD, C.; DONG, X. L.; EINOLGHOZATI, A.; SHIRALKAR, P. CERES: Distantly Supervised Relation Extraction from the Semi-Structured Web. *Proceedings of the VLDB Endowment*, v. 11, n. 10, p. 1084–1096, 6 2018. ISSN 21508097. Available at: <http://dl.acm.org/citation.cfm?doid=3231751.3242930>.

LUO, Z.; HUANG, S.; ZHU, K. Q. Knowledge empowered prominent aspect extraction from product reviews. *Information Processing and Management*, Elsevier, v. 56, n. 3, p. 408–423, 2019. ISSN 03064573. Available at: <https://doi.org/10.1016/j.ipm.2018.11.006>.

MCAULEY, J.; TARGETT, C.; SHI, Q.; HENGEL, A. van den. Image-Based Recommendations on Styles and Substitutes. In: *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

New York, NY, USA: ACM, 2015. p. 43–52. ISBN 9781450336215. Available at: <https://dl.acm.org/doi/10.1145/2766462.2767755>.

MCAULEY, J.; YANG, A. Addressing Complex and Subjective Product-Related Queries with Customer Reviews. In: *Proceedings of the 25th International Conference on World Wide Web*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2016. p. 625–635. ISBN 9781450341431. Available at: <https://dl.acm.org/doi/10.1145/2872427.2883044>.

MELO, T. de; SILVA, A. S. da; MOURA, E. S. de; CALADO, P. OpinionLink: Leveraging user opinions for product catalog enrichment. *Information Processing and Management*, Elsevier, v. 56, n. 3, p. 823–843, 2019. ISSN 03064573. Available at: <https://doi.org/10.1016/j.ipm.2019.01.004>.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G.; DEAN, J. Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, v. 90, n. 6, p. 795–803, 10 2013. ISSN 0002-8703. Available at: <http://arxiv.org/abs/1310.4546>.

MIN, S.; LEWIS, M.; ZETTLEMOYER, L.; HAJISHIRZI, H. MetaICL: Learning to Learn In Context. *NAACL 2022 - 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, n. 1, p. 2791–2809, 2022. Available at: <http://arxiv.org/abs/2110.15943>.

MINTZ, M.; BILLS, S.; SNOW, R.; JURAFSKY, D. Distant supervision for relation extraction without labeled data. In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - ACL-IJCNLP '09*. [S.l.: s.n.], 2009. v. 2, n. August, p. 1003. ISBN 9781932432466. ISSN 1932432469.

MOREIRA, J.; MELO, T. de; BARBOSA, L.; SILVA, A. d. A distantly supervised approach for enriching product graphs with user opinions. *Journal of Intelligent Information Systems*, 5 2022. ISSN 0925-9902. Available at: <https://link.springer.com/10.1007/s10844-022-00717-5>.

MOUSSA, M. E.; MOHAMED, E. H.; HAGGAG, M. H. A survey on opinion summarization techniques for social media. *Future Computing and Informatics Journal*, Elsevier Ltd, v. 3, n. 1, p. 82–109, 2018. ISSN 23147288. Available at: <https://doi.org/10.1016/j.fcij.2017.12.002>.

PETRONI, F.; ROCKTÄSCHEL, T.; RIEDEL, S.; LEWIS, P.; BAKHTIN, A.; WU, Y.; MILLER, A. Language Models as Knowledge Bases? In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2019. p. 2463–2473. ISBN 9781950737901. Available at: <https://www.aclweb.org/anthology/D19-1250>.

PONTIKI, M.; GALANIS, D.; PAPAGEORGIOU, H.; MANANDHAR, S.; ANDROUTSOPOULOS, I. SemEval-2015 Task 12: Aspect Based Sentiment Analysis. *SemEval 2015 - 9th International Workshop on Semantic Evaluation, co-located with the*

2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2015 - Proceedings, p. 486–495, 2015.

PONTIKI, M.; GALANIS, D.; PAPAGEORGIOU, H.; ANDROUTSOPOULOS, I.; MANANDHAR, S.; AL-SMADI, M.; AL-AYYOUB, M.; ZHAO, Y.; QIN, B.; CLERCQ, O. D.; HOSTE, V.; APIDIANAKI, M.; TANNIER, X.; LOUKACHEVITCH, N.; KOTELNIKOV, E.; BEL, N.; JIMÉNEZ-ZAFRA, S. M.; ERYIĞIT, G. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2016. P-232, p. 19–30. ISBN 9783885796268. ISSN 16175468. Available at: <http://aclweb.org/anthology/S16-1002>.

PONTIKI, M.; GALANIS, D.; PAVLOPOULOS, J.; PAPAGEORGIOU, H.; ANDROUTSOPOULOS, I.; MANANDHAR, S. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In: *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014. p. 27–35. Available at: <http://aclweb.org/anthology/S14-2004>.

PORIA, S.; CAMBRIA, E.; KU, L.-W.; GUI, C.; GELBUKH, A. A Rule-Based Approach to Aspect Extraction from Product Reviews. In: *Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics and Dublin City University, 2014. p. 28–37. Available at: <http://aclweb.org/anthology/W14-5905>.

RADFORD, A.; NARASIMHAN, K.; SALIMANS, T.; SUTSKEVER, I. Improving language understanding by generative pre-training. 2018.

RADFORD, A.; WU, J.; CHILD, R.; LUAN, D.; AMODEI, D.; SUTSKEVER, I. Language Models are Unsupervised Multitask Learners. 2019.

RAFFEL, C.; SHAZEER, N.; ROBERTS, A.; LEE, K.; NARANG, S.; MATENA, M.; ZHOU, Y.; PETER, W. L.; LIU, J. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv*, v. 21, p. 1–67, 2019. ISSN 23318422.

RIEDEL, S.; YAO, L.; MCCALLUM, A. Modeling Relations and Their Mentions without Labeled Text. In: *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases: Part III*. Berlin, Heidelberg: Springer-Verlag, 2010. p. 148–163. ISBN 3-642-15938-9, 978-3-642-15938-1.

ROBERTS, A.; RAFFEL, C.; SHAZEER, N. How Much Knowledge Can You Pack Into the Parameters of a Language Model? p. 5418–5426, 2020.

ROTH, B.; BARTH, T.; WIEGAND, M.; KLAKOW, D. A survey of noise reduction methods for distant supervision. In: *Proceedings of the 2013 workshop on Automated knowledge base construction - AKBC '13*. New York, New York, USA: ACM Press, 2013. p. 73–78. ISBN 9781450324113. Available at: <http://dl.acm.org/citation.cfm?id=2509571http://dl.acm.org/citation.cfm?doid=2509558.2509571>.

SANH, V.; DEBUT, L.; CHAUMOND, J.; WOLF, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. p. 2–6, 2019. Available at: <http://arxiv.org/abs/1910.01108>.

SONG, Q.; WU, Y.; DONG, X. L. Mining summaries for knowledge graph search. *Proceedings - IEEE International Conference on Data Mining, ICDM*, IEEE, p. 1215–1220, 2017. ISSN 15504786.

SURDEANU, M.; TIBSHIRANI, J.; NALLAPATI, R.; MANNING, C. D. Multi-instance Multi-label Learning for Relation Extraction. n. July, p. 455–465, 2010.

TAI, Y.-J.; KAO, H.-Y. Automatic Domain-Specific Sentiment Lexicon Generation with Label Propagation. In: *Proceedings of International Conference on Information Integration and Web-based Applications & Services - IIWAS '13*. New York, New York, USA: ACM Press, 2013. p. 53–62. ISBN 9781450321136. Available at: <http://dl.acm.org/citation.cfm?doid=2539150.2539190>.

TAKAMATSU, S.; SATO, I.; NAKAGAWA, H. Reducing Wrong Labels in Distant Supervision for Relation Extraction. *Jeju, Republic of Korea*, n. July, p. 721–729, 2012.

TAVOR, A. A.; CARMELI, B.; GOLDBRAICH, E.; KANTOR, A.; KOUR, G.; SHLOMOV, S.; TEPPER, N.; ZWERDLING, N. Do not have enough data? Deep learning to the rescue! *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, p. 7383–7390, 2020. ISSN 2159-5399.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. In: *Advances in Neural Information Processing Systems*. [S.l.]: Curran Associates, Inc., 2017. v. 30.

VICENTE, I. S.; AGERRI, R.; RIGAU, G. Simple, Robust and (almost) Unsupervised Generation of Polarity Lexicons for Multiple Languages. In: *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014. p. 88–97. ISBN 9781632663962. Available at: <http://aclweb.org/anthology/E14-1010>.

WANG, C.; LIU, X.; CHEN, Z.; HONG, H.; TANG, J.; SONG, D. Zero-Shot Information Extraction as a Unified Text-to-Triple Translation. 2021. Available at: <http://arxiv.org/abs/2109.11171>.

WANG, C.; LIU, X.; SONG, D. Language Models are Open Knowledge Graphs. p. 1–30, 2020. Available at: <http://arxiv.org/abs/2010.11967>.

WANG, W.; PAN, S. J.; DAHLMEIER, D.; XIAO, X. Recursive neural conditional random fields for aspect-based sentiment analysis. *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, p. 616–626, 2016.

WANG, W.; PAN, S. J.; DAHLMEIER, D.; XIAO, X. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, p. 3316–3322, 2017.

WANG, Z.; YU, A. W.; FIRAT, O.; CAO, Y. Towards Zero-Label Language Learning. 2021. Available at: <http://arxiv.org/abs/2109.09193>.

WU, S.; FEI, H.; REN, Y.; LI, B.; LI, F.; JI, D. High-Order Pair-Wise Aspect and Opinion Terms Extraction With Edge-Enhanced Syntactic Graph Convolution. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, IEEE, v. 29, p. 2396–2406, 2021. ISSN 2329-9290.

WU, Z.; YING, C.; ZHAO, F.; FAN, Z.; DAI, X.; XIA, R. Grid Tagging Scheme for Aspect-oriented Fine-grained Opinion Extraction. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020. p. 2576–2585. Available at: <https://www.aclweb.org/anthology/2020.findings-emnlp.234>.

WU, Z.; ZHAO, F.; DAI, X.-Y.; HUANG, S.; CHEN, J. Latent Opinions Transfer Network for Target-Oriented Opinion Words Extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, v. 34, n. 05, p. 9298–9305, 2020. ISSN 2159-5399.

XU, D.; RUAN, C.; KORPEOGLU, E.; KUMAR, S.; ACHAN, K. Product Knowledge Graph Embedding for E-commerce. In: *Proceedings of the 13th International Conference on Web Search and Data Mining*. New York, NY, USA: ACM, 2020. p. 672–680. ISBN 9781450368223. Available at: <https://dl.acm.org/doi/10.1145/3336191.3371778>.

XU, H.; LIU, B.; SHU, L.; YU, P. S. Double embeddings and cnn-based sequence labeling for aspect extraction. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, v. 2, p. 592–598, 2018.

XU, H.; SHU, L.; YU, P.; LIU, B. Understanding Pre-trained BERT for Aspect-based Sentiment Analysis. In: *Proceedings of the 28th International Conference on Computational Linguistics*. Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020. p. 244–250. Available at: <https://www.aclweb.org/anthology/2020.coling-main.21>.

YOO, K. M.; PARK, D.; KANG, J.; LEE, S. W.; PARK, W. GPT3Mix: Leveraging Large-scale Language Models for Text Augmentation. *Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021*, p. 2225–2239, 2021.

ZHANG, J.; LI, F.; ZHANG, Z.; XU, G.; WANG, Y.; WANG, X.; ZHANG, Y. Integrate syntax information for target-oriented opinion words extraction with target-specific graph convolutional network. *Neurocomputing*, Elsevier, v. 440, p. 321–335, 2021. ISSN 18728286. Available at: <https://doi.org/10.1016/j.neucom.2020.07.152>.

ZHANG, J.; ZHANG, Z.; GUO, Z.; JIN, L.; LIU, K.; LIU, Q. Enhancement of target-oriented opinion words extraction with multiview-trained machine reading comprehension model. *Computational Intelligence and Neuroscience*, v. 2021, 2021. ISSN 16875273.

ZHANG, M.; FAN, B.; ZHANG, N.; WANG, W.; FAN, W. Mining product innovation ideas from online reviews. *Information Processing and Management*, Elsevier, v. 58, n. 1, p. 102389, 2021. ISSN 03064573. Available at: <https://doi.org/10.1016/j.ipm.2020.102389>.

ZHANG, Y.; DAI, H.; KOZAREVA, Z.; SMOLA, A. J.; SONG, L. Variational Reasoning for Question Answering with Knowledge Graph. *AAAI*, v. 32, n. 1, p. 6069–6076, 9 2018.

ZHAO, H.; HUANG, L.; ZHANG, R.; LU, Q.; XUE, H. SpanMlt: A Span-based Multi-Task Learning Framework for Pair-wise Aspect and Opinion Terms Extraction. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020. p. 3239–3248. Available at: <https://www.aclweb.org/anthology/2020.acl-main.296>.

ZHAO, T. Z.; WALLACE, E.; FENG, S.; KLEIN, D.; SINGH, S. Calibrate Before Use: Improving Few-Shot Performance of Language Models. 2021. Available at: <http://arxiv.org/abs/2102.09690>.

ZHENG, W.; YU, J. X.; ZOU, L.; CHENG, H. Question answering over knowledge graphs: Question understanding via template decomposition. *Proceedings of the VLDB Endowment*, v. 11, n. 11, p. 1373–1386, 2018. ISSN 21508097.

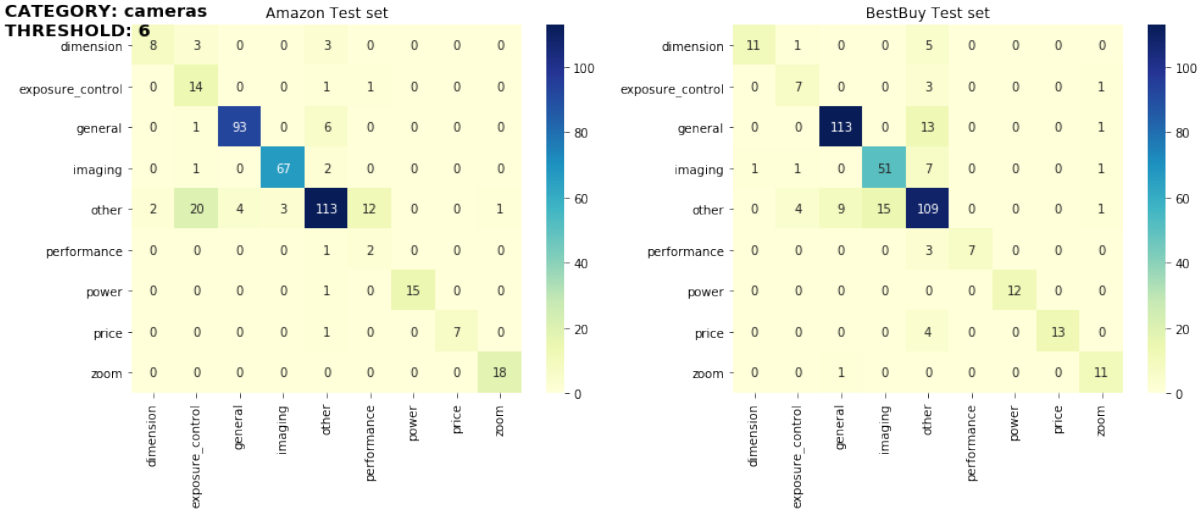
ZHOU, Y.; JIANG, W.; SONG, P.; SU, Y.; GUO, T.; HAN, J.; HU, S. Graph Convolutional Networks for Target-oriented Opinion Words Extraction with Adversarial Training. *Proceedings of the International Joint Conference on Neural Networks*, 2020.

ZHUANG, L.; JING, F.; ZHU, X.-Y.; HE, R.; LEE, W. S.; NG, H. T.; DAHLMEIER, D. Movie review mining and summarization. In: *Proceedings of the 15th ACM international conference on Information and knowledge management - CIKM '06*. New York, New York, USA: ACM Press, 2006. p. 43. ISBN 9781950737482. Available at: <http://portal.acm.org/citation.cfm?doid=1183614.1183625>.

APPENDIX A – PGOPI CONFUSION MATRICES AND CLASSIFICATION REPORTS

Here, we present the confusion matrices and the classification reports for details on the performance of the PGOpi pipeline on each target of each product category. We select the best threshold from Table 10 and used the best model (combination of hyperparameters) to report these values.

Figure 12 – Confusion matrix for PGOpi model on the category `Cameras` on both analyzed datasets (*Amazon* and *Bestbuy*) using $threshold = 0.6$.



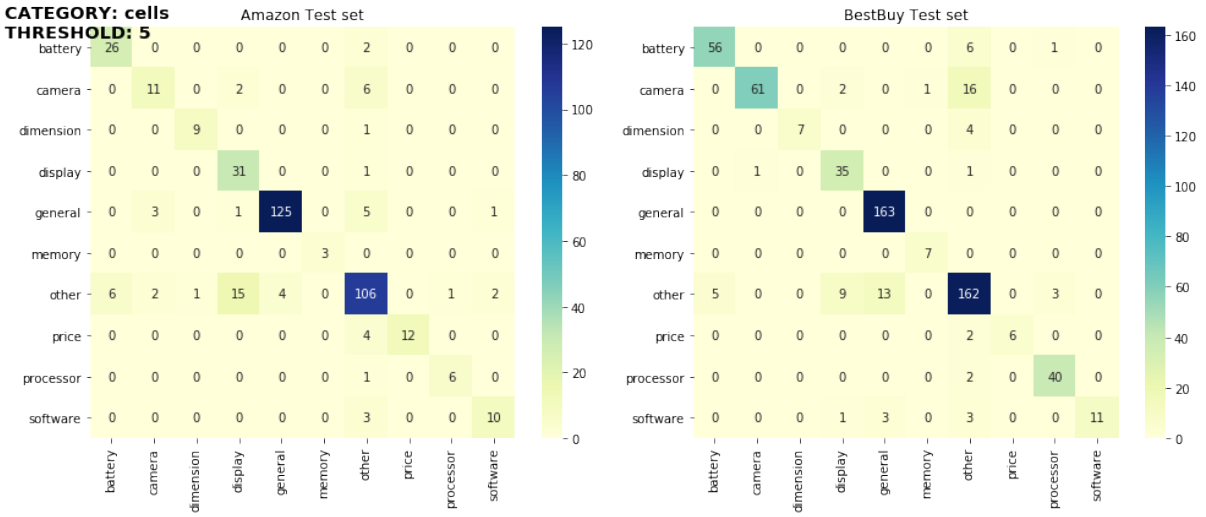
Source: Created by the author

Table 19 – Classification Reports for the `Cameras` category on the *Amazon* (left) and *BestBuy* (right) datasets with $threshold = 0.6$.

	AMAZON					BESTBUY			
	Precision	Recall	F-score	Support	Class	Precision	Recall	F-score	Support
dimension	0.8	0.57	0.67	14	dimension	0.92	0.65	0.76	17
exp_control	0.36	0.88	0.51	16	exp_control	0.54	0.64	0.58	11
general	0.96	0.93	0.94	100	general	0.92	0.89	0.9	127
imaging	0.96	0.96	0.96	70	imaging	0.77	0.84	0.8	61
other	0.88	0.73	0.8	155	other	0.76	0.79	0.77	138
performance	0.13	0.67	0.22	3	performance	1.0	0.7	0.82	10
power	1.0	0.94	0.97	16	power	1.0	1.0	1.0	12
price	1.0	0.88	0.93	8	price	1.0	0.76	0.87	17
zoom	0.95	1.0	0.97	18	zoom	0.73	0.92	0.81	12
accuracy			**0.84**	400	accuracy			**0.82**	405
macro avg	0.78	0.84	0.77	400	macro avg	0.85	0.8	0.81	405
avg	0.9	0.84	0.86	400	avg	0.83	0.82	0.83	405

Source: Created by the author

Figure 13 – Confusion matrix for PGOpi model on the category `Cells` on both analyzed datasets (*Amazon* and *Bestbuy*) using $threshold = 0.5$.



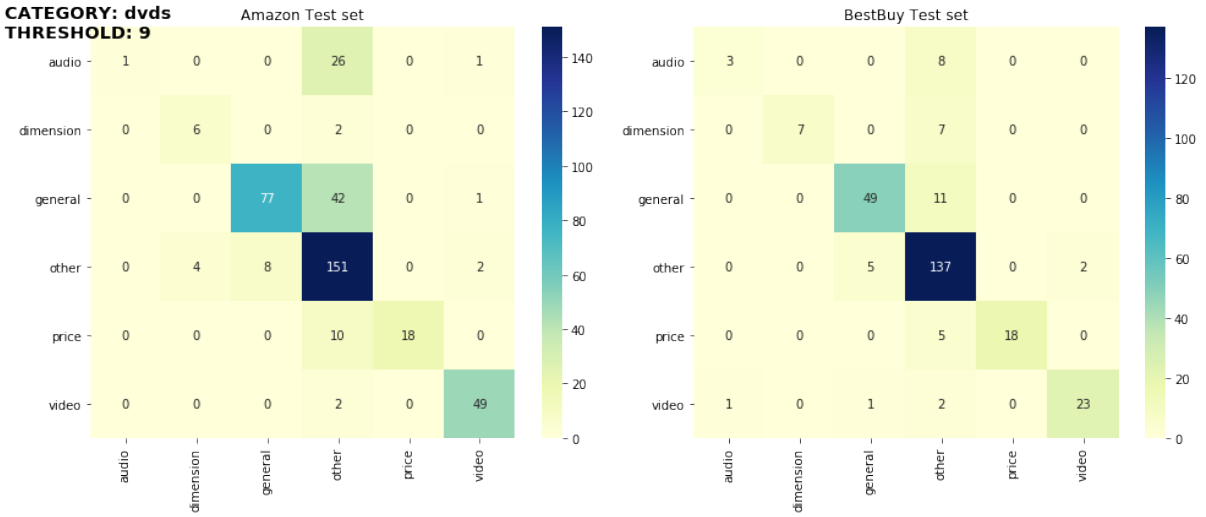
Source: Created by the author

Table 20 – Classification Reports for the `Cells` category on the *Amazon* (left) and *BestBuy* (right) datasets with $threshold = 0.5$.

	AMAZON					BESTBUY			
	Precision	Recall	F-score	Support	Class	Precision	Recall	F-score	Support
battery	0.81	0.93	0.87	28	battery	0.92	0.89	0.9	63
camera	0.69	0.58	0.63	19	camera	0.98	0.76	0.86	80
dimension	0.9	0.9	0.9	10	dimension	1.0	0.64	0.78	11
display	0.63	0.97	0.77	32	display	0.74	0.95	0.83	37
general	0.97	0.93	0.95	135	general	0.91	1.0	0.95	163
memory	1.0	1.0	1.0	3	memory	0.88	1.0	0.93	7
other	0.82	0.77	0.8	137	other	0.83	0.84	0.84	192
price	1.0	0.75	0.86	16	price	1.0	0.75	0.86	8
processor	0.86	0.86	0.86	7	processor	0.91	0.95	0.93	42
software	0.77	0.77	0.77	13	software	1.0	0.61	0.76	18
accuracy			**0.85**	400	accuracy			**0.88**	621
macro avg	0.84	0.85	0.84	400	macro avg	0.92	0.84	0.86	621
avg	0.86	0.85	0.85	400	avg	0.89	0.88	0.88	621

Source: Created by the author

Figure 14 – Confusion matrix for PGOpi model on the category DVDs on both analyzed datasets (*Amazon* and *Bestbuy*) using *threshold* = 0.9.



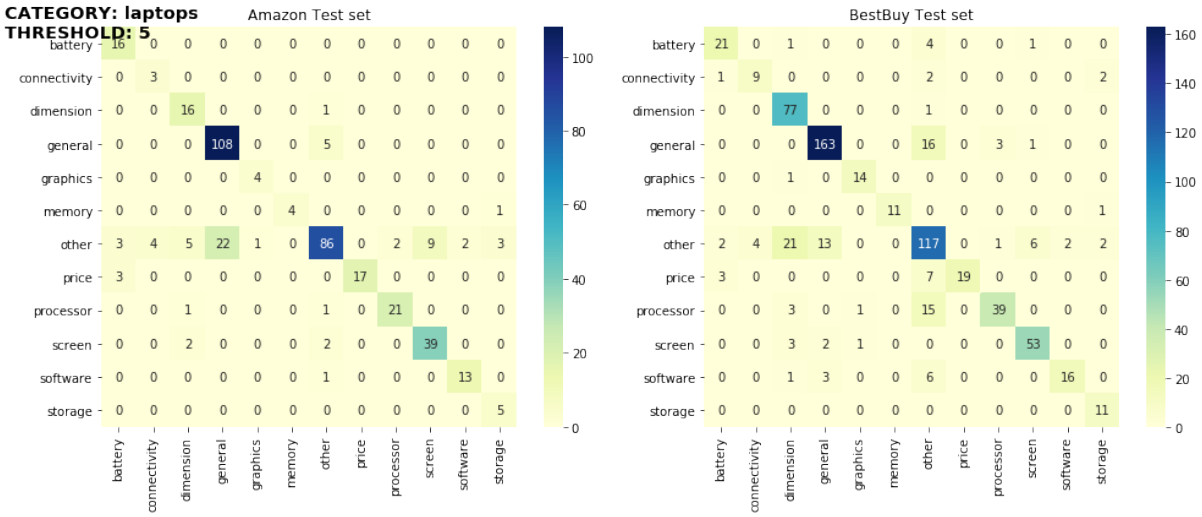
Source: Created by the author

Table 21 – Classification Reports for the DVDs category on the *Amazon* (left) and *BestBuy* (right) datasets with *threshold* = 0.9.

	AMAZON					BESTBUY			
	Precision	Recall	F-score	Support		Precision	Recall	F-score	Support
audio	1.0	0.04	0.07	28	audio	0.75	0.27	0.4	11
dimension	0.6	0.75	0.67	8	dimension	1.0	0.5	0.67	14
general	0.91	0.64	0.75	120	general	0.89	0.82	0.85	60
other	0.65	0.92	0.76	165	other	0.81	0.95	0.87	144
price	1.0	0.64	0.78	28	price	1.0	0.78	0.88	23
video	0.92	0.96	0.94	51	video	0.92	0.85	0.88	27
accuracy			**0.76**	400	accuracy			**0.85**	279
macro avg	0.85	0.66	0.66	400	macro avg	0.89	0.7	0.76	279
avg	0.81	0.76	0.73	400	avg	0.86	0.85	0.84	279

Source: Created by the author

Figure 15 – Confusion matrix for PGOpi model on the category `Laptops` on both analyzed datasets (*Amazon* and *Bestbuy*) using *threshold* = 0.5.



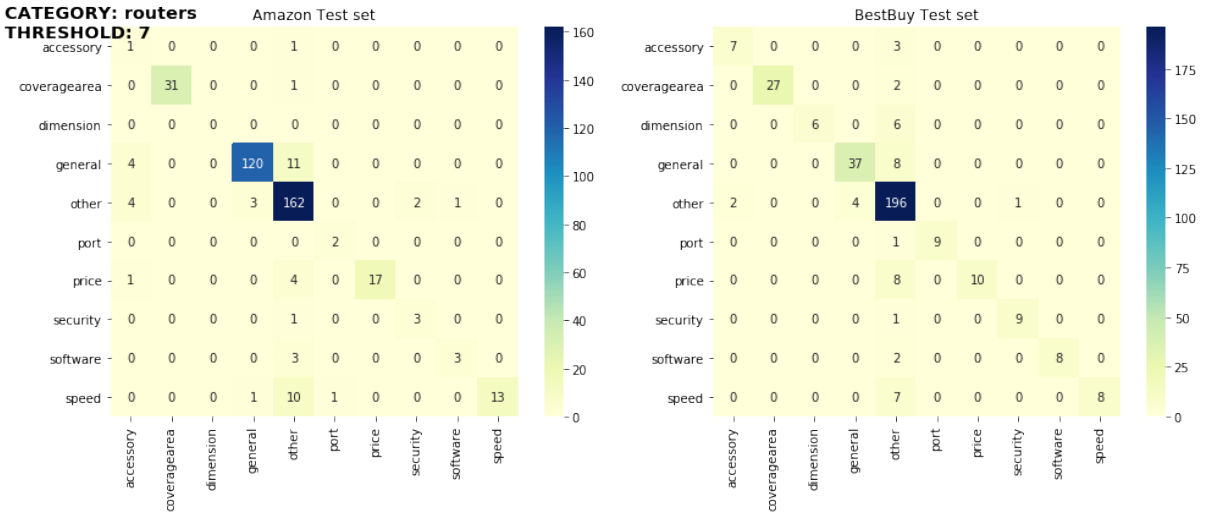
Source: Created by the author

Table 22 – Classification Reports for the `Laptops` category on the *Amazon* (left) and *BestBuy* (right) datasets with *threshold* = 0.5.

	AMAZON					BESTBUY			
Class	Precision	Recall	F-score	Support		Precision	Recall	F-score	Support
battery	0.73	1.0	0.84	16	battery	0.78	0.78	0.78	27
connectivity	0.43	1.0	0.6	3	connectivity	0.69	0.64	0.67	14
dimension	0.67	0.94	0.78	17	dimension	0.72	0.99	0.83	78
general	0.83	0.96	0.89	113	general	0.9	0.89	0.9	183
graphics	0.8	1.0	0.89	4	graphics	0.88	0.93	0.9	15
memory	1.0	0.8	0.89	5	memory	1.0	0.92	0.96	12
other	0.9	0.63	0.74	137	other	0.7	0.7	0.7	168
price	1.0	0.85	0.92	20	price	1.0	0.66	0.79	29
processor	0.91	0.91	0.91	23	processor	0.91	0.67	0.77	58
screen	0.81	0.91	0.86	43	screen	0.87	0.9	0.88	59
software	0.87	0.93	0.9	14	software	0.89	0.62	0.73	26
storage	0.56	1.0	0.71	5	storage	0.69	1.0	0.81	11
accuracy			**0.83**	400	accuracy			**0.81**	680
macro avg	0.79	0.91	0.83	400	macro avg	0.83	0.81	0.81	680
avg	0.85	0.83	0.83	400	avg	0.82	0.81	0.81	680

Source: Created by the author

Figure 16 – Confusion matrix for PGOpi model on the category `Routers` on both analyzed datasets (*Amazon* and *Bestbuy*) using *threshold* = 0.7.



Source: Created by the author

Table 23 – Classification Reports for the `Routers` category on the *Amazon* (left) and *BestBuy* (right) datasets with *threshold* = 0.7.

	AMAZON					BESTBUY			
	Precision	Recall	F-score	Support		Precision	Recall	F-score	Support
accessory	0.1	0.5	0.17	2	accessory	0.78	0.7	0.74	10
coveragearea	1.0	0.97	0.98	32	coveragearea	1.0	0.93	0.96	29
dimension	0.0	0.0	0.0	0	dimension	1.0	0.5	0.67	12
general	0.97	0.89	0.93	135	general	0.9	0.82	0.86	45
other	0.84	0.94	0.89	172	other	0.84	0.97	0.9	203
port	0.67	1.0	0.8	2	port	1.0	0.9	0.95	10
price	1.0	0.77	0.87	22	price	1.0	0.56	0.71	18
security	0.6	0.75	0.67	4	security	0.9	0.9	0.9	10
software	0.75	0.5	0.6	6	software	1.0	0.8	0.89	10
speed	1.0	0.52	0.68	25	speed	1.0	0.53	0.7	15
micro avg	0.88	0.88	**0.88**	400	accuracy			**0.88**	362
macro avg	0.69	0.68	0.66	400	macro avg	0.94	0.76	0.83	362
avg	0.91	0.88	0.88	400	avg	0.89	0.88	0.87	362

Source: Created by the author

APPENDIX B – FULL F1-SCORE RESULTS FOR THE MODELS' PERFORMANCE IN THE TOWE TASK

Table 24 – F1-score for benchmarking performance between methods for TOWE using the original (ORG) datasets against the ones autonomously generated with In-Context Learning approaches. GEN - Methods were trained only with the synthetic data, AUG - Methods were trained with full original training set augmented with a percentage of synthetic samples. Boldface indicates scenarios where the In-Context Learning approach shows equal or superior performance compared to 100% of the manually built training set.

F1-SCORE		IOG						TSMSA					
		ZERO-SHOT		ONE-SHOT		FEW-SHOT		ZERO-SHOT		ONE-SHOT		FEW-SHOT	
	SUBSET (%)	ORG	GEN	AUG	GEN	AUG	GEN	AUG	ORG	GEN	AUG	GEN	AUG	GEN	AUG
14lap	10	0.57	0.64	**0.72**	0.53	0.68	0.54	0.67	0.63	0.69	0.78	0.63	**0.80**	0.60	0.79
	25	0.65	0.62	0.68	0.57	0.66	0.58	0.65	0.73	0.73	0.76	0.64	0.77	0.65	0.79
	50	0.66	0.66	0.70	0.59	0.70	0.60	0.69	0.76	0.69	0.74	0.68	0.76	0.65	0.78
	75	0.68	0.67	0.69	0.60	0.62	0.62	0.68	0.78	0.74	0.75	0.67	0.77	0.68	0.76
	100	**0.71**	0.68	0.70	0.62	0.63	0.64	0.66	**0.79**	0.73	0.76	0.69	0.76	0.67	0.75
14res	10	0.66	0.68	**0.79**	0.61	0.76	0.59	0.77	0.74	0.79	**0.85**	0.73	**0.85**	0.68	0.84
	25	0.70	0.68	0.77	0.65	0.77	0.63	0.77	0.82	0.80	0.83	0.76	**0.85**	0.73	0.82
	50	0.74	0.73	**0.79**	0.70	0.73	0.67	0.75	0.84	0.79	0.83	0.74	0.83	0.76	0.83
	75	0.76	0.73	0.77	0.70	0.77	0.66	0.72	0.85	0.79	0.83	0.76	0.83	0.75	0.82
	100	**0.78**	0.75	0.77	0.69	0.76	0.66	0.74	**0.85**	0.81	0.84	0.76	0.83	0.76	0.81
15res	10	0.57	0.64	**0.72**	0.62	**0.73**	0.54	0.69	0.73	0.70	**0.80**	0.69	**0.79**	0.66	0.78
	25	0.64	0.65	0.70	0.64	0.69	0.61	0.68	0.74	0.74	**0.79**	0.72	0.78	0.68	0.77
	50	0.66	0.65	**0.72**	0.64	0.69	0.60	0.69	0.78	0.71	**0.79**	0.73	0.77	0.68	0.77
	75	0.70	0.64	**0.71**	0.67	0.68	0.64	0.65	0.80	0.71	0.77	0.72	0.76	0.68	0.75
	100	**0.71**	0.69	0.68	0.68	**0.71**	0.62	0.65	**0.79**	0.74	0.77	0.71	0.75	0.70	0.75
16res	10	0.64	0.72	0.79	0.67	0.79	0.67	0.77	0.78	0.74	**0.89**	0.75	**0.89**	0.76	**0.88**
	25	0.74	0.72	0.79	0.72	**0.80**	0.66	0.78	0.82	0.76	0.87	0.79	0.87	0.76	0.86
	50	0.79	0.73	0.78	0.67	0.79	0.71	0.79	0.88	0.77	0.86	0.77	0.85	0.75	0.83
	75	0.78	0.75	0.79	0.68	0.77	0.69	0.79	0.86	0.76	0.85	0.77	0.83	0.75	0.83
	100	**0.80**	0.74	0.79	0.70	0.77	0.72	0.77	**0.88**	0.76	0.84	0.77	0.84	0.76	0.79

Source: Created by the author

APPENDIX C – VISUALIZATION OF THE EVALUATION SCENARIOS FOR SYNCOPATE IN THE TOWE TASK

Evaluation scenarios for the performance of the IO-LSTM + Global Context (IOG) and Target-Specified Sequence Labeling with Multi-head Self-Attention (TSMSA) models when trained on variations of the SemEval Datasets: `14lap`, `14res`, `15res`, and `16res`.

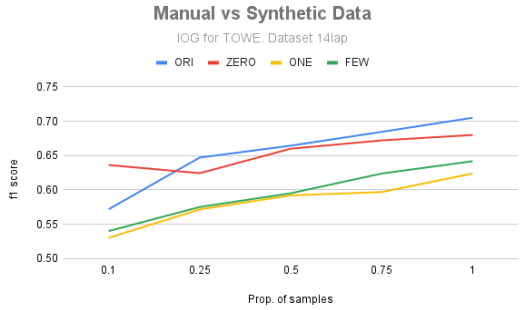
Evaluation Scenarios:

1. Training the models only on manually labeled data (ORI);
2. Training the models only on synthetic opinionated triples (ZERO, ONE, and FEW);
3. Training the models on manually built data augmented with synthetic opinionated triples (ZERO, ONE, and FEW).

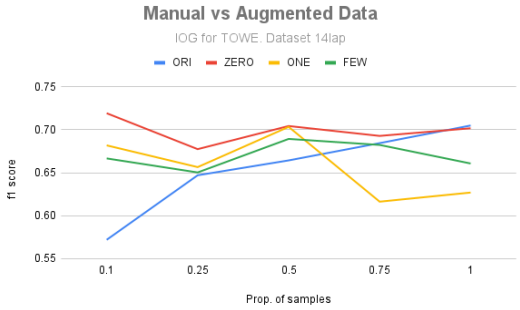
C.1 PERFORMANCES OF THE IOG MODEL

Figure 17 – Performance of the IOG model on variations of the `14lap` dataset

(a) Manual *versus* Synthetic



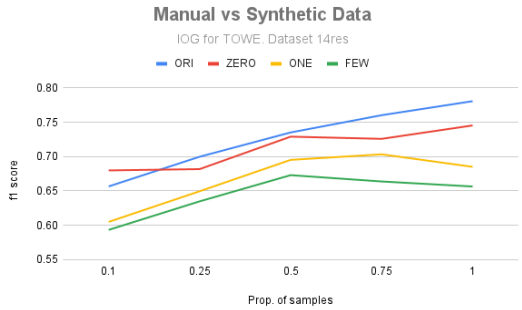
(b) Manual *versus* Augmented



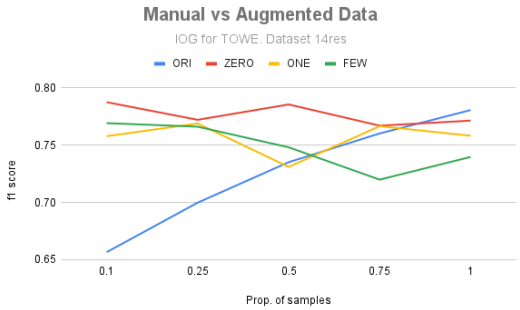
Source: Created by the author

Figure 18 – Performance of the IOG model on variations of the `14res` dataset

(a) Manual *versus* Synthetic



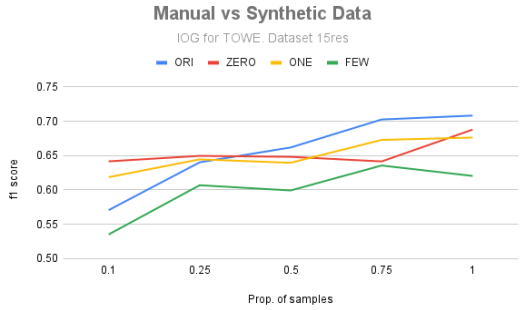
(b) Manual *versus* Augmented



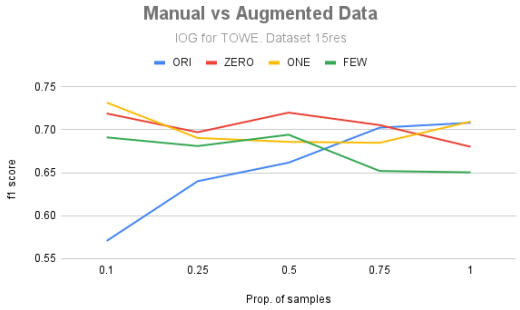
Source: Created by the author

Figure 19 – Performance of the IOG model on variations of the `15res` dataset

(a) Manual *versus* Synthetic



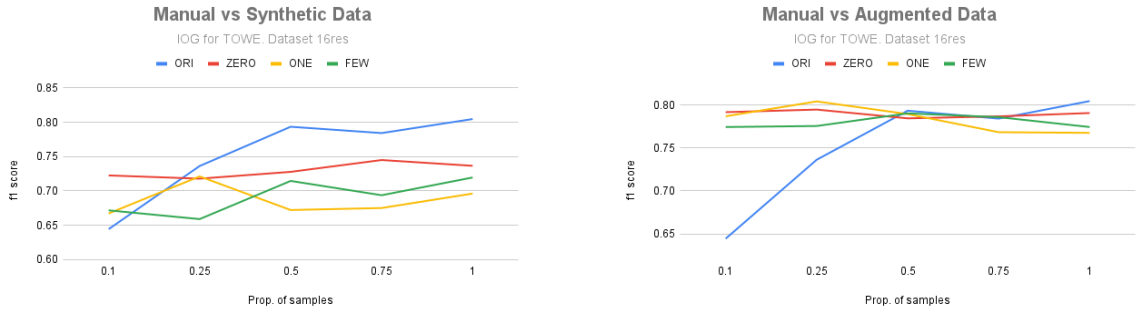
(b) Manual *versus* Augmented



Source: Created by the author

Figure 20 – Performance of the IOG model on variations of the `16res` dataset

(a) Manual *versus* Synthetic (b) Manual *versus* Augmented

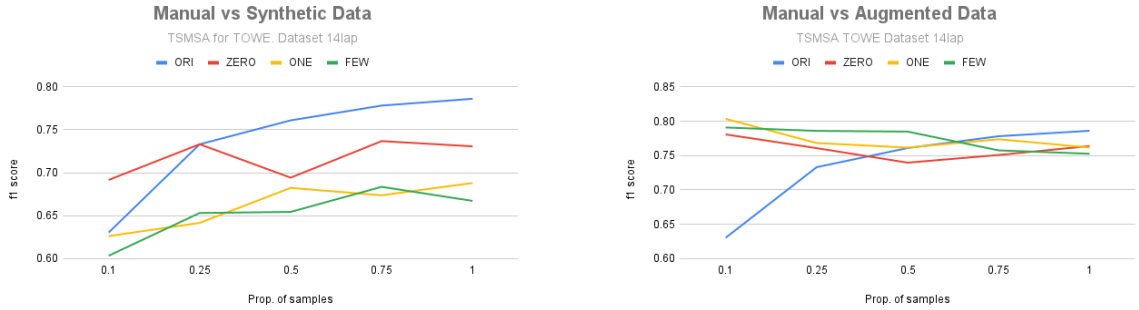


Source: Created by the author

C.2 PERFORMANCES OF THE TSMSA MODEL

Figure 21 – Performance of the TSMSA model on variations of the `14lap` dataset

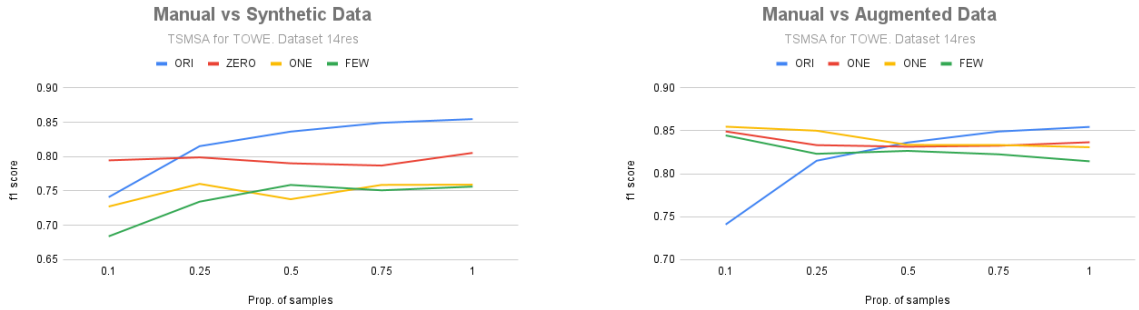
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



Source: Created by the author

Figure 22 – Performance of the TSMSA model on variations of the `14res` dataset

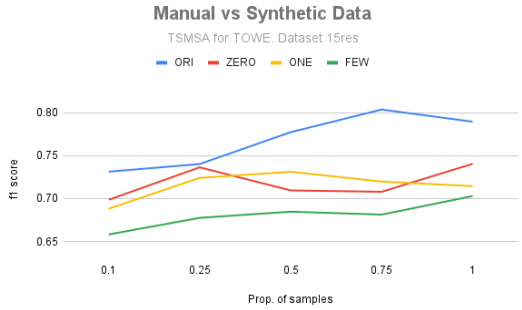
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



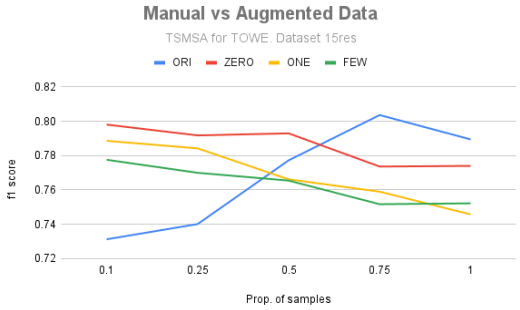
Source: Created by the author

Figure 23 – Performance of the TSMSA model on variations of the `15res` dataset

(a) Manual *versus* Synthetic



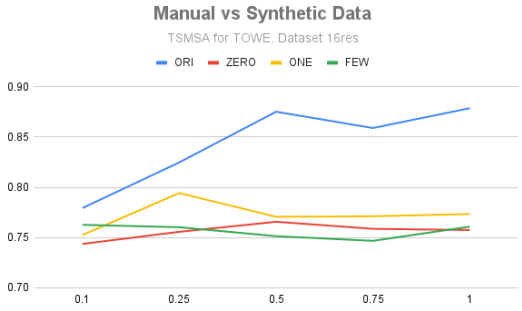
(b) Manual *versus* Augmented



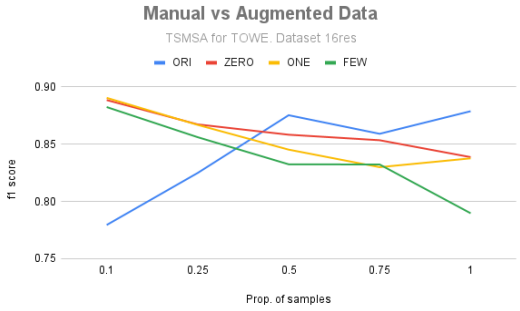
Source: Created by the author

Figure 24 – Performance of the TSMSA model on variations of the `16res` dataset

(a) Manual *versus* Synthetic



(b) Manual *versus* Augmented



Source: Created by the author

APPENDIX D – FULL F1-SCORE RESULTS FOR THE MODELS' PERFORMANCE IN THE AOPE TASK

Table 25 – F1-score for benchmarking performance between methods for AOPE using the original (ORG) datasets against the ones autonomously generated with In-Context Learning approaches. GEN - Methods were trained only with the synthetic data, AUG - Methods were trained with the original training set augmented with a percentage of synthetic samples. Boldface indicates scenarios where the In-Context Learning approach shows equal or superior performance compared to 100% of the manually built training set.

F1 SCORE		SDRN						MT-TSMSA					
		ZERO-SHOT		ONE-SHOT		FEW-SHOT		ZERO-SHOT		ONE-SHOT		FEW-SHOT	
	SUBSET (%)	ORG	GEN	AUG	GEN	AUG	GEN	AUG	ORG	GEN	AUG	GEN	AUG	GEN	AUG
14lap	10	0.37	0.38	0.51	0.34	0.51	0.31	0.50	0.47	0.47	0.60	0.38	0.59	0.38	**0.61**
	25	0.44	0.42	0.52	0.37	**0.53**	0.36	0.49	0.52	0.47	0.59	0.39	0.57	0.42	0.57
	50	0.48	0.46	0.52	0.43	0.51	0.39	0.51	0.57	0.47	0.57	0.45	0.55	0.40	0.56
	75	0.49	0.45	0.52	0.41	0.51	0.39	0.50	0.60	0.49	0.55	0.44	0.56	0.46	0.58
	100	**0.53**	0.46	0.52	0.43	0.52	0.41	0.49	**0.61**	0.49	0.56	0.46	0.57	0.45	0.56
14res	10	0.38	0.41	0.48	0.36	0.48	0.35	0.48	0.49	0.47	**0.55**	0.50	**0.56**	0.45	**0.53**
	25	0.43	0.45	**0.50**	0.40	**0.49**	0.37	0.46	0.54	0.50	**0.55**	0.50	**0.56**	0.47	**0.56**
	50	0.45	0.44	0.48	0.42	**0.49**	0.41	0.47	0.51	0.46	**0.56**	0.49	**0.57**	0.48	0.52
	75	0.47	0.43	**0.50**	0.42	0.47	0.41	0.46	0.57	0.51	**0.56**	0.52	**0.54**	0.49	**0.56**
	100	**0.49**	0.45	0.48	0.43	0.48	0.42	0.45	**0.53**	0.49	**0.57**	0.52	**0.54**	0.49	**0.55**
15res	10	0.39	0.40	0.52	0.36	0.52	0.34	0.49	0.52	0.43	**0.61**	0.45	**0.61**	0.44	0.57
	25	0.49	0.40	0.53	0.40	0.48	0.38	0.46	0.52	0.43	0.59	0.49	0.58	0.45	0.55
	50	0.51	0.43	0.53	0.43	0.49	0.39	0.45	0.55	0.45	0.59	0.46	0.56	0.44	0.54
	75	0.51	0.41	0.52	0.43	0.48	0.40	0.46	0.61	0.48	0.58	0.49	0.52	0.45	0.50
	100	**0.54**	0.43	0.52	0.45	0.48	0.42	0.46	**0.60**	0.46	0.56	0.47	0.51	0.46	0.51
16res	10	0.43	0.47	**0.61**	0.45	0.58	0.42	0.57	0.56	0.54	**0.65**	0.55	**0.66**	0.52	0.62
	25	0.52	0.49	0.57	0.50	0.58	0.49	0.58	0.60	0.56	0.64	0.57	0.63	0.55	0.61
	50	0.56	0.50	0.60	0.50	0.58	0.49	0.58	0.69	0.54	0.62	0.56	0.63	0.54	0.61
	75	0.59	0.50	0.59	0.52	0.57	0.52	0.57	0.66	0.52	0.67	0.56	0.63	0.53	0.61
	100	**0.61**	0.50	0.58	0.53	0.57	0.50	0.56	**0.65**	0.57	0.62	0.57	0.62	0.57	0.62

Source: Created by the author

APPENDIX E – VISUALIZATION OF THE EVALUATION SCENARIOS FOR SYNCOPATE IN THE AOPE TASK

Evaluation scenarios for the performance of the Synchronous Double-channel Recurrent Network (SDRN) and the Multi-Task TSMSA models when trained on variations of the SemEval Datasets: `14lap`, `14res`, `15res`, and `16res`.

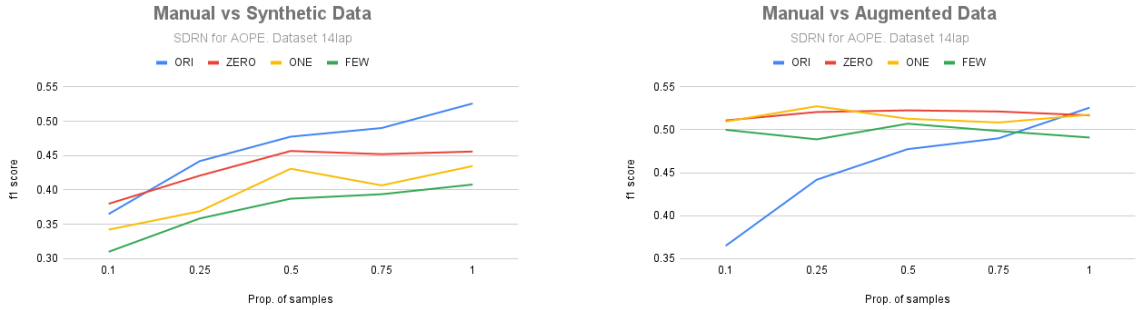
Evaluation Scenarios:

1. Training the models only on manually labeled data (ORI);
2. Training the models only on synthetic opinionated triples (ZERO, ONE, and FEW);
3. Training the models on manually built data augmented with synthetic opinionated triples (ZERO, ONE, and FEW).

E.1 PERFORMANCES OF THE SDRN MODEL

Figure 25 – Performance of the SDRN model on variations of the `14lap` dataset

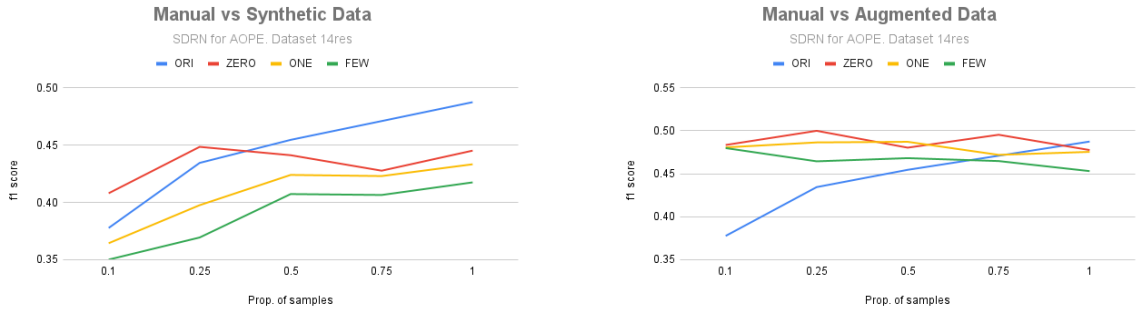
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



Source: Created by the author

Figure 26 – Performance of the SDRN model on variations of the `14res` dataset

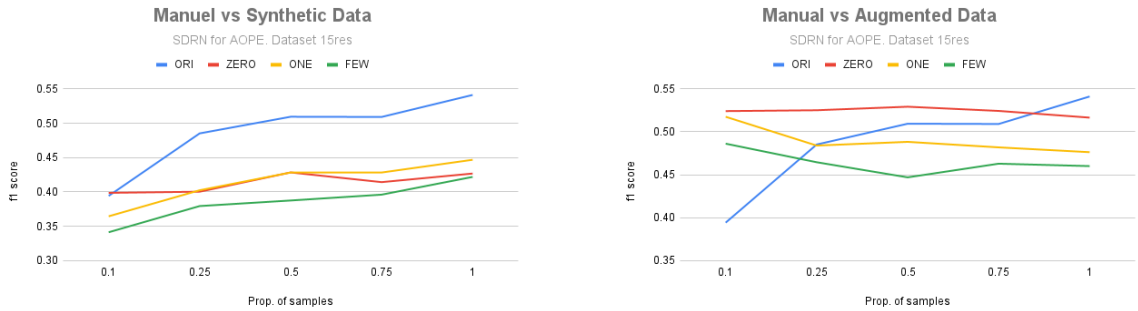
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



Source: Created by the author

Figure 27 – Performance of the SDRN model on variations of the `15res` dataset

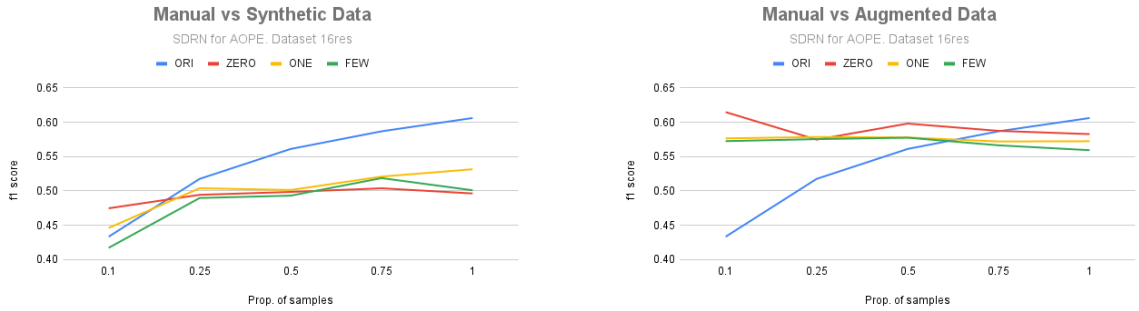
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



Source: Created by the author

Figure 28 – Performance of the SDRN model on variations of the `16res` dataset

(a) Manual *versus* Synthetic (b) Manual *versus* Augmented

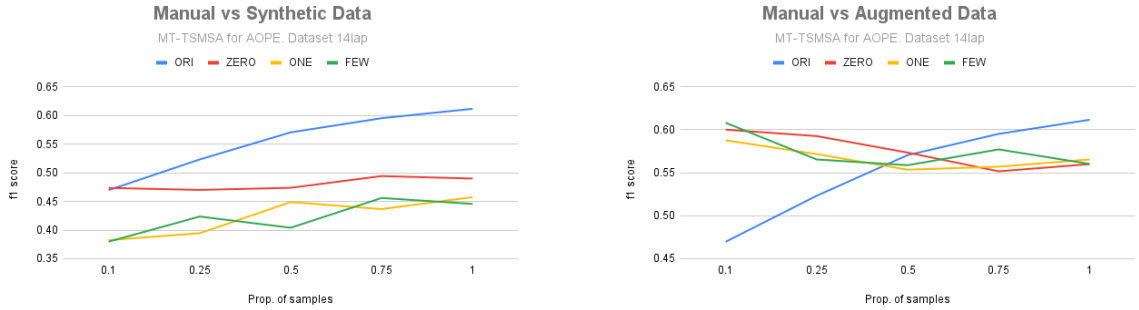


Source: Created by the author

E.2 PERFORMANCES OF THE MT-TSMSA MODEL

Figure 29 – Performance of the MT-TSMSA model on variations of the `14lap` dataset

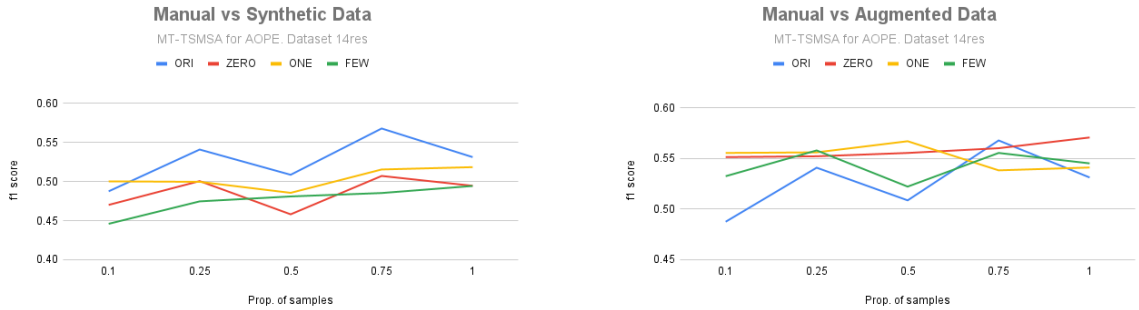
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



Source: Created by the author

Figure 30 – Performance of the MT-TSMSA model on variations of the `14res` dataset

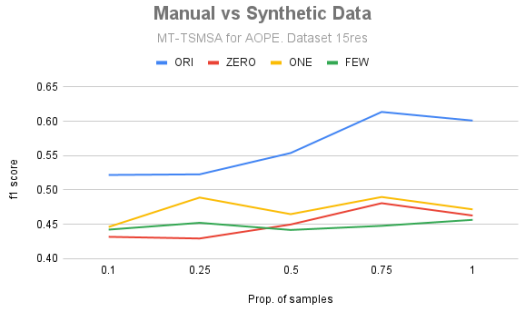
(a) Manual *versus* Synthetic (b) Manual *versus* Augmented



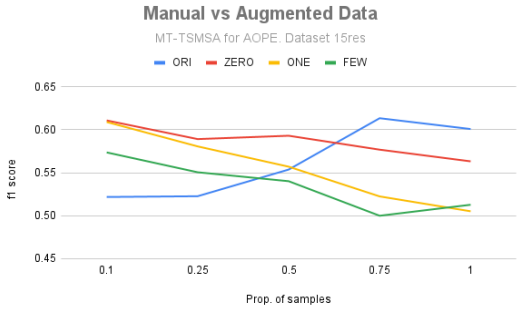
Source: Created by the author

Figure 31 – Performance of the MT-TSMSA model on variations of the `15res` dataset

(a) Manual *versus* Synthetic



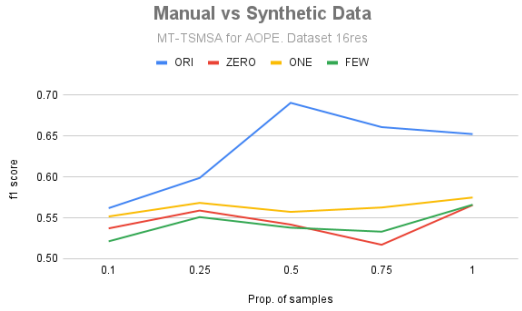
(b) Manual *versus* Augmented



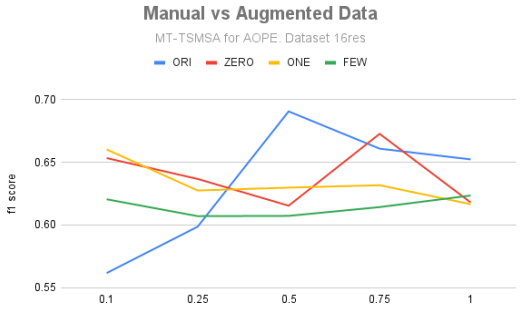
Source: Created by the author

Figure 32 – Performance of the MT-TSMSA model on variations of the `16res` dataset

(a) Manual *versus* Synthetic



(b) Manual *versus* Augmented



Source: Created by the author

APPENDIX F – INSTRUCTION GIVEN TO MANUAL RATERS

The three manual raters were voluntarily recruited. Hence, no payment was performed for participating in the evaluation. All three evaluators are non-native English speakers and present an intermediate to an upper-intermediate level of fluency in the language. All of them are Ph.D. Students in the area of Computational intelligence but without any involvement in this work.

We instructed the human raters about the meaning of each evaluated element: what is an opinionated sentence, an aspect, an opinion, and a relation between them. The instructions are shown below:

Opinionated Sentence The sentence must present an opinion about a *restaurant*'s characteristics or about a *laptop*.

Aspect A business, product, or service characteristic about which can be assigned some opinion.

Opinion Words One or multiple words denoting a positive, negative or neutral sentiment about the characteristic of a business, product, or service.

Relation: aspect and opinion words There is an explicit relation between opinionated words and the mentioned aspect.

Along with these definitions, we gave some already evaluated examples to anchor the concepts to the evaluators. These examples are shown in Table 26. The questions are: *Q1*) Is the sentence opinionated? *Q2*) Is the extracted aspect an aspect of a service or a product? *Q3*) Do the extracted opinion terms represent an opinion or sentiment? *Q4*) Are the extracted aspect and opinion terms related?

We instruct the raters to answer the three last questions only if the sentence is opinionated, i.e., if the first question is "YES".

Table 26 – Evaluation examples given to the human raters in order to guide their independent evaluation.

OPINIONATED SENTENCE	**ASPECT**	**OPINION WORD(S)**	**Q1**	**Q2**	**Q3**	**Q4**
Easy to start up and does not overheat as much as other laptops .	start up	Easy	YES	YES	YES	YES
This is an amazing place to try some roti rolls .	roti rolls	try	YES	YES	YES	YES
Fresh ingredients and everything is made to order .	ingredients	Fresh	YES	YES	YES	YES
The manager also left a bottle of wine in the bill as well.	manager	left	NO			
The manager also left a bottle of wine in the bill as well.	bottle of wine	left	NO			
It's very nice and the food is decent and fast.	food	decent fast	YES	YES	YES	YES
They serve rice and their drinks are huge.	rice	huge	YES	YES	YES	NO
They serve rice and their drinks are huge.	drinks	huge	YES	YES	YES	YES
We were sitting in an empty lot with a table.	table	empty	NO			
The battery life is great for movies, webpages, and other things.	battery life	great	YES	YES	YES	YES
The battery life is great for movies, webpages, and other things.	webpages	great	YES	YES	YES	NO

Source: Created by the author