An adaptive hybrid deep learning architecture for sentiment analysis-based recommendations on social networks

Doctoral Thesis

Ph.D. PROGRAM IN COMPUTER ENGINEERING

University of Salamanca

VNiVERSiDAD D SALAMANCA

Cach N. Dang

Supervisors

María N. Moreno García

Fernando De la Prieta Pintado

Salamanca, 2021

UNIVERSITY OF SALAMANCA

DOCTORAL THESIS

An adaptive hybrid deep learning architecture for sentiment analysis-based recommendations on social networks

---

Una arquitectura adaptativa e híbrida de *deep learning* para recomendaciones basadas en análisis de sentimientos en redes sociales

---

Author: Cach N. Dang

Supervisors: María N. Moreno García and Fernando De la Prieta Pintado

Salamanca, 2021

Statement of Authorship

Cach N. Dang presents the thesis work entitled “An adaptive hybrid deep learning architecture for sentiment analysis-based recommendations on social networks” to apply for the Doctorate Degree in Computer Engineering of the University of Salamanca, and states that it has been carried out under the supervision of Dr. María N. Moreno, Full Professor in the Department of Computer Science and Automation of the University of Salamanca, and Dr. Fernando De la Prieta Pintado, Associate Professor in the Department of Computer Science and Automation of the University of Salamanca.

Author:

Cach N. Dang

Supervisors:

MORENO GARCIA MARIA NAVELONGA - 07843490F

Firmado digitalmente por MORENO GARCIA MARIA NAVELONGA - 07843490F
Fecha: 2021.09.23 08:48:39 +02'00'

María N. Moreno García

DE LA PRIETA PINTADO FERNANDO - 71011007W

Firmado digitalmente por DE LA PRIETA PINTADO FERNANDO - 71011007W
Fecha: 2021.09.23 09:21:27 +01'00'

Fernando De la Prieta Pintado

Acknowledgments

I still do not believe that my Ph.D. has entered its last stage, which is the presentation of this thesis. I have had tremendously fruitful experiences, including both the struggle to find technical solutions for improving an SA model and the joy of knowing that I have reached my goal to share via journal articles the results of the model. It has been a long road for me since I started to write my Ph.D. proposal. I am grateful for the profound experiences I had on my path to attaining a doctoral degree and appreciative of the support I received from my supervisors, my family, my colleagues, and my friends.

First and foremost, I bow down to enormous generosity, deep care and wise guidance, kind patience, and support from my supervisors María N. Moreno García and Fernando De la Prieta. Without the guidance of Professor García, I would not have achieved my academic goals. She always provided thoughtful, detailed comments about my ideas, my writing, and my decisions. Those comments sometimes arrived very late at night during weekdays or during early mornings on the weekend, evidence of the high degree of support Dr. Garcia offered me and the faith she placed in the value of my work. The path to obtaining a doctoral degree is long and arduous. Their serious reminders last year were so effective for me to make a strong determination to complete this work. The continuous encouragement offered by Professor Garcia and Professor De la Prieta bolstered my resilience when I lost my hope of completing work on my Ph.D., as well as during these two special years of COVID. Moreover, their questions helped me to critically consider the technical or conceptual issues needed to improve an algorithm and respond to the comments by peer reviewers. They challenged me to write critically, clearly, and thoroughly, and their corrections helped me improve my writing skills every day. I am deeply thankful for their acceptance and guidance. I do believe this skill is very crucial for my research and teaching career now and will continue to be helpful in the future. I am so grateful to Professor De la Prieta, who motivated me to pursue my Ph.D. His advice carried so much positive energy in it, which nurtured me along this path.

From a personal level to an organizational level, I was fortunate to have physical support from both universities: The University of Salamanca and The Ho Chi Minh City University of Transport. The cooperation of both universities and the agreement between Professors Juan M. Corchado and Professor Kien X. Dang allowed me to enroll in this Ph.D. course of study. During these years

of study, Dr. Anh V.Q. Le, Dean of the Department of Information Technology, and Dr. Son N. Vo, Deputy Director of Institute of Fundamental and Applied Sciences, always provided the necessary space to work and made scheduling accommodations that allowed me both to work and to study. I am thankful for this opportunity and for these educational leaders and professors whose efforts and hopes are improving opportunities for young researchres and creating a more robust system of education in Vietnam. Even though I took a longer time than is typical to finish my Ph.D., the experience taught me to keep calm and keep working while maintaining my aspiration for a brighter future for coming generations.

I also had additional support from my English teacher Thu-Huong Luong who shared with me the most effective ways to express my thoughts in written English; my students Mr. Kien V. Nguyen and Mr. An T. Nguyen, who are curious and enthusiastic to learn about SA modeling; and my colleagues and friends, who have always been there for me no matter the up or down moments during my studies. I admit that I have been lucky to have had their support throughout the process.

Finally, I cannot sufficiently express in words my deep gratitude to my parents, my wife, and my two daughters who have been side-by-side with me for every single step I take and every thought I express. The home of my parents is like a sacred island for me to come back to take a rest and to recharge my battery. I really appreciate that they allowed me to lean on them when I was tired, although I am grown-up. They reminded me to shift my attention to my health and my family whenever I lost my work-life balance, and they always willingly cared for my children during the time my wife went overseas to complete her Ph.D. I cannot imagine myself sitting at my living room table today and writing this acknowledgment without the deep understanding kind patience, and emotional support of my whole family. Thinking about it brings tears to my eyes. Their acceptance is everything for me. I do wish the completion of this work, my Ph.D. study, is a reward not only for me but also for each member of my family. I send hugs and love to each one of them.

My course of study has been a blessing. Two hands join and bow down for each person who came to my life during my Ph.D. studies and gave me a hand, the hand of guidance, of support physically, emotionally, mentally, and intellectually. I am deeply grateful for this opportunity and wish to continue moving forward and to do the same for others.

Abstract

With the explosion of Web 2.0 and the rise of blogs, forums, and online social networks, different opinions about a particular topic can be easily found from millions of users on these websites. For example, users discuss current experiences, share their points of view on specific facts, and offer praise or complaints about specific products they have just bought. This kind of information plays a key role in various applications, such as to track comments or reviews of customers for recommender systems, and to analyze surveys that an organization itself conducts. The problem of automatically extracting opinions from online user-generated texts, known as “opinion mining” or “sentiment analysis,” has been a growing research topic recently.

The study of public opinion can provide us with valuable information. The analysis of sentiment on social networks has become a powerful means of learning about the users’ opinions and has a wide range of applications. However, the efficiency and accuracy of sentiment analysis is being hindered by the challenges encountered in natural language processing. In recent years, it has been demonstrated that deep learning models are a promising solution to these challenges. In addition, sentiment analysis can be beneficial to recommender systems. As well, a growing body of research examines how sentiment analysis is being applied in recommender systems. Such an analysis can improve the understanding of users’ attitudes, opinions, and emotions, which is beneficial to integrate it into recommender systems for achieving higher recommendation reliability.

Social media data has been exploited in different ways to address some problems, especially those associated with Collaborative Filtering (CF) methods. Sparsity and gray-sheep problems are two of the main reasons CF methods do not provide the reliability required in some recommender systems. In particular, when only sparse ratings data is available to a recommender system, sentiment analysis can play a key role in improving recommendation quality. This is due to the fact that recommendation algorithms mostly rely on users’ ratings on items to select the items to recommend. Such ratings are usually insufficient and very limited. On the other hand, sentiment ratings of items that can be derived from online news services, blogs, social media, or even from the recommender systems themselves are seen as capable of providing better recommendations to users. Sentiment-based models have been exploited in recommender systems to overcome the data-sparsity problem that exists in conventional recommender systems.

This thesis addresses that gap by means of a comprehensive comparison of sentiment analysis methods in the literature and by an experimental study to evaluate the performance of deep learning models and related techniques on datasets about different topics. The research question aims to determine whether it is possible to present outperforming methods for multiple types and sizes of datasets. Another question raised in this thesis is whether hybrid models perform better than single models regardless of the characteristics of the datasets. Therefore, the aim of our work is the proposal of hybrid models and the study of their behavior

with different types of datasets from different domains. Then, we present an approach to use sentiment analysis in recommender systems, in wich user opinions and explicits ratings are combined to provide recomendations. This application is based on an adaptive recommender system architecture, some techniques for feature extraction, and deep learning models for sentiment analysis. Hence, integrating sentiment in recommender systems may significantly enhance the recommendation quality of these systems.

We applied deep learning models with TF-IDF and word embedding to eight datasets, including tweets and reviews. We implemented the state-of-the-art sentiment analysis approaches based on deep learning, and combined models to increase the accuracy of sentiment analysis. In addition, we evaluated methods for integrating sentiment analysis into recommender systems on popular public review datasets. The experimental results show that the proposed approach significantly improves the recommender system performance.

Resumen

Con la explosión de la Web 2.0 y el auge de blogs, foros y redes sociales en línea, millones de usuarios de estos sitios web pueden encontrar fácilmente diferentes opiniones sobre un tema en particular. Por ejemplo, los usuarios discuten experiencias actuales, comparten sus puntos de vista sobre hechos específicos y ofrecen elogios o quejas sobre productos específicos que acaban de comprar. Este tipo de información juega un papel clave en diversas aplicaciones, como rastrear comentarios o reseñas de clientes para usarlos en sistemas de recomendación, y analizar encuestas que la propia organización realiza. El problema de extraer automáticamente opiniones de textos generados por usuarios en línea, conocido como "minería de opiniones" o "análisis de sentimientos", ha sido un tema de investigación creciente en los últimos años.

El estudio de la opinión pública puede aportarnos información valiosa. El análisis del sentimiento en las redes sociales se ha convertido en un poderoso medio para conocer las opiniones de los usuarios y tiene un amplio abanico de aplicaciones. Sin embargo, la eficiencia y precisión del análisis de sentimientos se ve obstaculizada por los desafíos encontrados en el procesamiento del lenguaje natural. En los últimos años, se ha demostrado que los modelos de aprendizaje profundo son una solución prometedora a estos desafíos. Además, el análisis de sentimientos puede ser beneficioso para mejorar los sistemas de recomendación. Por otra parte, un creciente cuerpo de investigación examina cómo se está aplicando el análisis de sentimientos en estos sistemas. Dicho análisis puede mejorar la comprensión de las actitudes, opiniones y emociones de los usuarios, lo cual es beneficioso para su integración en sistemas de recomendación y así lograr una mayor fiabilidad de las recomendaciones.

Los datos de las redes sociales se han aprovechado de diferentes formas para abordar algunos problemas, especialmente los asociados con los métodos de Filtrado Colaborativo (FC). El problema de la dispersión y el de la oveja negra son dos de las principales razones por las que los métodos de FC no proporcionan la fiabilidad requerida en algunos sistemas de recomendación. En particular, cuando hay pocos datos de valoraciones de ítems disponibles en un sistema de recomendación, el análisis de opiniones puede desempeñar un papel clave en la mejora de la calidad de las recomendaciones. Esto se debe al hecho de que los algoritmos de recomendación se basan principalmente en las valoraciones de los ítems por parte de los usuarios para seleccionar los ítems que se recomendarán. Estas valoraciones suelen ser insuficientes y muy limitadas. Por otro lado, se considera que las valoraciones de los ítems derivadas del análisis de sentimiento a partir de servicios de noticias en línea, blogs, redes sociales o incluso de los propios sistemas de recomendación, son capaces de brindar mejores recomendaciones a los usuarios. Los modelos basados en sentimientos se han aprovechado en sistemas de recomendación para superar el problema de escasez o dispersión de datos que existe en los sistemas de recomendación convencionales.

Esta tesis aborda esa brecha mediante una comparación exhaustiva de los métodos de análisis de sentimientos en la literatura y mediante un estudio experimental para evaluar el rendimiento

de modelos de *deep learning* y técnicas relacionadas en conjuntos de datos sobre diferentes temas. La pregunta de investigación que se formula tiene como objetivo determinar si es posible presentar métodos altamente fiables para múltiples tipos y tamaños de conjuntos de datos. Otra cuestión que se plantea en esta tesis es si los modelos híbridos funcionan mejor que los modelos individuales independientemente de las características de los conjuntos de datos. Por tanto, el objetivo de nuestro trabajo es la propuesta de modelos híbridos y el estudio de su comportamiento con diferentes tipos de conjuntos de datos de diferentes dominios. Adicionalmente, presentamos un enfoque de uso del análisis de sentimientos en sistemas de recomendación, en el cual las opiniones de los usuarios y sus valoraciones explícitas se combinan para proporcionar las recomendaciones. Esta aplicación se basa en una arquitectura de sistema de recomendación adaptativa, algunas técnicas para la extracción de características y modelos de *deep learning* basados en el análisis de sentimientos. Por lo tanto, integrar el sentimiento en los sistemas de recomendación puede mejorar significativamente la calidad de recomendación de estos sistemas.

Aplicamos modelos de aprendizaje profundo con TF-IDF e incrustación de palabras en ocho conjuntos de datos, incluidos tweets y reseñas. Así mismo, implementamos los enfoques de análisis de sentimientos de última generación basados en *deep learning* y modelos combinados para aumentar la precisión del análisis de sentimientos. Además, evaluamos métodos para integrar el análisis de opiniones en sistemas generales de recomendación para servicios de *streaming* con populares conjuntos de datos públicos de revisiones. Los resultados muestran que el enfoque propuesto mejora significativamente la fiabilidad de los sistemas de recomendación.

Contents

1. Modality of the thesis

The presentation of this doctoral thesis at the University of Salamanca is done in the format of a compendium of previously published articles. The thesis includes four contributions: three articles published in journals and a book chapter.

1.1. Surpervisors authorization

María N. Moreno García, Full Professor in the Department of Computer Science and Automation of the University of Salamanca, and Fernando De la Prieta Pintado, Associate Professor in the Department of Computer Science and Automation of the University of Salamanca, both supervisors of the doctoral thesis of Mr. Cach N. Dang,

AUTHORIZE

To Mr. Cach N. Dang to present and defend his doctoral thesis in the form of a compendium of articles.

MORENO GARCIA MARIA NAVELONGA - 07843490F
Firmado digitalmente por MORENO GARCIA MARIA NAVELONGA - 07843490F
Fecha: 2021.09.23 08:49:26 +02'00'

María N. Moreno García

DE LA PRIETA PINTADO FERNANDO - 71011007W
Firmado digitalmente por DE LA PRIETA PINTADO FERNANDO - 71011007W
Fecha: 2021.09.23 09:22:11 +01'00'

Fernando De la Prieta Pintado

1.2. List of contributions

The following is a list of published articles with their reference data and quality indexes.

CONTRIBUTION 1

Dang, Nhan Cach, María N. Moreno-García, and Fernando De la Prieta. "Sentiment analysis based on deep learning: A comparative study", *Electronics* 9, no. 3 (2020): 483.

DOI: 10.3390/electronics9030483

Authors:

- Cach N. Dang, University of Salamanca, Spain.
- Fernando De la Prieta, University of Salamanca, Spain.
- María N. Moreno García, University of Salamanca, Spain.

Journal: Electronics (ISSN: 2079-9292)

Quality indexes:

- **WoS JCR[1] Impact Factor 2020**: 2.397. **Rank**: 173/307 (Q3). **Area**: Computer Networks and Communications.
- **SCOPUS Cite Score 2020**: 2.7. **Rank**: 161/334 (Q2). **Area**: Computer Science.

CONTRIBUTION 2

Dang, Cach N., María N. Moreno-García, and Fernando De la Prieta. "Hybrid Deep Learning Models for Sentiment Analysis", *Complexity* 2021 (2021).

DOI: 10.1155/2021/9986920

Authors:

- Cach N. Dang, University of Salamanca, Spain.
- Fernando De la Prieta, University of Salamanca, Spain.
- María N. Moreno García, University of Salamanca, Spain.

Journal: Complexity (ISSN: 1099-0526)

Quality indexes:

- **WoS JCR Impact Factor 2020**: 2.862, **Rank**: 28/106 (Q2). **Area**: Mathematics-Interdisciplinary Applications.
- **SCOPUS Cite Score 2020**: 3.3. **Rank**: 64/226 (Q1). Area: General Computer Science.

[1] Web of Science Journal Citation Reports

CONTRIBUTION 3

Dang, Cach N., María N. Moreno-García, and Fernando De la Prieta. "An Approach to Integrating Sentiment Analysis into Recommender Systems", *Sensors* 21.16 (2021): 5666.

DOI: 10.3390/s21165666

Authors:

- Cach N. Dang, University of Salamanca, Spain.
- Fernando De la Prieta, University of Salamanca, Spain.
- María N. Moreno García, University of Salamanca, Spain.

Journal: Sensors (ISSN: 1424-8220)

Quality indexes:

- **WoS JCR Impact Factor 2020**: 3.576. Rank: 32/91 (Q2). **Area**: Engineering.
- **SCOPUS Cite Score 2020**: 5.8. **Rank**: 69/329 (Q2). **Area**: Computer Science.

CONTRIBUTION 4

Dang, Nhan Cach, Fernando De la Prieta, Juan Manuel Corchado, and María N. Moreno. "Framework for retrieving relevant contents related to fashion from online social network data." In International Conference on Practical Applications of Agents and Multi-Agent Systems, pp. 335-347. Springer, Cham, 2016.

DOI: 10.1007/978-3-319-40159-1_28

Authors:

- Cach N. Dang, University of Salamanca, Spain.
- Fernando De la Prieta, University of Salamanca, Spain.
- Juan Manuel Corchado, University of Salamanca, Spain.
- María N. Moreno García, University of Salamanca, Spain.

Book: Trends in Practical Applications of Scalable Multi-Agent Systems, the PAAMS Collection (ISBN: 978-3-319-40159-1)

Series: Advances in Intelligent Systems and Computing (ISSN: 2194-5357)

Quality indexes:

- **SCOPUS Cite Score 2016**: 0.7. **Rank**: 144/197 (Q3). **Area**: General Computer Science.

2. Overview of the contributions

2.1. Introduction

We are living in the "Age of Big Data", where the amount of data in the world has been exploding. Blogs, forums, and online social networks are sources of differing opinions about a particular topic and can be easily found from millions of users on these websites. For example, users discuss their current experiences, share their points of view on specific facts, and offer praise or complaints about specific products they have just bought. This kind of information plays a key role in various applications, such as to track comments or reviews of customers for recommender systems and to analyze surveys that an organization itself conducts. Sentiment analysis and recommender systems are areas of significant focus in academic research and commercial development in this "Age of Big Data".

Sentiment analysis is a process of extracting information about an entity and automatically identifying any of the subjectivities of that entity. The objective is to determine whether text generated by users conveys their positive, negative, or neutral opinions. Sentiment classification can be carried out on three levels of extraction: the aspect or feature level; the sentence level; and the document level. Three approaches currently exist to address the problem of sentiment analysis [1]: lexicon-based techniques; machine-learning-based techniques; and hybrid approaches.

Although much work has been done in sentiment analysis, there are still many challenges to be addressed, including improving model reliability, reducing processing time, and applying techniques developed for specific types of data and specific data domains [2].

In recent years, deep learning models have been extensively applied in the field of sentiment analysis, where their great potential has been proved. Both shallow neural networks and deep neural networks are capable of approximating any function. But when contrasted to shallow neural networks, deep neural networks have the advantage in feature extraction in the process of learning on large datasets. This is primarily because the deep models are able to extract/build better features than shallow models, using the intermediate hidden layers to achieve this [3, 4]. For the same level of accuracy, deep neural networks (DNN) can be much more efficient in terms of computation and number of parameters. DNN are able to create deep representations; at every layer, the network learns a new, more abstract representation of the input.

Although a single machine learning method is relatively reliable when applied within certain domains, each deep learning approach has its own advantages and disadvantages. Long Short-Term Memory (LSTM) [5] normally yields better results but requires more processing time than Convolutional Neural Networks (CNN) [6], and CNN requires fewer hyper parameters and less supervision. Meanwhile, the LSTM performs more accurately for long sentences but requires a longer time to process [2].

The approach of combining two (or more) methods is introduced [7-9] as a means of incorporating the advantages of both methods while inheriting the disadvantages of neither. A hybrid system with collaborative functions, therefore, is better able to address potential pitfalls, if any exist, associated with one single system. The effectiveness of the integrated models may vary based on different tasks. The CNN enhanced by Support Vector Machines (SVM) [10-12], CNN with recursive neural networks (RNN) [13-16], Lexicon-based analysis with machine learning [17, 18] showed an enhanced result. The combination of CNN, LSTM, and SVM aims to take advantage of the two deep network architecture models and SVM algorithms when performing sentiment analysis on different domains and types of datasets. Moreover, different types of input data are obtained from social networks, such as tweets, reviews, and so on. Within and across these types, the input data also contains differences, e.g., the distribution of the lengths of the tweets and reviews, the diversity of topics in each dataset, the sample size, as well as the greater or lesser presence of explicit sentiments and irrelevant information. Some approaches may be unable to perform well in different domains, with inadequate accuracy and performance in sentiment analysis [2, 19]. As a result, it is difficult to apply to several types of input data.

Recommender systems intend to provide personalized recommendations about products or services to support decision-making in the continuous increase of online information. Recommender systems have expanded widely in recent decades especially in three main domains (business, government, and education) across eight categories (e-government, e-business, e-commerce/e-shopping, e-library, e-learning, e-tourism, e-resource services and e-group activities) [20]. E-commerce has widely applied recommender systems to suggest additional products for customers to choose from among the multiple products available. For example, Amazon uses this system to suggest preferred products for customers, YouTube uses it to suggest related videos on the auto play function, and Facebook uses it to recommend people and web pages to connect and follow.

The most common methods used in recommender systems may be grouped into three categories: content-based; collaborative filtering (CF); and hybrid recommender systems [21, 22]. These techniques vary depending on the types of social media data that are used. Lu et al. [20] analyzed typical recommender systems and effectively identified the specific requirements for recommendation techniques in the domain. This work also directly motivates and supports researchers and practitioners to promote the popularization and application of recommender systems in different domains.

Sentiment analysis can be beneficial to recommender systems. A sample of this can be found in the work of Preethi et al. [23], in which a cloud-based recommender system uses RNN to analyze sentiments of reviews in order to improve and validate restaurant and movie recommendations. Along with behavioral analysis, sentiment analysis is also an efficient tool for commodity markets [24].

Social network data has been exploited in different ways to address some problems, especially associated with collaborative filtering methods [25]. Sparsity and gray-sheep problems are two of the main reasons collaborative filtering methods do not provide the reliability required in some recommender systems [26]. In particular, when only sparse ratings data is available, sentiment analysis can play a key role in improving recommendation quality. This is due to the

fact that recommendation algorithms mostly rely on users' ratings to select the items to recommend. Such ratings are usually insufficient and very limited. On the other hand, sentiment ratings of items that can be derived from online news services, blogs, social media or even from the recommender systems themselves are seen as capable of providing better recommendations to users. Sentiment-based models have been exploited in recommender systems to overcome the data-sparsity problem that exists in conventional recommender systems.

In comparative sentiment analysis studies, most papers focus on reliability metrics, such as overall accuracy or F-score, and leave out processing time. In addition, the evaluations of the models are conducted on a small number of datasets. This research addresses that gap by means of a comprehensive comparison of sentiment analysis methods in the literature, and an experimental study to evaluate the performance of deep learning models and related techniques on datasets about different topics. Our research question aims to determine whether it is possible to present outperforming methods for multiple types and sizes of datasets. We build upon on previous studies of improvement of sentiment analysis performance by evaluating the results from the viewpoint of a combination of three criteria: overall accuracy, F-score, and processing time. The purpose of this comparative study is to give an objective overview of different techniques that can guide researchers towards the achievement of better results.

An additional question raised in this thesis is whether hybrid models perform better than single models regardless of the characteristics of the datasets. Therefore, another objective of our work is the proposal of hybrid models and the study of their behavior with different types of datasets from different domains. So we evaluated and validated the combination of three models — CNN, LSTM and SVM — considering the relationship between models and their advanced capacities to extract characteristics, to store past information and nodes, and to classify text. Then, we exploited the sentiment analysis output into recommender systems in an approach that combines user opinions and explicit ratings. This application is based on an adaptive recommender system architecture, some techniques for feature extraction, and deep learning models based on sentiment analysis. The study proves that integrating sentiment into recommender systems may significantly enhance the recommendation quality of these systems.

We experimented by applying deep learning models with TF-IDF and word embedding (word2vec and BERT) to eight datasets, including tweets and reviews. We implemented the state-of-the-art sentiment analysis approaches based on deep learning and constructed hybrid models to increase sentiment analysis accuracy. Methods integrating sentiment analysis into recommender systems were evaluated on popular public review datasets. The experimental results show that the proposed approach significantly improves the performance of recommender systems.

2.2. Objectives

The main objective of the thesis presented here is to propose hybrid deep learning models to improve the sentiment analysis on social network data and find methods for creating performant recommender systems. This work focused on different types of datasets from several domains. The purpose is to integrate sentiment analysis into recommender systems that combine hybrid deep learning models of sentiment analysis, collaborative filtering methods to improve recommendations. The specific objectives are as follows:

- Studying popular deep learning models with TF-IDF and word embedding applied to social network data and implementing state-of-the-art sentiment analysis approaches based on deep learning.
- Proposing hybrid deep learning models to increase the accuracy of sentiment analysis in comparison to single models on all types of datasets.
- Integrating sentiment analysis on reviews and collaborative filtering in recommender systems in order to improves the system's performance.

2.3. State of the art

2.3.1. Sentiment analysis problems

Sentiment analysis can be performed on three levels of extraction: the sentence level; the document level; and the aspect or feature level. It is a process of extracting information about an entity and automatically identifying any of the subjectivities of that entity. The aim is to determine whether text generated by users conveys their positive, negative, or neutral opinions. Three approaches currently exist to address the problem of sentiment analysis [1]: lexicon-based techniques; machine-learning-based techniques; and hybrid approaches. Lexicon-based techniques are divided into two approaches: dictionary-based and corpus-based [27]. They were the first to be used for sentiment classification. Machine learning-based methods [28] that have been proposed for sentiment analysis include traditional and deep learning techniques. The hybrid approaches combine machine learning and lexicon-based approaches [29]. Sentiment lexicons regularly play a key role in most of these strategies.

Deep learning techniques can provide better results than traditional techniques. Different kinds of deep learning models can be used for sentiment classification, including Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), and recursive neural networks (RNN). These models address classification problems at the document level, sentence level, or aspect level. In addition, some approaches that combine two models are introduced [10-16]. The CNN enhanced by Support Vector Machines (SVM) [10-12], CNN with RNN [13-16] showed improved results.

Many researchers began evaluating this trend in 2015. Tang et al. [30] introduced techniques based on deep learning approaches for several sentiment analysis, such as learning word embedding, sentiment classification, and opinion extraction. Zhang and Zheng [28] discussed machine learning for sentiment analysis. Both research groups used part of speech (POS) as a text feature and used TF-IDF to calculate the weight of words for the analysis. Sharef et al. [31] discussed the opportunities of sentiment analysis approaches for big data. In papers [32-34], the latest deep-learning-based techniques (namely CNN, RNN, and LSTM) were reviewed and compared with each other in the context of sentiment analysis problems.

Some other studies applied deep-learning-based sentiment analysis in different domains, including finance [35, 36], weather-related tweets [37], trip advisors [38], recommender systems for cloud services [23], and movie reviews [32, 33, 39-42]. In [37], where text features were automatically extracted from different data sources, user information and weather knowledge were transferred into word embedding using the Word2vec tool. The same techniques have been used in several works [34, 35]. Jeong et al. [43] identified product development opportunities by

combining topic modeling and the results of a sentiment analysis that had been performed on customer-generated social media data. It has been used as a real-time monitoring tool for analysis of changing customer needs in rapidly evolving product environments. Pham et al. [38] used multiple layers of knowledge representation to analyze travel reviews and determine sentiments for five aspects, including value, room, location, cleanliness, and service [38]. Another approach [44] combines sentiment and semantic features in an long short-term memory model based on emotion detection. Preethi et al. [23] applied deep learning to sentiment analysis for a recommender system in the cloud using the food dataset from Amazon. For the health domain, Salas-Zárate et al. [27] applied an ontology-based, aspect-level sentiment analysis method to tweets about diabetes.

Sentiment polarity analysis using deep learning models on tweets data was found in [29, 45-49]. The authors described how they used deep learning models to increase the accuracy of their respective sentiment analysis. Most of the models are used for content written in English, but there are a few that manage tweets in other languages, including Spanish [50], Thai [47], and Persian [51]. Previous researchers have analyzed tweets by applying different models of polarity-based sentiment deep learning. Those models include DNN [49], CNN [46], and hybrid approaches [29].

Other works using neural network models are focused not only on the sentiment polarity of textual content, but also on aspect sentiment analysis [27, 36, 38, 52-54]. Salas-Zárate et al. [27] used semantic annotation (diabetes ontology) to identify aspects from which they performed aspect-based sentiment analysis using SentiWordNet. Pham et al. [38] included the determination of sentiment ratings and importance degrees of product aspects. A novel, multilayer architecture was proposed to represent customer reviews aiming at extracting more effective sentiment features.

In addition, the hybrid models can increase the accuracy for sentiment analysis in comparison to a single model performance. There are many ways to build hybrid models. In [10-12], the authors combined a CNN model and SVM that can improve the accuracy of image recognition. A convolutional network layer is used for extracting features and SVM functions as a recognizer. Original CNN is used with Softmax functions. Srinidhi et al. [55] proposed a hybrid model that combined LSTM and SVM with a radial basis function kernel for the textual classification of positive and negative sentiments. The hybrid model was evaluated on the IMDB movie review datasets. These models are combined from single deep learning models with SVM for classification. Some of them are applied for image recognition.

Akhtar et al. [56] built a hybrid deep learning architecture which is highly efficient for sentiment analysis in resource-poor languages. They used CNN for learning sentiment embedded vectors and SVM for sentiment classification. The model was tested on four Hindi datasets covering varied domains. Vo et al. [15] used a multi-channel LSTM-CNN model for sentiment analysis on reviews/comments from e-commerce sites. In addition, hybrid CNN – LSTM models are applied for sentiment analysis on movie reviews by Rehman et al. [14]. The same techniques are used in several works, e.g., [13, 57-59]. Harleen et al. [60] designed an algorithm called a Hybrid Heterogeneous Support Vector Machine (H-SVM). They performed sentiment analysis on Twitter data related to COVID-19. Zenun et al. [61] employed three different deep learning models such

as: CNN, LSTM and CNN-LSTM for classifying Facebook comments related to the COVID-19 pandemic. They used a pre-trained word embedding method called FastText (an extension to Word2vec proposed by Facebook in 2016) and a contextualized word embedding model—BERT (published by researchers at Google AI Language in 2018 [62]) to learn and generate word vectors. Both researchers scored tweets/comment as positive, negative, or neutral. However, these models were individually tested on different datasets in a particular domain or tested on few sample datasets. Therefore, their validity is not generally proven.

A study by Jnoub et al. [63] focused on providing a generalized model for sentiment analysis that combined CNN with their own algorithm to transform reviews to vectors. The model was evaluated on three different datasets: IMDB, Movie Reviews, and their own dataset collected from Amazon reviews. Ombabi et al. [64] proposed a hybrid deep learning model that combines CNN and LTSM. In addition, FastText is used for word embedding and SVM for classification in the Arabic language. In our work, both Word2vec and BERT were applied for word embedding. We proposed four types of hybrid deep learning models based on CNN, LSTM and SVM for classifying both tweets and reviews.

Furthermore, other studies combine Lexicon-based analysis with machine learning [17, 18], or sentiment lexicons and Polarity Shifting Devices [65]. The research by Sánchez-Rada et al. [9] deals with the problem of user and content sentiment classification. They proposed a hybrid model that merges features from different levels of social context. The model is evaluated in different datasets. A study from Wang et al. [66] presented a hybrid approach in which sentiment analysis of reviews about movies is used to improve a preliminary recommendation list obtained from the combination of collaborative filtering and content-based methods. Followig the same approach, Singh et al. [29] proposed the use of a sentiment classifier induced from movie reviews as a second filter after collaborative filtering.

2.3.2. Integrating sentiment analysis into recommender systems

A recommender system intends to provide personalized recommendations about products or services to support decision-making in the continuous increase of online information. The most common methods used for recommender systems may be grouped into three categories: content-based; collaborative filtering; and hybrid recommender systems [22].

Sparsity and gray-sheep problems are two of the main reasons why collaborative filtering methods do not provide the reliability required in some recommender systems [26]. In particular, when only sparse ratings data are available, sentiment analysis can play a key role in improving recommendation quality. This is because recommendation algorithms mostly rely on users' ratings to select the items to recommend. Such ratings are usually insufficient and very limited.

Recommender systems can be improved in a variety of ways. In [25], social tag embedding is used in a collaborative filtering approach in which user similarities based on both tag embedding and ratings are combined to generate the recommendations. Recommender systems have also benefited from sentiment analysis. An example of this can be found in the work of Preethi et al. [23], where recursive neural networks were applied to analyze sentiments in reviews. The output was used to improve and validate restaurant and movie recommendations of a cloud-based recommender system. Along with behavioral analysis, sentiment analysis is also an efficient tool

for commodity markets [24]. Wang et al. [66] combined a hybrid recommender system and sentiment analysis to optimize the preliminary list and obtain the final recommendation list. Kumar et al. [67] proposed a hybrid recommender system by combining collaborative filtering and content-based filtering with the use of sentiment analysis of movie tweets to boost up the recommender system.

Rao et al. [68] designed a recommender system that contains the user list and item list with user reviews. Using the sentiment dictionaries, the researchers divided the items into three categories: brand, quality, and price. They leveraged sentiment dictionaries to calculate sentiment of a particular user on item/product. Gurini et al. [69] adopted a different approach to describe a user recommender system for Twitter. Their work emphasized the use of implicit sentiment analysis in order to improve the performance of the recommendation process. They defined a novel weighting function that considers sentiment, volume, and objectivity related to the users' interests.

In yet another approach, Osman et al. [70] presented an electronic product recommender system based on contextual information from sentiment analysis. Because ratings are usually insufficient and very limited, they constructed a contextual information sentiment model for a recommender system by making use of user comments and preferences. In a similar way, Contratre et al. [71] proposed a recommender process that includes sentiment analysis of textual data extracted from Facebook and Twitter in order to increase conversion by matching product offers and consumer preferences. We can find similar combinations in other studies [72-74].

In addition, Rosa et al. [75] used a sentiment intensity metric to build a music recommender system. Users' sentiments are extracted from sentences posted on social networks and the recommendations are made using a framework of low complexity that suggests songs based on the current user's sentiment intensity. The research by Osman, Nurul Aida, and Shahrul [76] addressed the data-sparsity problem of recommender systems by integrating a sentiment-based analysis. Their work was applied to the IMDB and Movie Lens datasets, but improvements in sentiment analysis have been made since the paper was published. Rayan et al. [77] also tried to improve recommendations by addressing the data-sparsity problem. They proposed a smart recommender system based on methods of hybrid learning that integrates the most effective and efficient learning algorithms. These methods switch among content-based and collaborative filtering, identify the user context with the integration of dynamic filtering, and finally learn the profiles.

Several research teams [66, 67, 73, 78-80] introduced the techniques for applying sentiment analysis in recommender systems. The techniques that are applicable for performing the analysis of sentiments include SVM, CNN, RNN, and DNN.

2.4. Proposals

All the proposals included in the thesis are aimed at improving the sentiment analysis on social network data and find methods to increase the performanceof recommender systems.

First, we review the latest studies that have employed deep learning to solve sentiment analysis problems, such as sentiment polarity. We also perform a comparative study and discuss the

experimental results obtained from different models and input features. Then, we propose four hybrid models and study their behavior with different types of datasets from several domains. Next, we evaluate and validate the combination of three models - CNN, LSTM, and SVM - considering the relationship between models and their advanced capacities to extract characteristics, store past information and nodes, and classify text. Finally, we integrate sentiment analysis into recommender systems. This application involves an adaptive recommender system architecture, some techniques for feature extraction, and deep learning models based on sentiment analysis.

The following is a summary of the contributions of this thesis work organized according to the specific topic addressed by each one.

2.4.1. Sentiment Analysis Based on Deep Learning: A Comparative Study

This contribution presents a comprehensive comparison of sentiment analysis methods in the literature and an experimental study to evaluate the performance of deep learning models and related techniques on datasets about different topics. Our research question aims to determine whether it is possible to present outperforming methods for multiple types and sizes of datasets.

This comparative study aims to give an objective overview of different techniques that can guide researchers towards the achievement of better results. This work looks at the latest studies that have used deep learning models to solve various problems related to sentiment analysis. We applied deep learning models with TF-IDF and word embedding to social network datasets and implemented the state-of-the-art sentiment analysis approaches based on deep learning.

We used eight datasets in our experiments on sentiment polarity analysis. Three of them contain tweets; the largest has 1.6 million tweets, with each one labeled as either positive or negative sentiment, while the other two datasets contain 14,640 and 17,750 tweets, respectively, labeled as positive, negative, or neutral. The remaining five datasets include a total of 125,000 comments from user reviews of movies, books, and music labeled as either positive or negative sentiments.

Two approaches for preparing inputs to the classification algorithms are compared in our experiments: word embedding and TF-IDF. For word embedding, we applied Word2vec, which contains models such as skip-gram and continuous bag-of-words (CBOW). Skip-gram makes it possible to start with a known word and predict the words that are likely to surround it. Continuous bag-of-words reverses that and enables the prediction of a word that is likely to occur in the context of known words. For TF-IDF, we used the vectorizer class in the scikit-learn library.

After reviewing the proposed sentiment analysis methods, we identified three popular approaches that have been used frequently in recent studies, namely DNN, CNN, and RNN. These models have been employed in a majority of the 32 reviewed papers, have been widely tested, and have provided highly accurate results when working with different types of datasets. However, as far as we know, no comparative study involving those algorithms has been reported in the literature.

The focus of this research was to evaluate deep learning approaches for sentiment analysis; therefore, we performed a comparative study of the performance of the three most popular deep learning models (DNN, CNN, and RNN) on eight datasets. Moreover, two text processing

techniques (word embedding and TF-IDF) were employed in data preprocessing. The objective of the experiments was to compare the performance of these techniques, contributing in this way to the state-of-the-art literature on sentiment analysis. These algorithms were applied to predict the sentiment polarity of the text and classify it according to that polarity.

2.4.2. Hybrid Deep Learning Models for Sentiment Analysis

Once the comparative study of the individual models had been performed, our next step was to test the reliability of several hybrid techniques on various datasets from different domains. This work is the subject of the second contribution where the research questions are aimed at determining whether it is possible to produce hybrid models that outperform single models with different types of datasets from different domains. Hybrid deep sentiment-analysis learning models that combine LSTM networks, CNN, and SVM are built and tested on eight textual tweets and reviews datasets of different domains. The combination of CNN, LSTM, and SVM aims to take advantage of the two deep network architecture models and SVM algorithms when performing sentiment analysis on different domains and types of datasets.

In this contribution, we proposed four hybrid deep learning models, which are on variations in the use of CNN and LSTM in deep learning layers and variations of CNN and SVM in the classifier layers. We start by using Word2vec or a pre-trained BERT model to create the feature vector. We then vary the order of the CNN and LSTM models used in the next stages: Word2vec/BERT -> CNN -> LSTM or Word2vec/BERT -> LSTM -> CNN. We also vary the final stage of the model, using a Relu function or using an SVM.

Two approaches were used in our experiments to create feature vectors. The first approach was Word2vec initialized with random weights to learn the embedding for all words in our training datasets. Because Word2vec does not include contextual analysis to handle complex semantical or polymorphic cases in natural languages, our second approach was BERT. A pre-trained BERT model was used in this study. After adjusting the parameters, the BERT model was used as a feature extractor to generate input data for the proposal of hybrid models. The tweets and reviews data were fed into the BERT model to generate the feature vectors, which are the input to the hybrid models that perform the classification. The next step combines CNN and LSTM deep learning models, which are used because of their good performance on sentiment analysis, as well as to take advantage of the two network architectures when performing sentiment analysis on data in different domains. The final stage is classification. We use the activate function of Relu instead of Sigmoid because of the high convergence. In addition, SVM was chosen for classification because of its efficiency in word processing, especially in high dimensional contexts such as natural language processing. We have applied linear SVMs for classification with the proposed hybrid deep learning models. We extract feature vectors from the top hidden layer and feed it to SVM that will classify for prediction (“positive”, “negative”). The architecture of these hybrid models are discussed below.

The first hybrid model combines CNN and LSTM models. The embedding function is the embedding layer that is initialized with random weights and which will learn the embedding for all words in the training datasets. The first layer of the hybrid model is the CNN, which receives the vector produced by word embedding. It has three convolution layers consisting of 512, 256,

and 128 filters respectively, with a kernel size = 3, which receives and processes data before feeding it into next deep learning layer. The second layer of the hybrid model is the LSTM, which produces a 1x500 matrix that is fed into the classifier. Next, the hybrid model's classifier is composed of two continuous, fully-connected layers with 128 nodes and, finally, the output layer with a Relu activation function.

The second hybrid model combines LSTM and CNN models. The input data is pre-processed to reshape data for the embedding matrix. The first layer of the hybrid model is the LSTM layer. That output has a matrix 13x500 and is fed into the second model of the hybrid deep-learning model. The next layer of the hybrid model is the CNN. It has three convolution layers consisting of 512, 256, and 128 filters respectively, with a kernel size = 3, which are in charge of receiving and processing data before feeding it into the next layer. The CNN output is flattened and transferred to a fully connected layer. And finally, the hybrid model's classifier is a CNN composed of two continuous fully connected layers with 128 nodes and the Relu activation function as the output layer.

Our final hybrid model is based on the hybrid models from scenarios 1 and 2. We use the deep learning stages from those models (CNN-LSTM and LSTM-CNN) but replace the classifier. While there are multiple alternatives to the CNN-based Relu function used, we have chosen to use SVM for the replacement classifier. Scenario 3 is based on CNN-LSTM, and Scenario 4 is based on LSTM-CNN.

2.4.3. An Approach to Integrating Sentiment Analysis into Recommender Systems

In the case of social data, sentiment analysis can help gain better understanding of a user's attitudes, opinions and emotions, which is beneficial to integrate in recommender systems for achieving higher recommendation reliability. On the one hand, this information can be used to complement explicit ratings given to products by users. On the other hand, sentiment analysis of items that can be derived from online news services, blogs, social media, or even from the recommender systems themselves is seen as capable of providing better recommendations to users. Since the ultimate goal of our research is to apply previous findings regarding sentiment analysis in improving recommender systems, in this contribution, we propose a recommendation method that combines sentiment analysis and collaborative filtering. The method is implemented in an adaptive recommender system architecture in which techniques for feature extraction and deep learning-based sentiment analysis is included. The aim is to improve reliability of the recommendations to the user by combining sentiment analysis of reviews or comments of users with traditional recommendation methods.

The architecture makes it easy to configure the modules and their interactions, allowing the application to be composed by choosing from supported techniques and methods. The architecture has two separate parts, one part in charge of generating the sentiment models and the other part to provide recommendations to a given user making use of the models previously generated. The reviews' data were preprocessed and used to conduct and train a sentiment-based hybrid deep-learning model. Then, a user-based (user-user) collaborative filtering method is combined with sentiment-based models for rating prediction.

We used the combination of several successful approaches [81] proposed in the previous contribution. We start by using a pre-trained BERT model to create the feature vectors. We then vary the order of the CNN and LSTM models used in the next stages: BERT → CNN → LSTM or BERT → LSTM → CNN. The final stage of the model uses a ReLu activation function. We labeled the reviews with one value of an ordinal scale of five classes (very negative; negative; neutral; positive; and very positive), analogous to the explicit ratings, to train and validate the results of sentiment analysis.

The reviews data were fed into the BERT model to generate the feature vectors, which were input to the proposal of hybrid models that performed the classification. The combining CNN and LSTM deep learning models as well as take advantage of the two network architectures when performing sentiment analysis on data in different domains. The final stage was classification. We used the activate function of Relu instead of Sigmoid because of the high convergence.

The proposed recommendation method is a user-based collaborative filtering approach that considers explicit ratings and sentiment analysis extracted from users' reviews. We tested Singular Value Decomposition (SVD), Non-Negative Matrix Factorization (NMF), and SVD++ (a derivative of SVD) as collaborative filtering methods. The objective was to achieve better predictive accuracy because of the addition of implicit feedback information provided by the sentiment. Results from the collaborative filtering recommendation method and sentiment analysis were combined to generate a rating and used to create a list of recommendations.

2.4.4. Framework for retrieving relevant contents related to fashion from online social network data

This work was a preliminary study conducted before the contributions described above. We souht to understand the data about social networks, techniques, libraries and metrics in order to evaluate the models. We also wanted implement crawling, data preprocessing, and building the analysis model. This work led to the subsequent research which in turn has given rise to the rest of the contributions. We proposed a comprehensive framework for retrieving relevant contents from online social network data. Our proposed approach was based on Vector Space Model and Support Vector Machine to process and classify raw text data. Our experiments demonstrated the utility and accuracy of the framework in retrieving fashion related contents from Twitter and Facebook.

Our framework is as follows: we use a measure to determine the importance of a word in the text, called TF-IDF: we transform the document text of information into Vector Space Model (VSM). This model allows the text to be represented by the vector in n-dimensional space, each dimension corresponding to the index. In this space, each component of the text vector represents the weighted measure of the index corresponding to that text. First, based on the vectors representation in the document, a text classification model is built from a training set by means of a SVM algorithm. To build the training sample, we use accounts that provide updates about fashion (Ex accounts of www.Fashionista.com andwww.elle.com in Twitter and Facebook). Experiments on two popular social networks Facebook and Twitter, have been chosen to see the efficiency and accuracy of the model in the extracted information about fashion.

Our proposed framework automatically trains a model for retrieving fashion data from Facebook and Twitter. Basically, there are two stages for retrieving information in this framework (1) model training; and (2) retrieving fashion contents using the induced classification model.

In the stage of model training, social network data (such as comments and posts) are collected in real time by using REST and Graph protocol (as REST APIs in Twitter and Graph API in Facebook). Some popular accounts or fan pages providing status of fashion are used to train a SVM model. Text data are transformed into a Space Vector Model by using IF-IDF measure. Using the model trained from training data, new information from social network is classified and related fashion information is retrieved and saved. In the next stage, we also collect data from accounts in Facebook and Twitter that are not related to fashion such as: economics; weather; traffic; and technology. These data are merged with related fashion data to test the accuracy of the SVM model.

2.5. Validation

The validation of the proposals has been conducted with different public datasets that are widely accepted by the research community. The datasets were obtained from various different sources and they cover different topics in order to perform a wide range of experiments.

In all cases, we applied k-fold cross-validation to the data in order to evaluate the models. The common values for K-Fold validation method are k=3, k=5, and k=10, and by far the most popular value used in applied machine learning to evaluate models is k=10.

Regarding sentiment analysis, Accuracy, AUC, and F-score were the metrics used to evaluate the performance of the deep-learning models for sentiment analysis through all experiments. Since F-score is derived from recall and precision, we also show these two measures for reference purposes. Because time is one of the most valuable resources and the one most often considered when evaluating the performance of algorithms, we include the analysis of the computational time of the models involved in the comparative study, as this is a reflection of the time complexity. It includes the time for data division and creating the classification model but excludes the time to display the classification results.

To validate our recommendation approaches, we compared the performance of widely used collaborative filtering recommendation methods in their traditional form as the baseline and the same methods improved with our proposal involving the use of sentiment analysis of reviews. The comparative study was conducted for both rating predictions and item recommendations (recommendation of top-N lists).

The metrics used to evaluate the reliability of rating predictions were Root-Mean-Square Error (RMSE), Mean Absolute Error (MAE) and Normal MAE (NMAE). In addition, Mean Reciprocal Rank (MRR), Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG) were used for evaluating top-N recommendations.

2.6. Conclusions and future work

This thesis described the core of deep learning models and related techniques applied for sentiment analysis on social network data. The architectures of DNN, CNN, and RNN were analyzed and combined with word embedding and TF-IDF to perform sentiment analysis. Then, we proposed hybrid deep learning models for sentiment analysis from social network data. In addition, we have offered an application of sentiment analysis in recommender systems based on hybrid deep-learning models and collaborative filtering. The system architecture presented in this work can integrate a variety of techniques that have been proposed to perform recommendations, including the preprocessing strategy, hybrid deep-learning models for sentiment analysis, and methods for recommender systems. The architecture can be used to develop a recommender system in social networks that take advantage of sentiment analysis performed on user opinions and reviews in the network.

We studied the impacts on different types of datasets, feature extraction techniques, and deep learning models, focusing on the problem of sentiment polarity analysis and its integration into recommendation systems. The results show that combining deep learning techniques with word embedding is better than with TF-IDF when performing sentiment analysis. The experiments of hybrid models outperformed all the models tested for sentiment polarity analysis. Combining deep learning models with the SVM technique yields better results for performing sentiment analysis than using an individual model. In most of the datasets that we tested, the reliability of hybrid models using SVM was higher than those models not using it; however, the computational time is much longer for the former. The experiments also revealed that CNN outperforms other models, providing a good balance between accuracy and CPU time. RNN reliability is slightly higher than CNN reliability with most datasets, but its computational time is longer. To integrate sentiment into the recommender system, we conducted experiments with food, and movies review datasets. Based on such experiments, we demonstrated the utility and applicability of our approaches in producing personalized recommendations on online social networks. The results showed that the everyday use of deep learning-based sentiment analysis and collaborative filtering methods improved the results. This was achieved through the exploitation of additional information from user data. Its integration into the traditional recommendation methods makes the recommender system more reliable and capable of providing better recommendations to users.

We have commented on and analyzed the results obtained in this thesis. However, our ideas offer many directions for further development that would broaden the application of the research, increase user convenience, and suggest paths for subsequent researchers. We intend to address aspect sentiment analysis to gain deeper insight into user sentiments by associating them with specific features or topics in future work. This work has great relevance for many companies because it allows them to obtain detailed feedback from users and thus know which aspects of their products or services should be improved. Future work will also consider new sentiment analysis techniques, such as graph convolutional networks, to improve this aspect. Finally, we will aim at a comprehensive framework that monitors multiple data sources generated by online social network users to extract user opinions for recommendations and collaborative filtering on online social networks.

2.7. References

1. Bhavitha, B.; Rodrigues, A. P.; Chiplunkar, N. N. In *Comparative study of machine learning techniques in sentimental analysis*, 2017 International conference on inventive communication and computational technologies (ICICCT), 2017; IEEE: 2017; pp 216-221.
2. Dang, N. C.; Moreno-García, M. N.; De la Prieta, F., Sentiment analysis based on deep learning: A comparative study. *Electronics* **2020,** 9, (3), 483.
3. Mhaskar, H.; Liao, Q.; Poggio, T. In *When and why are deep networks better than shallow ones?*, Proceedings of the AAAI Conference on Artificial Intelligence, 2017; 2017.
4. Schindler, A.; Lidy, T.; Rauber, A. In *Comparing Shallow versus Deep Neural Network Architectures for Automatic Music Genre Classification*, FMT, 2016; 2016; pp 17-21.
5. Hochreiter, S.; Schmidhuber, J. In *LSTM can solve hard long time lag problems*, Advances in neural information processing systems, 1997; 1997; pp 473-479.
6. Zhang, L.; Wang, S.; Liu, B., Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining Knowledge Discovery* **2018**, e1253.
7. Alfrjani, R.; Osman, T.; Cosma, G., A Hybrid Semantic Knowledgebase-Machine Learning Approach for Opinion Mining. *Data & Knowledge Engineering* **2019,** 121, 88-108.
8. Gupta, I.; Joshi, N., Enhanced Twitter Sentiment Analysis Using Hybrid Approach and by Accounting Local Contextual Semantic. *Journal of Intelligent Systems* **2019,** 29, (1), 1611-1625.
9. Sánchez-Rada, J. F.; Iglesias, C. A., CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection. *Applied Sciences* **2020,** 10, (5), 1662.
10. Elleuch, M.; Maalej, R.; Kherallah, M., A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition. *Procedia Computer Science* **2016,** 80, 1712-1723.
11. Tang, Y., Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239* **2013**.
12. Xue, D.-X.; Zhang, R.; Feng, H.; Wang, Y.-L., CNN-SVM for microvascular morphological type recognition with data augmentation. *Journal of medical and biological engineering* **2016,** 36, (6), 755-764.
13. Chen, T.; Xu, R.; He, Y.; Wang, X., Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications* **2017,** 72, 221-230.
14. Rehman, A. U.; Malik, A. K.; Raza, B.; Ali, W., A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis. *Multimedia Tools and Applications* **2019,** 78, (18), 26597-26613.
15. Vo, Q.-H.; Nguyen, H.-T.; Le, B.; Nguyen, M.-L. In *Multi-channel LSTM-CNN model for Vietnamese sentiment analysis*, 2017 9th international conference on knowledge and systems engineering (KSE), 2017; IEEE: 2017; pp 24-29.
16. Martín, C. A.; Torres, J. M.; Aguilar, R. M.; Diaz, S., Using deep learning to predict sentiments: case study in tourism. *Complexity* **2018,** 2018.
17. Elshakankery, K.; Ahmed, M. F., HILATSA: A hybrid Incremental learning approach for Arabic tweets sentiment analysis. *Egyptian Informatics Journal* **2019,** 20, (3), 163-171.
18. Putra, S. J.; Khalil, I.; Gunawan, M. N.; Amin, R. I.; Sutabri, T. In *A Hybrid Model for Social Media Sentiment Analysis for Indonesian Text*, Proceedings of the 20th International Conference on Information Integration and Web-based Applications & Services, 2018; 2018; pp 297-301.
19. Astya, P. In *Sentiment analysis: approaches and open issues*, 2017 International Conference on Computing, Communication and Automation (ICCCA), 2017; IEEE: 2017; pp 154-158.

20. Lu, J.; Wu, D.; Mao, M.; Wang, W.; Zhang, G., Recommender system application developments: a survey. *Decision Support Systems* **2015,** 74, 12-32.
21. Betru, B. T.; Onana, C. A.; Batchakui, B., A Survey of State-of-the-art: Deep Learning Methods on Recommender System. *International Journal of Computer Applications* **2017,** 162, (10).
22. Kardan, A. A.; Ebrahimi, M., A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. *Information Sciences* **2013,** 219, 93-110.
23. Preethi, G.; Krishna, P. V.; Obaidat, M. S.; Saritha, V.; Yenduri, S. In *Application of Deep Learning to Sentiment Analysis for recommender system on cloud*, 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), 2017; IEEE: 2017; pp 93-97.
24. Keenan, M. J. S., *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets: Bridging Fundamental and Technical Analysis*. 2nd ed.; 2018.
25. Sánchez-Moreno, D.; Moreno-García, M. N.; Mobasher, B.; Sonboli, N.; Burke, R., Using Social Tag Embedding in a Collaborative Filtering Approach for Recommender Systems. In *The 2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*.
26. Sánchez-Moreno, D.; López Batista, V. F.; Muñoz Vicente, M. D.; Gil González, A. B.; Moreno-García, M. N., A session-based song recommendation approach involving user characterization along the play power-law distribution. *Complexity* **2020**.
27. Salas-Zárate, M. d. P.; Medina-Moreira, J.; Lagos-Ortiz, K.; Luna-Aveiga, H.; Rodriguez-Garcia, M. A.; Valencia-Garcia, R., Sentiment analysis on tweets about diabetes: an aspect-level approach. *Computational mathematical methods in medicine* **2017,** 2017.
28. Zhang, X.; Zheng, X. In *Comparison of Text Sentiment Analysis Based on Machine Learning*, 15th International Symposium on Parallel and Distributed Computing (ISPDC), 2016; IEEE: 2016; pp 230-233.
29. Pandey, A. C.; Rajpoot, D. S.; Saraswat, M., Twitter sentiment analysis using hybrid cuckoo search method. *Information Processing & Management* **2017,** 53, (4), 764-779.
30. Tang, D.; Qin, B.; Liu, T., Deep learning for sentiment analysis: successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **2015,** 5, (6), 292-303.
31. Sharef, N. M.; Zin, H. M.; Nadali, S., Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data. *JCS* **2016,** 12, (3), 153-168.
32. Ain, Q. T.; Ali, M.; Riaz, A.; Noureen, A.; Kamran, M.; Hayat, B.; Rehman, A. J. I. J. A. C. S. A., Sentiment analysis using deep learning techniques: a review. **2017,** 8, (6), 424.
33. Singhal, P.; Bhattacharyya, P., Sentiment analysis and deep learning: a survey. In 2016.
34. Rojas-Barahona, L. M., Deep learning for sentiment analysis. *Language and Linguistics Compass* **2016,** 10, (12), 701-719.
35. Sohangir, S.; Wang, D.; Pomeranets, A.; Khoshgoftaar, T. M., Big Data: Deep Learning for financial sentiment analysis. *Journal of Big Data* **2018,** 5, (1), 3.
36. Jangid, H.; Singhal, S.; Shah, R. R.; Zimmermann, R. In *Aspect-Based Financial Sentiment Analysis using Deep Learning*, Companion of the The Web Conference 2018 on The Web Conference 2018, 2018; International World Wide Web Conferences Steering Committee: 2018; pp 1961-1966.
37. Qian, J.; Niu, Z.; Shi, C. In *Sentiment Analysis Model on Weather Related Tweets with Deep Neural Network*, Proceedings of the 2018 10th International Conference on Machine Learning and Computing, 2018; ACM: 2018; pp 31-35.
38. Pham, D.-H.; Le, A.-C., Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data & Knowledge Engineering* **2018,** 114, 26-39.

39. Gao, Y.; Rong, W.; Shen, Y.; Xiong, Z. In *Convolutional neural network based sentiment analysis using Adaboost combination*, 2016 International Joint Conference on Neural Networks (IJCNN), 2016; IEEE: 2016; pp 1333-1338.
40. Hassan, A.; Mahmood, A. In *Deep learning approach for sentiment analysis of short texts*, Third International Conference on Control, Automation and Robotics (ICCAR), 2017; IEEE: 2017; pp 705-710.
41. Kraus, M.; Feuerriegel, S., Sentiment analysis based on rhetorical structure theory: Learning deep neural networks from discourse trees. *Expert Systems with Applications* **2019,** 118, 65-79.
42. Li, L.; Goh, T.-T.; Jin, D., How textual quality of online reviews affect classification performance: a case of deep learning sentiment analysis. *Neural Computing and Applications* **2018**, 1-29.
43. Jeong, B.; Yoon, J.; Lee, J.-m., Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis. *International Journal of Information Management* **2017**.
44. Gupta, U.; Chatterjee, A.; Srikanth, R.; Agrawal, P., A sentiment-and-semantics-based approach for emotion detection in textual conversations. *arXiv preprint arXiv:.06996* **2017**.
45. Alharbi, A. S. M.; de Doncker, E., Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. *Cognitive Systems Research* **2019,** 54, 50-61.
46. Abid, F.; Alam, M.; Yasir, M.; Li, C., Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter. *Future Generation Computer Systems* **2019,** 95, 292-308.
47. Vateekul, P.; Koomsubha, T. In *A study of sentiment analysis using deep learning techniques on Thai Twitter data*, 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2016; IEEE: 2016; pp 1-6.
48. Malik, V.; Kumar, A., Sentiment Analysis of Twitter Data Using Naive Bayes Algorithm. *International Journal on Recent and Innovation Trends in Computing and Communication* **2018,** 6, (4), 120-125.
49. Ramadhani, A. M.; Goo, H. S. In *Twitter sentiment analysis using deep learning methods*, 2017 7th International Annual Engineering Seminar (InAES), 2017; IEEE: 2017; pp 1-4.
50. Paredes-Valverde, M. A.; Colomo-Palacios, R.; Salas-Zárate, M. d. P.; Valencia-García, R. J. S. P., Sentiment analysis in Spanish for improvement of products and services: A deep learning approach. **2017,** 2017.
51. Roshanfekr, B.; Khadivi, S.; Rahmati, M. In *Sentiment analysis using deep learning on Persian texts*, Electrical Engineering (ICEE), 2017 Iranian Conference on, 2017; IEEE: 2017; pp 1503-1508.
52. Yang, C.; Zhang, H.; Jiang, B.; Li, K., Aspect-based sentiment analysis with alternating coattention networks. *Information Processing & Management* **2019,** 56, (3), 463-478.
53. Do, H. H.; Prasad, P.; Maag, A.; Alsadoon, A., Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review. *Expert Systems with Applications* **2019,** 118, 272-299.
54. Schmitt, M.; Steinheber, S.; Schreiber, K.; Roth, B., Joint Aspect and Polarity Classification for Aspect-based Sentiment Analysis with End-to-End Neural Networks. *arXiv preprint arXiv:1808.09238* **2018**.
55. Srinidhi, H.; Siddesh, G.; Srinivasa, K., A hybrid model using MaLSTM based on recurrent neural networks with support vector machines for sentiment analysis. *Engineering and Applied Science Research* **2020,** 47, (3), 232-240.
56. Akhtar, M. S.; Kumar, A.; Ekbal, A.; Bhattacharyya, P. In *A hybrid deep learning architecture for sentiment analysis*, Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, 2016; 2016; pp 482-493.

57. Al-Azani, S.; El-Alfy, E.-S. M. In *Hybrid deep learning for sentiment polarity determination of arabic microblogs*, International Conference on Neural Information Processing, 2017; Springer: 2017; pp 491-500.
58. Liu, G.; Xu, X.; Deng, B.; Chen, S.; Li, L. In *A hybrid method for bilingual text sentiment classification based on deep learning*, 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2016; IEEE: 2016; pp 93-98.
59. Zhang, Q.; Zhang, Z.; Yang, M.; Zhu, L., Exploring Coevolution of Emotional Contagion and Behavior for Microblog Sentiment Analysis: A Deep Learning Architecture. *Complexity* **2021,** 2021.
60. Kaur, H.; Ahsaan, S. U.; Alankar, B.; Chang, V., A Proposed Sentiment Analysis Deep Learning Algorithm for Analyzing COVID-19 Tweets. *Information Systems Frontiers* **2021**, 1-13.
61. Kastrati, Z.; Ahmedi, L.; Kurti, A.; Kadriu, F.; Murtezaj, D.; Gashi, F., A Deep Learning Sentiment Analyser for Social Media Comments in Low-Resource Languages. *Electronics* **2021,** 10, (10), 1133.
62. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K., Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:.04805* **2018**.
63. Jnoub, N.; Al Machot, F.; Klas, W., A Domain-Independent Classification Model for Sentiment Analysis Using Neural Models. *Applied Sciences* **2020,** 10, (18), 6221.
64. Ombabi, A. H.; Ouarda, W.; Alimi, A. M.; Mining, Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks. *Social Network Analysis and Mining* **2020,** 10, (1), 1-13.
65. Yoo, G.; Nam, J. In *A Hybrid Approach to Sentiment Analysis Enhanced by Sentiment Lexicons and Polarity Shifting Devices*, The 13th Workshop on Asian Language Resources, 2018; 2018.
66. Wang, Y.; Wang, M.; Xu, W., A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework. *Wireless Communications and Mobile Computing* **2018**.
67. Kumar, S.; De, K.; Roy, P. P., Movie recommendation system using sentiment analysis from microblogging data. *IEEE Transactions on Computational Social Systems* **2020,** 7, (4), 915-923.
68. Rao, K. Y.; Murthy, G.; Adinarayana, S., Product recommendation system from users reviews using sentiment analysis. *International Journal of Computer Applications* **2017,** 975, 8887.
69. Gurini, D. F.; Gasparetti, F.; Micarelli, A.; Sansonetti, G., A Sentiment-Based Approach to Twitter User Recommendation. *RSWeb@ RecSys* **2013,** 1066.
70. Osman, N.; Noah, S.; Darwich, M., Contextual sentiment based recommender system to provide recommendation in the electronic products domain. *International Journal of Machine Learning and Computing* **2019,** 9, (4), 425-431.
71. Contratres, F. G.; Alves-Souza, S. N.; Filgueiras, L. V. L.; DeSouza, L. S. In *Sentiment analysis of social network data for cold-start relief in recommender systems*, World Conference on Information Systems and Technologies, 2018; Springer: 2018; pp 122-132.
72. Nabil, S.; Elbouhdidi, J.; Yassin, M. In *Recommendation system based on data analysis-application on tweets sentiment analysis*, 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), 2018; IEEE: 2018; pp 155-160.
73. Ziani, A.; Azizi, N.; Schwab, D.; Aldwairi, M.; Chekkai, N.; Zenakhra, D.; Cheriguene, S. In *Recommender system through sentiment analysis*, 2nd International Conference on Automatic Control, Telecommunications and Signals, 2017; 2017.
74. Abbasi, F.; Khadivar, A.; Yazdinejad, M., A Grouping Hotel Recommender System Based on Deep Learning and Sentiment Analysis. *Journal of Information Technology Management* **2019,** 11, (2), 59-78.

75. Rosa, R. L.; Rodriguez, D. Z.; Bressan, G., Music recommendation system based on user's sentiments extracted from social networks. *IEEE Transactions on Consumer Electronics* **2015,** 61, (3), 359-367.
76. Osman, N. A.; Noah, S. A. M. In *Sentiment-based model for recommender systems*, 2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP), 2018; IEEE: 2018; pp 1-6.
77. Nouh, R. M.; Lee, H.-H.; Lee, W.-J.; Lee, J.-D., A smart recommender based on hybrid learning methods for personal well-being services. *Sensors* **2019,** 19, (2), 431.
78. Devipriya, K.; Prabha, D.; Pirya, V.; Sudhakar, S., Deep learning sentiment analysis for recommendations in social applications. *Int J Sci Technol Res* **2020,** 9, (1), 3812-3815.
79. Singh, V. K.; Mukherjee, M.; Mehta, G. K. In *Combining collaborative filtering and sentiment classification for improved movie recommendations*, International Workshop on Multi-disciplinary Trends in Artificial Intelligence, 2011; Springer: 2011; pp 38-50.
80. Nimirthi, P.; Krishna, P. V.; Obaidat, M. S.; Saritha, V., A framework for sentiment analysis based recommender system for agriculture using deep learning approach. In *Social Network Forensics, Cyber Security, and Machine Learning*, Springer: 2019; pp 59-66.
81. Dang, C. N.; Moreno-García, M. N.; De la Prieta, F., Hybrid Deep Learning Models for Sentiment Analysis. *Complexity* **2021,** 2021.

3. Resumen en español de las contribuciones

3.1. Contribución 1

3.1.1. Referencia

Dang, Nhan Cach, María N. Moreno-García, and Fernando De la Prieta. "Sentiment analysis based on deep learning: A comparative study", *Electronics* 9, no. 3 (2020): 483.

3.1.2. Objetivos

Nuestro objetivo es determinar si es posible aplicar métodos de análisis de sentimiento de alto rendimiento y fiabilidadpara múltiples tipos y tamaños de conjuntos de datos. Varios estudios han propuesto análisis de sentimientos basados en deep learning, que tienen características y niveles de rendimiento diferentes. Este trabajo analiza los últimos estudios que han utilizado modelos de aprendizaje profundo, como DNN, RNN y CNN, para resolver diferentes problemas relacionados con el análisis de sentimientos. Aplicamos modelos de aprendizaje profundo con frecuencia de documentos de frecuencia inversa de términos (TF-IDF) e word embedding en datos de redes sociales e implementamos el estado del arte de enfoques de análisis de sentimientos basados en el aprendizaje profundo.

3.1.3. Enfoque metodológico

En este trabajo se identifican tres enfoques populares que se han utilizado con frecuencia en estudios recientes, DNN, CNN y RNN, y se realiza un estudio comparativo del rendimiento de los tres modelos de *deep learning* más populares en ocho conjuntos de datos. Además, se utilizan dos técnicas de procesamiento de texto (word embedding y TF-IDF) en el preprocesamiento de datos. El objetivo de los experimentos es comparar el desempeño de estas técnicas, contribuyendo de esta manera en la literatura relativa al análisis de sentimientos. Los algoritmos se aplicaron para predecir la polaridad de sentimiento del texto y clasificarlo de acuerdo con esa polaridad. El desempeño de esos métodos se evaluó mediante las métricas más adecuadas utilizadas para los problemas de clasificación: exactitud, *recall*, medida F y AUC. Usamos la validación cruzada (*k-fold cross validation*) con k = 10 en la aplicación de las métricas.

La limpieza de datos y la extracción de características se realizaron en las etapas de preprocesamiento. La limpieza de texto es un paso de procesamiento previo que elimina palabras u otros componentes que no contienen información relevante y, por lo tanto, puede reducir la efectividad del análisis de opiniones. Los datos de texto incluyen espacios en blanco, puntuación y palabras vacías que tienen que eliminarse.

Una vez que se limpiaron los conjuntos de datos, las oraciones se dividieron en palabras individuales, que se devolvieron a su forma base mediante lematización. En este punto, las oraciones se convirtieron en vectores de números reales continuos utilizando dos métodos: *word embedding* y TF-IDF. Ambos tipos de vectores de características fueron las entradas a los

algoritmos de *deep learning* evaluados en el estudio. Esos algoritmos fueron CNN, DNN y RNN. Así, se indujeron dos modelos por algoritmo, uno para cada tipo de vector.

La mayoría de los modelos tradicionales utilizan funciones conocidas, como bolsa de palabras, n-gramas y TF-IDF. Tales características no consideran la similitud semántica entre palabras. Actualmente, muchos modelos de aprendizaje profundo en Procesamiento del Lenguaje Natural (PLN) requieren resultados de *word embedding* como características de entrada. La función embedding es la capa de incrustación que se inicializa con pesos aleatorios y que aprenderá la incrustación de todas las palabras en los conjuntos de datos de entrenamiento. En nuestro caso, el tamaño del vocabulario es 15.000, el dim de salida es 300 y la longitud máxima es 40. Los resultados están en una matriz de 40 × 300.

La primera capa 1D CNN define un filtro de tamaño de kernel 3. Para ello, definiremos 64 filtros. Esto nos permite entrenar 64 características diferentes en la primera capa de la red. Por tanto, la salida de la primera capa de red neuronal es una matriz de neuronas de 40 × 64, y el resultado de la primera capa de CNN se introducirá en la segunda capa de CNN. Nuevamente definiremos 32 filtros diferentes para entrenar en este nivel. Siguiendo la misma lógica que la primera capa, la matriz de salida medirá 40 × 32.

La capa de *pooling* máxima se utiliza a menudo después de una capa de CNN para reducir la complejidad de la salida y evitar el sobreajuste de los datos. En nuestro caso, elegimos un tamaño de tres. Esto significa que el tamaño de la matriz de salida de esta capa es 13 × 32.

La tercera y cuarta capas de CNN 1D se encargan de aprender las funciones de nivel superior. Las salidas de esas dos capas son una matriz de 13 × 16 y una matriz de 13 × 8.

La capa de *pooling* media es una capa de agrupación que se utiliza para evitar aún más el sobreajuste. Usaremos el valor promedio en lugar del valor máximo porque dará mejores resultados en este caso. La matriz de salida tiene un tamaño de 1 x 8 neuronas.

La capa completamente conectada con activación sigmoidea es la capa final que reducirá el vector de altura de 8 a 1 para la predicción ("positivo", "negativo").

3.1.4. Resultados

Como primer objetivo de la investigación, esta contribución revisa los últimos estudios que han empleado el aprendizaje profundo para resolver problemas de análisis de sentimientos, como la polaridad de sentimientos. Los modelos que utilizan TF-IDF y *word embedding* se han aplicado a una serie de conjuntos de datos. Se ha realizado un estudio comparativo de los resultados experimentales obtenidos para los diferentes modelos y características de entrada. Se realizaron muchos experimentos para evaluar los modelos DNN, CNN y RNN en conjuntos de datos de diferentes temas, incluidos tweets y reseñas. Así mismo, discutimos investigaciones relacionadas en el campo. Además, abordamos las siguientes contribuciones relacionadas:

- Se describieron y discutieron modelos de *deep learning* adecuados para el análisis de sentimientos y técnicas para preparar las entradas a los algoritmos de clasificación de sentimientos.
- Se encontró que el modelo de CNN ofrece el mejor equilibrio entre el tiempo de procesamiento y la precisión de los resultados. Aunque el modelo RNN tuvo la mayor

precisión cuando se usó con *word embedding*, su tiempo de procesamiento fue diez veces mayor que el modelo CNN. El modelo RNN no es efectivo cuando se usa con la técnica TF-IDF, y su tiempo de procesamiento mucho mayor conduce a resultados que no son significativamente mejores. DNN es un modelo de aprendizaje profundo simple que tiene tiempos de procesamiento intermedio y produce resultados medios.

- En cuanto a las técnicas de preprocesamiento, los resultados cuando se usa TF-IDF son peores que cuando se usa *word embedding*. Además, TF-IDF que utiliza el modelo RNN tiene un tiempo de procesamiento más largo y produce resultados menos fiables. Sin embargo, cuando RNN se usa con *word embedding*, los resultados son mucho mejores.
- Los resultados de los conjuntos de datos que contienen tweets y los conjuntos de datos de revisión de películas de IMDB son mejores que los de los otros conjuntos de datos que contienen revisiones. Además, los modelos inducidos a partir del conjunto de datos de Tweets Airline centrados en un tema específico muestran un mejor rendimiento que los construidos a partir de conjuntos de datos sobre cuestiones genéricas.

3.1.5. Conclusiones

En este trabajo se describen los aspectos fundamentales de los modelos de deep learning y técnicas relacionadas que se han aplicado en el análisis de sentimientos para datos de redes sociales. Usamos *word embedding* y TF-IDF para transformar los datos de entrada antes de introducirlos en modelos de deep learning. Las arquitecturas de DNN, CNN y RNN se analizaron y combinaron con incrustaciones de palabras y TF-IDF para realizar análisis de sentimientos. Realizamos algunos experimentos para evaluar los modelos DNN, CNN y RNN en conjuntos de datos de diferentes temas, incluidos tweets y reseñas. También discutimos investigaciones relacionadas en el campo. Esta información, combinada con los resultados de nuestros experimentos, nos brinda una perspectiva amplia sobre la aplicación de modelos de *deep learning* para el análisis de sentimientos, así como sobre la combinación de estos modelos con técnicas de preprocesamiento de texto.

Los enfoques DNN, CNN e híbridos se identificaron como los modelos más utilizados para el análisis de polaridad de sentimiento. Otra conclusión extraída del análisis fue el hecho de que las técnicas comunes, como CNN, RNN y LSTM, se prueban individualmente en estos estudios en diferentes conjuntos de datos, sin embargo, existe una falta de un análisis comparativo para ellas. Además, los resultados presentados en la mayoría de los artículos se dan en términos de confiabilidad, sin considerar el tiempo computacional.

Los experimentos llevados a cabo en este trabajo fueron diseñados para ayudar a llenar los vacíos mencionados anteriormente. Estudiamos los impactos de diferentes tipos de conjuntos de datos, técnicas de extracción de características y modelos de deep learning, con un enfoque especial en el problema del análisis de polaridad de sentimientos. Los resultados mostraron que al realizar un análisis de sentimientos, era mejor combinar técnicas de deep learning con word embeddingque con TF-IDF. Los experimentos también revelaron que CNN supera a otros modelos, presentando un buen equilibrio entre precisión y tiempo de ejecución de la CPU. La

confiabilidad de RNN es ligeramente más alta que la confiabilidad de CNN con la mayoría de los conjuntos de datos, pero su tiempo de cálculo es mucho más largo. Una última conclusión derivada del estudio es la observación de que la efectividad de los algoritmos depende en gran medida de las características de los conjuntos de datos, de ahí la conveniencia de probar métodos de deep learning con más conjuntos de datos para cubrir una mayor diversidad de características.

3.2. Contribución 2

3.2.1. Referencia

Dang, Cach N., María N. Moreno-García, and Fernando De la Prieta. "Hybrid Deep Learning Models for Sentiment Analysis", Complexity 2021 (2021).

3.2.2. Objetivos

El objetivo de esta contribución es probar la fiabilidad de varias técnicas híbridas en varios conjuntos de datos de diferentes dominios. Nuestras preguntas de investigación tienen como objetivo determinar si es posible producir modelos híbridos que superen los modelos individuales con conjuntos de datos de diferentes dominios y tipos. En el trabajo se pretende proponer modelos híbridos que aumenten la precisión del análisis de sentimientos en comparación con modelos individuales en todos los tipos de conjuntos de datos.

3.2.3. Metodología propuesta

Existen numerosos métodos para construir un modelo híbrido para el análisis de sentimientos. En esta contribución, probamos la combinación de varios enfoques exitosos. Comenzamos usando Word2vec o un modelo BERT previamente entrenado para crear el vector de características. Luego variamos el orden de los modelos CNN y LSTM usados en las siguientes etapas: Word2vec / BERT -> CNN -> LSTM o Word2vec / BERT -> LSTM -> CNN. También variamos la etapa final del modelo, usando una función Relu o usando una SVM (Support Vector Machine). La combinación de estos dos tipos de variación produce los cuatro enfoques híbridos que hemos probado:

- Word2vec/BERT -> CNN -> LSTM -> Relu
- Word2vec/BERT -> LSTM -> CNN -> Relu
- Word2vec/BERT -> CNN -> LSTM -> SVM
- Word2vec/BERT -> LSTM -> CNN -> SVM

Se utilizaron dos técnicas distintas en nuestros experimentos para crear vectores de características. La primera fue Word2vec inicializado con pesos aleatorios para aprender la incrustación de todas las palabras (*word embedding*) en nuestros conjuntos de datos de entrenamiento. Debido a que Word2vec no incluye análisis contextual para manejar casos semánticos o polimórficos complejos en lenguajes naturales, nuestro segundo enfoque fue BERT. En este estudio se utilizó un modelo BERT previamente entrenado. Después de ajustar los parámetros, se utilizó el modelo BERT como extractor de características para generar datos de entrada para la propuesta de modelos híbridos. Los datos de los tweets y las revisiones se

introdujeron en el modelo BERT para generar los vectores de características, que fue la entrada a los modelos híbridos que realizaron la clasificación.

El siguiente paso combinó los modelos CNN y LSTM de deep learning, que se utilizaron debido a su buen rendimiento en el análisis de sentimientos, así como para aprovechar las dos arquitecturas de red al realizar análisis de sentimientos en datos de diferentes dominios.

La etapa final fue la clasificación. Usamos la función de activación Relu en lugar de Sigmoid debido a la alta convergencia. Además, se eligió SVM para la clasificación debido a su eficiencia en el procesamiento de palabras, especialmente en contextos de alta dimensionalidad, como el procesamiento del lenguaje natural. La máquina de vectores de soporte es un algoritmo de aprendizaje automático supervisado que se puede utilizar para tareas de clasificación y regresión. Ha sido ampliamente explotado con resultados positivos en muchas áreas. En nuestra investigación, hemos aplicado SVM lineales para la clasificación con los modelos híbridos de deep learning propuestos. Extraemos los vectores de características de la capa superior oculta y con ellos alimentamos SVM que los clasificará para la predicción ("positivo", "negativo").

Propusimos cuatro modelos híbridos de deep learning sobre variaciones en el uso de CNN y LSTM en capas de deep learning y variaciones de CNN y SVM en las capas de clasificador. La arquitectura de estos modelos híbridos se analiza a continuación.

- El primer modelo híbrido combina los modelos CNN y LSTM. La función incrustación es la capa de incrustación que se inicializa con pesos aleatorios y que aprenderá la incrustación de todas las palabras en los conjuntos de datos de entrenamiento. La primera capa del modelo híbrido es la CNN, que recibe el vector producido por la incrustación de palabras. Tiene tres capas de convolución que constan de 512, 256 y 128 filtros respectivamente, con un tamaño de kernel = 3, que recibe y procesa datos antes de introducirlos en la siguiente capa de deep learning. La segunda capa del modelo híbrido es el LSTM, que produce una matriz de 1x500 que se alimenta al clasificador. A continuación, el clasificador del modelo híbrido se compone de dos capas continuas y completamente conectadas con 128 nodos y, finalmente, la capa de salida con una función de activación Relu.
- El segundo modelo híbrido combina los modelos LSTM y CNN. Los datos de entrada se preprocesan para remodelar los datos de la matriz de incrustación. La primera capa del modelo híbrido es la capa LSTM. Esa salida tiene una matriz de 13x500 y se introduce en el segundo modelo del modelo híbrido de deep learning. La siguiente capa del modelo híbrido es la CNN. Tiene tres capas de convolución que constan de 512, 256 y 128 filtros respectivamente, con un tamaño de kernel = 3, que se encargan de recibir y procesar los datos antes de introducirlos en la siguiente capa. La salida de CNN se aplana y se transfiere a una capa completamente conectada. Y finalmente, el clasificador del modelo híbrido es una CNN compuesta por dos capas continuas completamente conectadas con 128 nodos y la función de activación Relu como capa de salida.
- Nuestro modelo híbrido final se basa en los modelos híbridos de los escenarios 1 y 2. Usamos las etapas de deep learning de esos modelos (CNN-LSTM y LSTM-CNN) pero reemplazamos el clasificador. Si bien existen múltiples alternativas a la función Relu basada en CNN utilizada, hemos optado por usar SVM para el clasificador de reemplazo. El escenario 3 se basa en CNN-LSTM y el escenario 4 se basa en LSTM-CNN.

3.2.4. Resultados

Esta contribución tiene como objetivo construir un modelo híbrido de aprendizaje profundo para el análisis de sentimientos que funcione bien en varios conjuntos de datos de diferentes dominios. Evaluamos y validamos la combinación de CNN, LSTM y SVM, considerando la relación entre los modelos y sus capacidades avanzadas para extraer características, almacenar información pasada y nodos, y clasificar texto. Los resultados obtenidos son positivos y confiables porque se han evaluado en muchos conjuntos de datos con diferentes temas. Los resultados específicos con respecto a los modelos híbridos para el análisis de sentimientos se resumen a continuación:

- Los modelos híbridos aumentaron la precisión en el análisis de sentimientos en comparación con el rendimiento de un solo modelo en todos los tipos de conjuntos de datos, aunque el tiempo de cálculo de los modelos SVM es mayor.
- El uso de BERT previamente entrenado produce mejores resultados que el uso de Word2vec para el análisis de sentimientos con todos los modelos y todos los conjuntos de datos.
- La combinación ayudó a aprovechar las fortalezas de CNN, LSTM y SVM: CNN tiene la capacidad de extraer características, LSTM tiene la capacidad de almacenar información pasada en los nodos estatales y SVM tiene la capacidad de clasificar.
- El uso de SVM como método de clasificación mejoró los resultados de LSTM-CNN y CNN-LSTM. SVM es eficaz en la estratificación de datos multidimensionales y ayuda a minimizar los mínimos locales de las redes neuronales.

3.2.5. Conclusiones

Propusimos el uso de modelos híbridos de deep learning para el análisis de sentimientos a partir de datos de redes sociales. Probamos el rendimiento de combinaciones de SVM, CNN y LSTM con técnicas de incrustación de dos palabras, word2vec y BERT, en ocho conjuntos de datos textuales de tweets y reseñas. Después de eso, comparamos cuatro modelos híbridos generados con modelos únicos. Estos experimentos se realizaron para comprender la adaptabilidad de los modelos híbridos, específicamente si los enfoques híbridos se pueden adaptar a una amplia gama de tipos y tamaños de conjuntos de datos. Estudiamos la influencia de diferentes tipos de conjuntos de datos, técnicas de extracción de características y modelos de deep learning en la fiabilidad del análisis de polaridad de sentimiento.

Nuestros experimentos revelaron que la fiabilidad de los modelos híbridos superó a todos los modelos probados para el análisis de polaridad de sentimiento. La combinación de modelos de *deep learning* con la técnica SVM produce mejores resultados que el uso de un modelo individual para realizar análisis de sentimientos. En la mayoría de los conjuntos de datos probados, la fiabilidad de los modelos híbridos que usan SVM es mayor que en aquellos que no lo usan; sin embargo, el tiempo de cálculo es mucho más largo para aquellos con SVM. También observamos que la efectividad de los algoritmos depende en gran medida de las características y la calidad de los conjuntos de datos.

3.3. Contribución 3

3.3.1. Referencia

Dang, Cach N., María N. Moreno-García, and Fernando De la Prieta. "An Approach to Integrating Sentiment Analysis into Recommender Systems", Sensors 21.16 (2021): 5666.

3.3.2. Objetivos

El objetivo del trabajo presentado en este artículo es proponer y evaluar un sistema de recomendación que integre análisis de sentimientos y métodos de filtrado colaborativo. El método para construir el sistema se basa en una arquitectura adaptativa, que incluye técnicas mejoradas para la extracción de características y modelos de *deep learning* para el análisis de sentimientos. En este estudio se utilizaron dos conjuntos de datos de reseñas de alimentos de Amazon y reseñas de libros y música.

3.3.3. Modelado del efecto del tiempo en las recomendaciones

Usamos la combinación de varios enfoques exitosos. Comenzamos usando un modelo BERT previamente entrenado para crear los vectores de características. Luego variamos el orden de los modelos CNN y LSTM usados en las siguientes etapas: BERT → CNN → LSTM o BERT → LSTM → CNN. La etapa final del modelo utiliza una función de activación ReLu. Rotulamos las reseñas con un valor de una escala ordinal de cinco clases (muy negativo; negativo; neutral; positivo y muy positivo), análogo a las valoraciones explícitas de los usuarios, para entrenar y validar el resultado del análisis de sentimiento.

El método de recomendación propuesto es un enfoque de filtrado colaborativo basado en el usuario que considera las valoraciones explícitas y el análisis de opiniones extraídos de las reseñas de los usuarios. Como métodos de filtrado colaborativo probamos SVD (Singular Value Decomposition), NMF (Non-negative Matrix Factorization) y SVD ++ (un derivado de SVD). El objetivo es lograr una mejor precisión predictiva debido a la adición de información de retroalimentación implícita proporcionada por el análisis de sentimiento.

Los resultados del método de recomendación de filtrado colaborativo y el análisis de sentimientos se combinaron para generar una valoración y se utilizaron para crear una lista de recomendaciones.

Dada una matriz de valoraciones $R_{m \times n}$ $(\mathbb{N})$, donde m es el número de usuarios y n es el número de elementos, $r_{ij} \in R_{m \times n}$ denota la valoración que el usuario u_i da al elemento i_j.

La calificación del usuario u_a en el elemento i_j en el conjunto de prueba se predice de la siguiente manera:

$$pr_{aj} = \beta * pr\_mf_{aj} + (1 - \beta) * pr\_sent_{aj} \quad (1)$$

Dónde:

pr_mf_{aj}: Valoración para el usuario u_a y el elemento i_j predicho por los métodos de factorización matricial (SVD, SVD ++ y NMF) sin usar sentimientos

pr_sent_{aj}: Valoración para el usuario u_a y el elemento i_j se predicha mediante el modelo de opinión.

β: parámetro utilizado para ajustar la importancia de cada término de la ecuación.

Como se mencionó anteriormente, los modelos de sentimiento híbridos se utilizan para clasificar cada reseña en una de las cinco clases posibles. Estas clases se convierten en puntuaciones de sentimiento de 1 a 5, análogas a las puntuaciones explícitas asignadas por los usuarios a los items. En primer lugar, para cada usuario u_a, encontramos todos los elementos que el usuario u_a ya calificó y cuya puntuación de opinión de la revisión correspondiente coincide con la valoración explícita. En segundo lugar, para cada elemento i_j, también encontramos a todos los usuarios que ya valoraron el elemento i_j y el elemento i_k (que se encuentran en el primer paso) en el conjunto de entrenamiento y cuyas puntuaciones de revisión también coinciden con las valoraciones explícitas.

A continuación, se utilizan dos listas de datos, que incluyen elementos y usuarios que se crean a partir del paso 1 y el paso 2, para predecir la valoración del usuario u_a para cada elemento i_j. Para hacer eso, calculamos la similitud entre usuarios aplicando la métrica del coseno. Luego, aplicamos la Ecuación (2) para la predicción de valoración basada en la similitud del usuario. Las valoraciones de los k usuarios más similares se utilizan para estimar las preferencias del usuario activo u_a sobre el elemento i_j que no ha valorado.

$$pr_{aj} = \overline{r_a} + \frac{\sum_{i=1}^{K} Sim(u_a, u_i)(r_{ij} - \overline{r_i})}{\sum_{i=1}^{K} |Sim(u_a, u_i)|} \quad (2)$$

Donde r_{ij} es la valoración que el usuario u_i otorga al elemento i_j respectivamente; $\overline{r_a}$ y $\overline{r_i}$ son las valoraciones promedio del usuario u_a y del usuario u_i, respectivamente; y $Sim(u_a, u_i)$ es la similitud entre el usuario activo u_a y su vecino u_i, la cual se obtendría utilizando la métrica del coseno (Ecuación (3)). En nuestro caso, los vecinos del usuario u_a son usuarios que han calificado los mismos elementos que el usuario u_a de manera similar o la puntuación de sus reseñas sobre los mismos elementos es similar.

$$Sim(u_a, u_i) = \frac{\sum_{j=1}^{n} r_{aj}\, r_{ij}}{\sqrt{\sum_{j=1}^{n} r_{aj}^2}\sqrt{\sum_{j=1}^{n} r_{ij}^2}} \quad (3)$$

3.3.4. Resultados

Esta contribución presenta la aplicación del análisis de sentimientos en sistemas de recomendación. Proponemos un método de recomendación que combina el análisis de opiniones y el filtrado colaborativo basado en el usuario. Realizamos experimentos con dos configuraciones diferentes con/sin análisis de sentimiento. En el primero, las recomendaciones se basan en métodos del sistema de recomendación sin sentimiento, mientras en el segundo, los

resultados de realizar análisis de sentimiento en las revisiones se incorporan al proceso de recomendación. Los resultados relacionados con la aplicación del análisis de sentimiento en el sistema de recomendación de esta tesis son, en resumen:

- Los resultados del estudio empírico realizado con dos conjuntos de datos populares muestran que los modelos de aprendizaje profundo basados en sentimientos y los métodos de filtrado colaborativo pueden mejorar significativamente la fiabilidad de los sistemas de recomendación.
- Los métodos basados en sentimientos propuestos en este trabajo brindan mejores resultados que aquellos basados únicamente en valoraciones explícitas. Estas mejoras se han producido en los dos conjuntos de datos utilizados en el estudio.
- Se probaron tres algoritmos (SVD, NMF y SVD ++) de dos formas, solo con valoraciones explícitas y combinando calificaciones explícitas con opiniones extraídas de las revisiones. En la mayoría de los casos, el enfoque combinado con los sentimientos de dos modelos de clasificación en conjuntos de datos de reseñas de alimentos y películas dio mejores resultados.
- Los resultados muestran que los valores de RSME, MAE y NMAE producidos por el enfoque que combina CF con análisis de sentimiento son mejores que las tasas de error producidas por los métodos tradicionales de CF sin sentimiento.
- Los valores de MRR, MAP y NDCG muestran que el método propuesto también mejora las recomendaciones top-N.

3.3.5. Conclusiones

Hemos propuesto una aplicación de análisis de sentimiento en sistemas de recomendación que se basa en modelos híbridos de *deep learning* y filtrado colaborativo en redes sociales online. La arquitectura del sistema presentada en este trabajo, puede integrar una variedad de técnicas que se han propuesto para realizar recomendaciones, incluida la estrategia de preprocesamiento, modelos híbridos de deep learning para análisis de sentimientos y métodos para sistemas de recomendación. La arquitectura se puede utilizar para desarrollar un sistema de recomendación en el contexto de las redes sociales que aproveche el análisis de sentimiento realizado sobre las opiniones y reseñas de los usuarios en la red. Realizamos experimentos con reseñas de comida y películas. Con base en tales experimentos, demostramos la utilidad y aplicabilidad de nuestros enfoques para producir recomendaciones personalizadas en las redes sociales en línea.

Los resultados mostraron que el uso conjunto de análisis de sentimientos basados en *deep learning* y métodos de filtrado colaborativo mejoró significativamente el rendimiento de los últimos. Esto se logra mediante la explotación de información adicional de los datos de opiniones/comentarios de los usuarios. Su integración en los métodos de recomendación tradicionales hace que el sistema de recomendación sea más confiable y capaz de brindar mejores recomendaciones a los usuarios.

3.4. Contribución 4

3.4.1. Referencia

Dang, Nhan Cach, Fernando De la Prieta, Juan Manuel Corchado, and María N. Moreno. "Framework for retrieving relevant contents related to fashion from online social network data." In International Conference on Practical Applications of Agents and Multi-Agent Systems, pp. 335-347. Springer, Cham, 2016.

3.4.2. Objetivos

El propósito de este trabajo es la propuesta de un marco integral para recuperar contenidos relevantes de los datos de redes sociales. Nuestro enfoque se basa en modelos de espacio vectorial y máquinas de vectores de soporte para procesar y clasificar datos de texto sin procesar. Lo experimentos de validación pretenden demostrar la utilidad y precisión del marco propuesto para recuperar contenidos de Twitter y Facebook relacionados con la moda.

3.4.3. Marco para recuperar información de la red social

Nuestro proceso marco es el siguiente: utilizamos una medida para determinar la importancia de una palabra en el texto, llamada TF-IDF, y transformamos el texto del documento en el Modelo de Espacio Vectorial. Este modelo permite que el texto sea representado por un vector en un espacio n-dimensional, correspondiendo cada dimensión al índice. En este espacio, cada componente del vector de texto representa la medida ponderada del índice correspondiente a ese texto. Primero, en base a la representación de los vectores en el documento, se construye el modelo de clasificación de texto a partir de un conjunto de entrenamiento mediante un algoritmo de máquina de vectores de soporte. Para crear el conjunto de entrenamiento, utilizamos lcuentas que proporciona actualizaciones sobre moda. Se han realizado experimentos en dos populares redes sociales, Facebook y Twitter, para ver la eficiencia y precisión del modelo inducido a partir de la información extraída sobre moda.

Nuestro marco entrena automáticamente un modelo para recuperar datos de moda de Facebook y Twitter. Básicamente, hay dos etapas para recuperar información en este marco (1) entrenamiento del modelo y (2) recuperación de contenido de moda utilizando el modelo de clasificación inducido.

En la etapa de entrenamiento del modelo, los datos de las redes sociales (como comentarios y publicaciones) se recopilan en tiempo real mediante el uso de REST y el protocolo Graph (como API REST en Twitter y API Graph en Facebook). Algunas cuentas populares o páginas de fans que proporcionan el estado de la moda (cuentas anteriores de www.Fashionista.com y www.elle.com en Twitter y Facebook) se utilizan para entrenar un modelo máquina de vectores de soporte. Los datos de texto se transforman en el modelo de vector espacial utilizando la medida TF-IDF. Utilizando el modelo creado a partir de datos de entrenamiento, se clasifica la nueva información de la red social y se recupera y guarda la información relacionada con la moda. En la siguiente etapa, también recopilamos datos de cuentas en Facebook y Twitter que no están relacionadas

con la moda como: economía; clima; tráfico; tecnología. Estos datos se combinan con datos de moda relacionados para probar la precisión del modelo SVM.

3.4.4. Resultados

Esta contribución presenta un marco integral con el propósito de recuperar automáticamente el contenido de un tema específico. Nuestro enfoque se propone sobre la base del Modelo de espacio vectorial y la Máquina de vectores de soporte para procesar y clasificar datos de texto sin procesar. Aunque recientemente se ha estudiado SVM para problemas de clasificación de texto, su aplicación para recuperar y analizar contenido relevante relacionado con la moda a partir de datos de redes sociales en línea no se ha examinado exhaustivamente. Nuestros experimentos en dos redes sociales populares, Facebook y Twitter, demuestran la utilidad y precisión del marco en la información extraída sobre la moda.

3.4.5. Conclusiones

En este trabajo se ha desarrollado un marco para recuperar automáticamente contenidos relevantes relacionados con la moda a partir de datos de redes sociales en línea. Además, empleamos técnicas de aprendizaje automático, para clasificar los contenidos de moda. El marco probado con conjuntos de datos de Facebook y Twitter muestra un buen rendimiento.

Acknowledgement/Support

This work was supported by the Ministry of Science, Innovation and Universities and the Ministry of education and culture of the Junta de Castilla y León, Spain. The details are as follow:

- "InEDGEMobility: Towards Sustainable Intelligent Mobility Supported by Multi-Agent Systems and Edge Computing" (Reference: RTI2018-095390-B-C32) funded by Ministry of Science, Innovation and Universities. Projects of the State Programme for R+D+i oriented towards the challenges of society.
- Recommender systems for streaming services based on social media data analysis using machine learning techniques (Reference: SA064G19). Ministry of education and culture of the Junta de Castilla y León, Spain.

APPENDIX. Copy of the contributions

electronics

MDPI

Article

Sentiment Analysis Based on Deep Learning: A Comparative Study

Nhan Cach Dang [1], María N. Moreno-García [2] and Fernando De la Prieta [3,*]

1 Department of Information Technology, HoChiMinh City University of Transport (UT-HCMC), Ho Chi Minh 70000, Vietnam; tucach@hcmutrans.edu.vn
2 Data Mining (MIDA) Research Group, University of Salamanca, 37007 Salamanca, Spain; mmg@usal.es
3 Biotechnology, Intelligent Systems and Educational Technology (BISITE) Research Group, University of Salamanca, 37007 Salamanca, Spain
* Correspondence: fer@usal.es; Tel.: +34-677-522-678

Received: 31 January 2020; Accepted: 10 March 2020; Published: 14 March 2020

Abstract: The study of public opinion can provide us with valuable information. The analysis of sentiment on social networks, such as Twitter or Facebook, has become a powerful means of learning about the users' opinions and has a wide range of applications. However, the efficiency and accuracy of sentiment analysis is being hindered by the challenges encountered in natural language processing (NLP). In recent years, it has been demonstrated that deep learning models are a promising solution to the challenges of NLP. This paper reviews the latest studies that have employed deep learning to solve sentiment analysis problems, such as sentiment polarity. Models using term frequency-inverse document frequency (TF-IDF) and word embedding have been applied to a series of datasets. Finally, a comparative study has been conducted on the experimental results obtained for the different models and input features.

Keywords: sentiment analysis; deep learning; machine learning; neural network; natural language processing

1. Introduction

Web 2.0 has led to the emergence of blogs, forums, and online social networks that enable users to discuss any topic and share their opinions about it. They may, for example, complain about a product that they have bought, debate current issues, or express their political views. Exploiting such information about users is key to the operation of many applications (such as recommender systems), in the survey analyses conducted by organizations, or in the planning of political campaigns. Moreover, analyzing public opinions is also very important to governments because it explains human activity and behavior and how they are influenced by the opinions of others. In the area of recommender systems and personalization, the inference of user sentiment can be very useful to make up for the lack of explicit user feedback on a provided service. In addition to machine learning, other methods, such as those based on the similarity of results, can be used for this purpose [1]. The sources of data for sentiment analysis (SA) are online social media, the users of which generate an ever-increasing amount of information. Thus, these types of data sources must be considered under the big data approach, given that additional issues must be dealt with to achieve efficient data storage, access, and processing, and to ensure the reliability of the obtained results [2].

The problem of automatic sentiment analysis (SA) is a growing research topic. Although SA is an important area and already has a wide range of applications, it clearly is not a straightforward task and has many challenges related to natural language processing (NLP). Recent studies on sentiment

analysis continue to face theoretical and technical issues that hinder their overall accuracy in polarity detection [3,4]. Hussein et al. [4] studied the relationship between those issues and the sentiment structure, as well as their impact on the accuracy of the results. This work verifies that accuracy is a matter of high concern among the latest studies on sentiment analysis and proves that it is affected by some challenges, such as addressing negation or domain dependence.

Social media are important sources of data for SA. Social networks are continuously expanding, generating much more complex and interrelated information.

In this context, Thai et al. suggested not to focus solely on the structure and correlations of data, but on a lifelong learning approach to dealing with data presentation, analysis, inference, visualization, search and navigation, and decision making in complex networks [2].

Several studies focus on building powerful models to solve the continuously increasing complexity of big data, as well as to expand sentiment analysis to a wide range of applications, from financial forecasting [5,6] and marketing strategies [7] to medicine analysis [8,9] and other areas [10–18]. However, few of them pay attention to evaluating different deep learning techniques in order to provide practical evidence of their performance [5,17,19,20].

When examining the performance of a single method on a single dataset in a particular domain, the results show a relatively high overall accuracy [15,19,20] for Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). Hassan and Mahmood [15] proved that CNN and RNN models can overcome shortcoming of short text in deep learning models. Qian et al. [10] showed that Long Short-Term Memory (LSTM) behaves efficiently when used on different text levels of weather-and-mood tweets.

Li et al. [17] studied the impact of data quality on sentiment classification performance. They considered three criteria, namely informativeness, readability, and subjectivity, to assess the quality of online product reviews. The study highlighted two factors that affect the level of accuracy of sentiment analysis—readability and length of the reviews. Higher readability and shorter text datasets yielded higher quality of sentiment classification. However, when the size or domain of the data varies, the reliability of the proposed method is questionable

In comparison studies, most papers focus on reliability metrics, such as overall accuracy or F-score, and leave out processing time. In addition, the evaluations of the models are conducted on a small number of datasets. This research addresses that gap by means of a comprehensive comparison of sentiment analysis methods in the literature, and an experimental study to evaluate the performance of deep learning models and related techniques on datasets about different topics. Our research question aims to determine whether it is possible to present outperforming methods for multiple types and sizes of datasets. We build upon on previous studies of improvement of SA performance by evaluating the results from the viewpoint of a combination of three criteria: overall accuracy, F-score, and processing time. The purpose of this comparative study is to give an objective overview of different techniques that can guide researchers towards the achievement of better results

In recent years, several studies have proposed deep-learning-based sentiment analyses, which have differing features and performance. This work looks at the latest studies that have used deep learning models, such as deep neural networks (DNN), recurrent neural networks (RNN), and convolutional neural networks (CNN), to solve different problems related to sentiment analysis (e.g., sentiment polarity and aspect-based sentiment). We applied deep learning models with TF-IDF and word embedding to Twitter datasets and implemented the state-of-the-art of sentiment analysis approaches based on deep learning.

The rest of this paper is organized as follows. Section 2 provides background knowledge on this research area. Section 3 discusses related work. Section 4 describes the comparative study. Section 5 outlines the experimental results, followed by the conclusions in Section 6.

2. Background

2.1. Deep Learning

Deep learning adapts a multilayer approach to the hidden layers of the neural network. In traditional machine learning approaches, features are defined and extracted either manually or by making use of feature selection methods. However, in deep learning models, features are learned and extracted automatically, achieving better accuracy and performance. In general, the hyper parameters of classifier models are also measured automatically. Figure 1 shows the differences in sentiment polarity classification between the two approaches: traditional machine learning (Support Vector Machine (SVM), Bayesian networks, or decision trees) and deep learning. Artificial neural networks and deep learning currently provide the best solutions to many problems in the fields of image and speech recognition, as well as in natural language processing. Several types of deep learning techniques are discussed in this section.

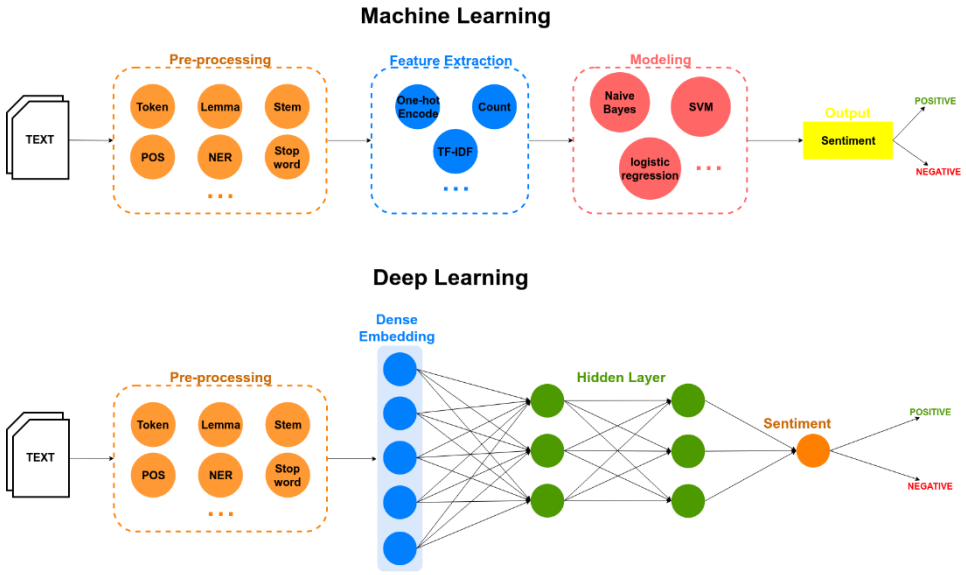


Figure 1. Differences between two classification approaches of sentiment polarity, machine learning (top), and deep learning (bottom). Part of Speech (POS); Named Entity Recognition (NER); Term Frequency-Inverse Document Frequency (TF-IDF).

2.1.1. Deep Neural Networks (DNN)

A deep neural network [21] is a neural network with more than two layers, some of which are hidden layers (Figure 2). Deep neural networks use sophisticated mathematical modeling to process data in many different ways. A neural network is an adjustable model of outputs as functions of inputs, which consists of several layers: an input layer, including input data; hidden layers, including processing nodes called neurons; and an output layer, including one or several neurons, whose outputs are the network outputs.

2.1.2. Convolutional Neural Networks (CNN)

A convolutional neural network is a special type of feed-forward neural network originally employed in areas such as computer vision, recommender systems, and natural language processing. It is a deep neural network architecture [22], typically composed of convolutional and pooling or subsampling layers to provide inputs to a fully-connected classification layer. Convolution layers filter their inputs to extract features; the outputs of multiple filters can be combined. Pooling or subsampling layers reduce the resolution of features, which can increase the CNN's robustness to noise and distortion. Fully connected layers perform classification tasks. An example of a CNN

architecture can be seen in Figure 3. The input data was preprocessed to reshape it for the embedding matrix. The figure shows an input embedding matrix processed by four convolution layers and two max pooling layers. The first two convolution layers have 64 and 32 filters, which are used to train different features; these are followed by a max pooling layer, which is used to reduce the complexity of the output and to prevent the overfitting of the data. The third and fourth convolution layers have 16 and 8 filters, respectively, which are also followed by a max pooling layer. The final layer is a fully connected layer that will reduce the vector of height 8 to an output vector of one, given that there are two classes to be predicted (Positive, Negative).

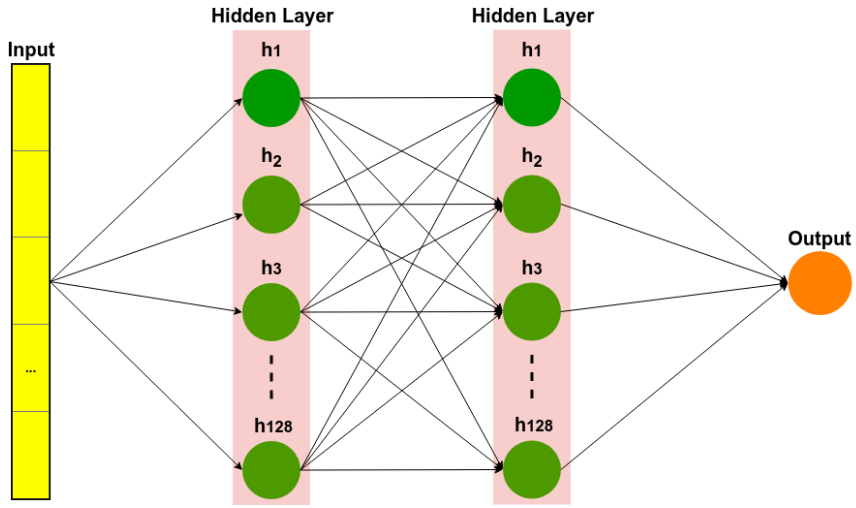


Figure 2. Deep neural network (DNN).

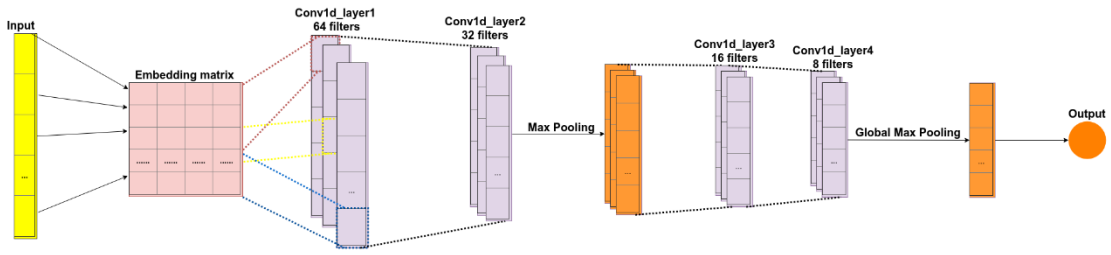


Figure 3. A convolutional neural network.

2.1.3. Recurrent Neural Networks (RNN)

Recurrent neural networks [23] are a class of neural networks whose connections between neurons form a directed cycle, which creates feedback loops within the RNN. The main function of RNN is the processing of sequential information on the basis of the internal memory captured by the directed cycles. Unlike traditional neural networks, RNN can remember the previous computation of information and can reuse it by applying it to the next element in the sequence of inputs. A special type of RNN is long short-term memory (LSTM), which is capable of using long memory as the input of activation functions in the hidden layer. This was introduced by Hochreiter and Schmidhuber (1997) [24]. Figure 4 illustrates an example of the LSTM architecture. The input data is preprocessed to reshape data for the embedding matrix (the process is similar to the one described for the CNN). The next layer is the LSTM, which includes 200 cells. The final layer is a fully connected layer, which includes 128 cells for text classification. The last layer uses the sigmoid activation function to reduce the vector of height 128 to an output vector of one, given that there are two classes to be predicted (positive, negative).

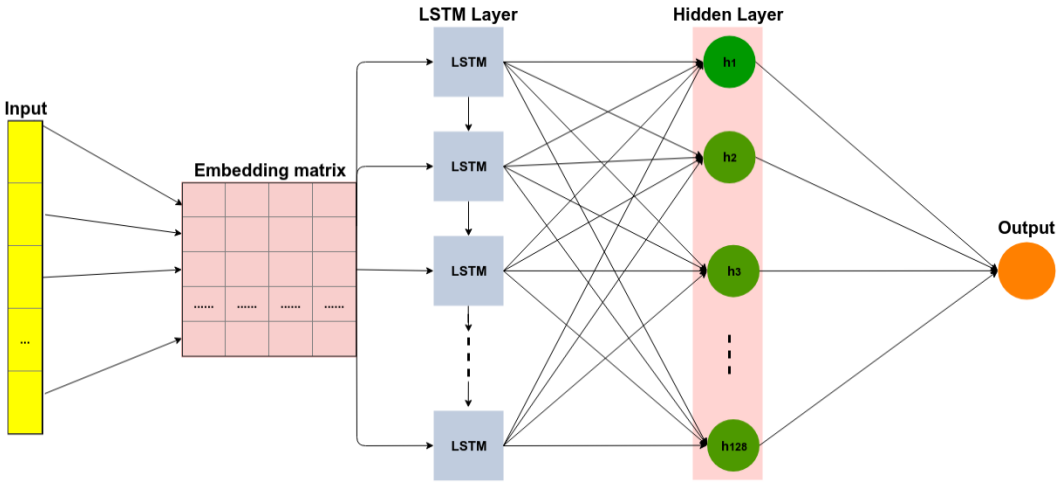


Figure 4. A long short-term memory network. LSTM, long short-term memory.

2.1.4. Other Neural Networks

One type of deep neural network is called a deep belief network (DBN) [25]. It comprises multiple layers of a graphical model, having both directed and undirected edges. Each network is composed of multiple layers of hidden units and each layer is connected to the next one, but the units within a layer are not connected. A DBN is learned by using a greedy layer-wise learning algorithm.

A recursive neural network (RecNN) [26] is a type of neural network that can be viewed as a generalization of RNN. Recursive neural networks are usually used to learn a directed acyclic graph structure from data. The hidden state vectors of the left and right child nodes in the graph can be used to compute for the hidden state vector of the current node.

Another category is hybrid deep learning [27], which combines two or more deep learning techniques together, such as convolutional neural networks (CNN) and long short-term memory (LSTM) [28], or probabilistic neural networks (PNN) and a two-layered restricted Boltzmann machine (RBM) [29].

2.2. Sentiment Analysis

Sentiment analysis is a process of extracting information about an entity and automatically identifying any of the subjectivities of that entity. The objective is to determine whether text generated by users conveys their positive, negative, or neutral opinions. Sentiment classification can be carried out on three levels of extraction: the aspect or feature level, the sentence level, and the document level. Currently, there are three approaches to address the problem of sentiment analysis [30]: (1) lexicon-based techniques, (2) machine-learning-based techniques, and (3) hybrid approaches.

Lexicon-based techniques were the first to be used for sentiment analysis. They are divided into two approaches: dictionary-based and corpus-based [31]. In the former type, sentiment classification is performed by using a dictionary of terms, such as those found in SentiWordNet and WordNet. Nevertheless, corpus-based sentiment analysis does not rely on a predefined dictionary but on statistical analysis of the contents of a collection of documents, using techniques based on k-nearest neighbors (k-NN) [32], conditional random field (CRF) [33], and hidden Markov models (HMM) [34], among others.

Machine-learning-based techniques [35] proposed for sentiment analysis problems can be divided into two groups: (1) traditional models and (2) deep learning models. Traditional models refer to classical machine learning techniques, such as the naïve Bayes classifier [36], maximum entropy classifier [37,38], or support vector machines (SVM) [39]. The input to those algorithms includes lexical features, sentiment lexicon-based features, parts of speech, or adjectives and adverbs. The accuracy of these systems depends on which features are chosen. Deep learning models can provide better

results than traditional models. Different kinds of deep learning models can be used for sentiment analysis, including CNN, DNN, and RNN. Such approaches address classification problems at the document level, sentence level, or aspect level. These deep learning methods will be discussed in the following section.

The **hybrid approaches** [40] combine lexicon- and machine-learning-based approaches. Sentiment lexicons commonly play a key role within a majority of these strategies. Figure 5 illustrates a taxonomy of deep-learning-based methods for sentiment analysis.

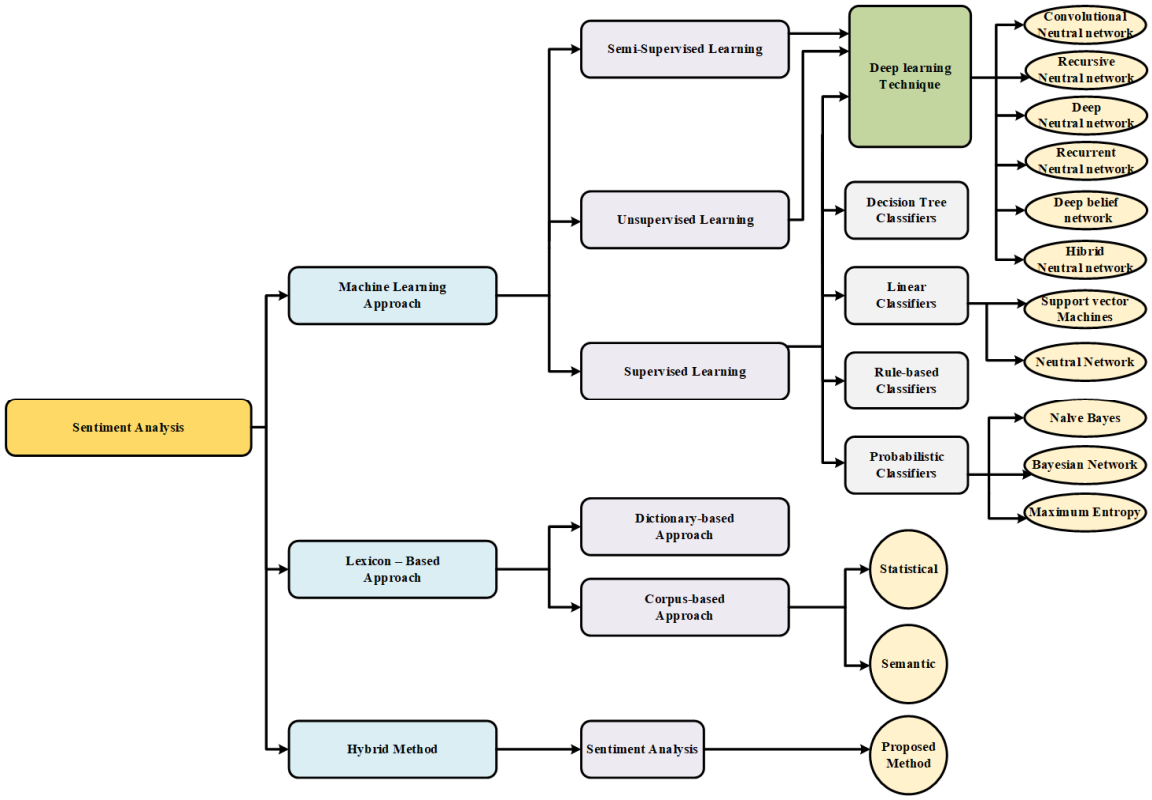


Figure 5. Taxonomy of sentiment analysis techniques. Source: [30,41].

Sentiment analysis, whether performed by means of deep learning or traditional machine learning, requires that text training data be cleaned before being used to induce the classification model. Tweets usually contain white spaces, punctuation marks, non-characters, Retweet (RT), "@ links", and stop words. These characters could be removed using libraries such as BeautifulSoup because they do not contain any information that would be useful for sentiment analysis. After cleaning, tweets can be split into individual words, which are transformed into their base form by lemmatization, then converted into numerical vectors by using methods such as word embedding or term frequency-inverse document frequency (TF-IDF).

Word embedding [42] is a technique for language modeling and feature learning, where each word is mapped to a vector of real values in such a way that words with similar meanings have a similar representation. Value learning can be done using neural networks. A commonly used word embedding system is Word2vec (GloVe, or Gensim), which contains models such as skip-gram and continuous bag-of-words (CBOW). Both models are based on the probability of words occurring in proximity to each other. Skip-gram makes it possible to start with a word and predict the words that are likely to surround it. Continuous bag-of-words reverses that by predicting a word that is likely to occur on the basis of specific context words.

TF-IDF is a statistical measure reflecting how important a word is to a document in a collection or corpus. This metric considers the frequency of the word in the target document, as well as the

frequency in the other documents of the corpus. The higher the frequency of a word in a target document and the lower its frequency in other documents, the greater its importance. The vectorizer class in the scikit-learn library is usually used to compute TF-IDF.

Both word embedding and TF-IDF are used as input features of deep learning algorithms in NLP. Sentiment analysis tasks transform collections of raw data into vectors of continuous real numbers.

There are different kinds of tasks, such as objective or subjective classification, polarity sentiment detection, and feature- or aspect-based sentiment analysis. The subjectivity of words and phrases may depend on their context and an objective document may contain subjective sentences. Aspect-based sentiment analysis refers to sentiments expressed towards specific aspects of entities (e.g., value, room, location, cleanliness, or service). Polarity and intensity are two components used to score sentiment analysis. Polarity indicates whether the sentiment is negative, neutral, or positive. Intensity indicates the relative strength of the sentiment.

2.3. Application of Sentiment Analysis

It is widely accepted that sentiment analysis is very useful in a wide range of application domains, such as business, government, and biomedicine.

In the fields of business intelligence and e-commerce, companies can study customers' feedback to provide better customer support, build better products, or improve their marketing strategies to attract new customers. Sentiment analysis can be used to infer the users' opinions on events or products. The results of SA help to gain greater insight into the customers' interests or opinions on industrial trends. In this context, Jain and Dandannavar [43] proposed a fast, flexible, and scalable SA framework for sentiment analysis of Twitter data that involves the use of some machine learning methods and Apache spark.

As pointed out in the introduction, the area of recommender systems has also benefited from sentiment analysis. A sample of this can be found in the work of Preethi et al. [12], where recursive neural networks were applied to analyze sentiments in reviews. The output was used to improve and validate the restaurant and movie recommendations of a cloud-based recommender system. Along with behavioral analysis, sentiment analysis is also an efficient tool for commodity markets [7].

The medical domain is another field of potential interest. The applications of opinion mining in health-related texts on social media and blogs were explored in [8]. In addition to traditional machine learning and text processing techniques, the author offers new approaches and proposes a medical lexicon to support experts and patients in the varied methodology that is used to describe symptoms and diseases. In the field of mental health, sentiment analysis is performed on texts written by patients' posts on social media as a means of supplementing or replacing the questionnaires they usually fill in [9].

3. Related Work

The purpose of this study is to review different approaches and methods in sentiment analysis that can be taken as a reference in future empirical studies. We have focused on key aspects of research, such as technical challenges, datasets, the methods proposed in each study, and their application domains.

Recently, deep learning models (including DNN, CNN, and RNN) have been used to increase the efficiency of sentiment analysis tasks. In this section, state-of-the-art sentiment analysis approaches based on deep learning are reviewed.

Beginning in 2015, many authors have since evaluated this trend. Tang et al. [44] introduced techniques based on deep learning approaches for several sentiment analyses, such as learning word embedding, sentiment classification, and opinion extraction. Zhang and Zheng [35] discussed machine learning for sentiment analysis. Both research groups used part of speech (POS) as a text feature and used TF-IDF to calculate the weight of words for the analysis. Sharef et al. [45] discussed the opportunities of sentiment analysis approaches for big data. In papers [13,18,46], the latest

deep-learning-based techniques (namely CNN, RNN, and LSTM) were reviewed and compared with each other in the context of sentiment analysis problems.

Some other studies applied deep-learning-based sentiment analysis in different domains, including finance [5,6], weather-related tweets [10], trip advisors [11], recommender systems for cloud services [12], and movie reviews [13–18]. In [10], where text features were automatically extracted from different data sources, user information and weather knowledge were transferred into word embedding using the Word2vec tool. The same techniques have been used in several works [5,47]. Jeong et al. [48] identified product development opportunities by combining topic modeling and the results of a sentiment analysis that had been performed on customer-generated social media data. It has been used as a real-time monitoring tool for analysis of changing customer needs in rapidly evolving product environments. Pham et al. used multiple layers of knowledge representation to analyze travel reviews and determine sentiments for five aspects, including value, room, location, cleanliness, and service [11]. Another approach [49] combines sentiment and semantic features in an LSTM model based on emotion detection. Preethi et al. [12] applied deep learning to sentiment analysis for a recommender system in the cloud using the food dataset from Amazon. For the health domain, Salas-Zárate et al. [31] applied an ontology-based, aspect-level sentiment analysis method to tweets about diabetes.

Polarity-based sentiment deep learning applied to tweets was found in [19,20,28,36,40,50]. The authors described how they used deep learning models to increase the accuracy of their respective sentiment analysis. Most of the models are used for content written in English, but there are a few that manage tweets in other languages, including Spanish [51], Thai [28], and Persian [47]. Previous researchers have analyzed tweets by applying different models of polarity-based sentiment deep learning. Those models include DNN [50], CNN [20], and hybrid approaches [40].

Other works using neural network models are focused not only on the sentiment polarity of textual content, but also on aspect sentiment analysis [6,11,31,52–54]. Salas-Zárate et al. [31] used semantic annotation (diabetes ontology) to identify aspects from which they performed aspect-based sentiment analysis using SentiWordNet. Pham et al. [11] included the determination of sentiment ratings and importance degrees of product aspects. A novel, multilayer architecture was proposed to represent customer reviews aiming at extracting more effective sentiment features.

From among 32 of the analyzed studies, we identified three popular models for sentiment polarity analysis using deep learning: DNN [50], CNN [20], and hybrid [40]. In [13,18,46], three deep learning techniques, namely CNN, RNN, and LSTM, were individually tested on different datasets. However, there was a lack of a comparative analysis of these three techniques.

Many studies use the same process for sentiment analysis. First, text features are automatically extracted from different data sources, then they are transferred into word embedding using the Word2vec tool [5,10,47].

Sentiment analysis has also been the target of extensive research in the application domain of recommender systems. Most methods in this area are based on information filtering, and they can be classified into four categories: content-based, collaborative filtering (CF), demographic-based, and hybrid. Social media data can be used with these techniques in different ways. Content-based methods make use of characteristics of items and user's profiles, CF methods require implicit or explicit user preferences, demographic methods exploit user demographic information (age, gender, nationality, etc.), and hybrid approaches take advantage of any kind of item and user information that can be extracted or inferred from social media (actions, preferences, behavior, etc.).

Besides, when dealing with both explicit data (which are provided directly by users) and implicit data (which are inferred from the behavior and actions of users), hybrid methods and lifelong learning algorithms are considered as in-depth approaches for recommendation systems.

Shoham [55] proposed one of the first hybrid recommendation systems, which takes advantage of both content and collaborative filtering recommendation methods. The content-based part of the proposal involves the identification of user profiles based on their interest in topics extracted from web pages, while the collaborative filtering part of the system is based on the feedback of other users.

Although sentiment analysis is not performed in this work, it can be considered the precursor of other studies combining both approaches in which sentiment analysis is used to obtain implicit user feedback. A recent study from Wang et al. [56] presents a hybrid approach in which sentiment analysis of reviews about movies is used in order to improve a preliminary recommendation list obtained from the combination of collaborative filtering and content-based methods. In the same application domain, Singh et al. propose the use of a sentiment classifier induced from movie reviews as a second filter after collaborative filtering [57].

In addition, one advanced machine learning paradigm is the so-called holistic models or lifelong learning algorithms, which are argued to significantly improve sentiment analysis accuracy [58]. While other methods learn a model by using only data for a particular application, this method attains a continually updating knowledge base of attributes, such as sentiment polarity or sentiment aspects. Stai et al. [59] introduced a social recommendation framework, whose main objective is the creation of enriched multimedia content adapted to users. This is achieved through a holistic approach, where the explicit and implicit relevance feedback from users is derived from their interactions with both the video and its enrichment. Although this method represents a significant improvement over other approaches, it requires personal user information.

Table 1 summarizes 32 important papers related to our research. It includes the year of publication, authors' names, research work, methods, datasets, and the study target.

4. Comparative Study

In this section, we begin by introducing different topics pertaining to datasets, and then we offer details about the sentiment classification process.

We used eight datasets in our experiments on sentiment polarity analysis. Three of them contain tweets; the largest has 1.6 million tweets, with each one labeled as either positive or negative sentiment, while the other two datasets contain 14,640 and 17,750 tweets, respectively, labeled as positive, negative, or neutral. The remaining five datasets include a total of 125,000 comments from user reviews of movies, books, and music labeled as either positive or negative sentiments.

Two approaches for preparing inputs to the classification algorithms are compared in our experiments: word embedding and TF-IDF. For word embedding, we applied Word2vec, which contains models such as skip-gram and continuous bag-of-words (CBOW). Skip-gram makes it possible to start with a word and predict the words that are likely to surround it. Continuous bag-of-words reverses that and enables the prediction of a word that is likely to occur in the context of words. For TF-IDF, we used the vectorizer class in the scikit-learn library.

We conducted an experimental study where three models (DNN, CNN, and RNN) were trained and evaluated on different datasets, which had been preprocessed with both word embedding and TF-IDF. The objective was to compare the performance of all these techniques and improve the state-of-the-art of sentiment analysis tasks.

Table 1. Summary of deep-learning-based sentiment analysis.

No.	Year	Study	Research Work	Method	Dataset	Target
1	2019	Alharbi el al. [19]	Twitter sentiment analysis	CNN	SemEval 2016 workshop	Feature extraction from user behavior information
2	2019	Kraus et al. [16]	Sentiment analysis based on rhetorical structure theory	Tree-LSTM and Discourse-LSTM	Movie Database (IMD), food reviews (Amazon)	Aim to improve accuracy
3	2019	Do et al. [53]	Comparative review of sentiment analysis based on deep learning	CNN, LSTM, GRU, and hybrid approaches	SemEval workshop and social network sites	Aspect extraction and sentiment classification
4	2019	Abid et al. [20]	Sentiment analysis through recent recurrent variants	CNN, RNN	Twitter	Domain-specific word embedding
5	2019	Yang et al. [52]	Aspect-based sentiment analysis	Coattention-LSTM, Coattention-MemNet, Coattention-LSTM + location	Twitter, SemEval 2014	Target-level and context-level feature extraction
6	2019	Wu et al. [60]	Sentiment analysis with variational autoencoder	LSTM, Bi-LSTM	Facebook, Chinese VA, Emobank	Encoding, sentiment prediction, and decoding
7	2018	Pham et al. [11]	Aspect-based sentiment analysis	LRNN-ASR, FULL-LRNN-ASR	Tripadvisor	Enriching knowledge of the input through layers
8	2018	Sohangir et al. [5]	Deep learning for financial sentiment analysis	LSTM, doc2vec, and CNN	StockTwits	Improving the performance of sentiment analysis for StockTwits
9	2018	Li et al. [17]	How textual quality of online reviews affect classification performance	SRN, LSTM, and CNN	Movie reviews from imdb.com	Impact of two influential textual features, namely the word count and review readability
10	2018	Zhang et al. [61]	Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression	CNN	SemEval 2016, Sentiment Tree Bank	LSTM attention and attentive pooling is integrated with CNN model to extract sentence features based on sentiment embedding, lexicon embedding, and semantic embedding

Table 1. *Cont.*

No.	Year	Study	Research Work	Method	Dataset	Target
11	2018	Schmitt et al. [54]	Joint aspect and polarity classification for aspect-based sentiment analysis	CNN, LSTM	SemEval 2017	Approach based on aspect sentiment analysis to solve two classification problems (aspect categories + aspect polarity)
12	2018	Qian et al. [10]	Sentiment analysis model on weather-related tweets	DNN, CNN	Twitter, social network sites	Feature extraction
13	2018	Tang et al. [62]	Improving the state-of-the-art in many deep learning sentiment analysis tasks	CNN, DNN, RNN	Social network sites	Sentiment classification, opinion extraction, fine-grained sentiment analysis
14	2018	Zhang et al. [22]	Survey of deep learning for sentiment analysis	CNN, DNN, RNN, LSTM	Social network sites	Sentiment analysis with word embedding, sarcasm analysis, emotion analysis, multimodal data for sentiment analysis
15	2017	Choudhary et al. [30]	Comparative study of deep-learning-based sentimental analysis with existing techniques	CNN, DNN, RNN, lexicon, hybrid	Social network sites	Domain dependency, sentiment polarity, negation, feature extraction, spam and fake review, huge lexicon, bi-polar words
16	2018	Jangid et al. [6]	Financial sentiment analysis	CNN, LSTM, RNN	Financial tweets	Aspect-based sentiment analysis
17	2017	Araque et al. [63]	Enhancing deep learning sentiment analysis with ensemble techniques in social applications	Deep-learning-based sentiment classifier using a word embedding model and a linear machine learning algorithm	SemEval 2013/2014, Vader, STS-Gold, IMDB, PL04, and Sentiment140	Improving the performance of deep learning techniques and integrating them with traditional surface approaches based on manually extracted features
18	2017	Jeong et al. [48]	A product opportunity mining approach based on topic modeling and sentiment analysis	LDA-based topic modeling, sentiment analysis, and opportunity algorithm	Twitter, Facebook, Instagram, and Reddit	Identification of product development opportunities from customer-generated social media data
19	2017	Gupta et al. [49]	Sentiment-/semantic-based approaches for emotion detection	LSTM-based deep learning	Twitter	Combining sentiment and semantic features

Table 1. *Cont.*

No.	Year	Study	Research Work	Method	Dataset	Target
20	2017	Preethi et al. [12]	Sentiment analysis for recommender system in the cloud	RNN, naïve Bayes classifier	Amazon	Recommending the places that are near to the user's current location by analyzing the different reviews and consequently computing the score grounded on it
21	2017	Ramadhani et al. [50]	Twitter sentiment analysis	DNN	Twitter	Handling a huge amount of unstructured data
22	2017	Ain et al. [13]	A review of sentiment analysis using deep learning techniques	CNN, RNN, DNN, DBN	Social network sites	Analyzing and structuring hidden information extracted from social media in the form of unstructured data
23	2017	Roshanfekr et al. [47]	Sentiment analysis using deep learning on Persian texts	NBSVM-Bi, Bidirectional-LSTM, CNN	Customer reviews from www.digikala.com	Evaluating deep learning methods using the Persian language
24	2017	Paredes-Valverde et al. [51]	Sentiment analysis for improvement of products and services	CNN + Word2vec	Twitter in Spanish	Detecting customer satisfaction and identifying opportunities for improvement of products and services
25	2017	Jingzhou Liu et al. [64]	Extreme multilabel text classification	XML-CNN	RCV1, EUR-Lex, Amazon, and Wiki	Capturing richer information from different regions of the document
26	2017	Hassan et al. [15]	Sentiment analysis of short texts	CNN, LSTM, on top of pretrained word vectors	Stanford Large Movie Review, IMDB, Stanford Sentiment Treebank, SSTb	Achieving comparable performances with fewer parameters on sentiment analysis tasks

Table 1. *Cont.*

No.	Year	Study	Research Work	Method	Dataset	Target
27	2017	Chen et al. [65]	Multimodal sentiment analysis with word-level fusion and reinforcement learning	Gated multimodal embedding LSTM with temporal attention	CMU-MOSI	Developing a novel deep architecture for multimodal sentiment analysis that performs modality fusion at the word level
28	2017	Al-Sallab et al. [66]	Opinion mining in Arabic as a low-resource language	Recursive deep learning	Online comments from QALB, Twitter, and Newswire articles written in MSA	Providing more complete and comprehensive input features for the autoencoder and performing semantic composition
29	2016	Vateekul et al. [28]	A study of sentiment analysis in Thai	LSTM, DCNN	Twitter	Finding the best parameters of LSTM and DCNN
30	2016	Singhal, et al. [18]	A survey of sentiment analysis and deep learning	CNN, RNTN, RNN, LSTM	Sentiment Treebank dataset, movie reviews, MPQA, and customer reviews	Comparison of classification performance of different models on different datasets
31	2016	Gao et al. [14]	Sentiment analysis using AdaBoost combination	CNN	Movie reviews and IMDB	Studying the possibility of leveraging the contribution of different filter lengths and grasping their potential in the final polarity of the sentence
32	2016	Rojas-Barahona et al. [46]	Overview of deep learning for sentiment analysis	CNN, LSTM	Movie reviews, Sentiment Treebank, and Twitter	To extract the polarity from the data

Gated Recurrent Units (GRU); Bi-directional Long-Short-Term-Memory (Bi-LSTM); Latent Rating Neural Network-Aspect Semantic Representation (LRNN-ASR); Simple Recurrent Networks (SRN); Latent Dirichlet Allocation (LDA); Naive Bayes and Support Vector Machine Bidirectional (NBSVM-bi); Deep Convolutional Neural Network (DCNN); Recursive Neural Tensor Network (RNTN); Multi-Perspective Question Answering (MPQA); Multimodal Opinion Sentiment Intensity (CMU-MOSI); Qatar Arabic Language Bank (QALB)

4.1. Datasets

Studies that perform sentiment analyses either generate their own data or use available datasets. Generating a new dataset makes it possible to use data that fits the problem the analysis is targeted at; moreover, the use of personal data ensures that no privacy laws are violated [67]. However, the main drawback is having to label the dataset, which is a challenging task. Moreover, it is not always easy to generate a large volume of data. Our approach to selecting datasets was based on their availability and accessibility. Respecting personal privacy was another factor that was considered, given that it appears in the regulations of most journals as a requirement for article publication.

Thus, we carefully chose datasets that are widely accepted by the research community.

In addition, one of our main concerns was the extensibility of the results obtained in the study. Therefore, the datasets were obtained from different sources and they cover different topics in order to perform a wide range of experiments. In this way, the results have made it possible to make a comprehensive comparison of the performance of deep learning models in sentiment analysis. We also considered the size of the datasets; the larger they are, the more possibilities they offer, even though this also increases their complexity. We worked with labeled datasets from which personal information was removed, since this information was not needed to test the performance of sentiment analysis models. These datasets are described below:

- Sentiment140 was obtained from Stanford University [68]. It contains 1.6 million tweets about products or brands. The tweets were already labeled with the polarity of the sentiment conveyed by the person writing them (0 = negative, 4 = positive).
- Tweets Airline [69] is a tweet dataset containing user opinions about U.S. airlines. It was crawled in February 2015. It has 14,640 samples, and it was divided into negative, neutral, and positive classes.
- Tweets SemEval [70] is a tweet dataset that includes a range of named geopolitical entities. This dataset has 17,750 samples, and it was divided into positive, neutral, and negative classes.
- IMDB Movie Reviews [71] is a dataset of comments from audiences about the stories in films. It has 25,000 samples divided into positive and negative.
- IMDB Movie Reviews was obtained from Stanford University [72]. This dataset contains comments from audiences about the story of films. It has 50,000 samples, which are divided into positive and negative.
- Cornell Movie Reviews [73] contains comments from audiences about the stories in films. This dataset includes 10,662 samples for training and testing, which are labeled negative or positive.
- Book Reviews and Music Reviews is a dataset obtained from the Multidomain Sentiment of the Department of Computer Science of Johns Hopkins University. Biographies, Bollywood, Boom Boxes, and Blenders: Domain Adaptation for Sentiment Classification [74] contains user comments about books and music. Each has 2,000 samples with two classes—negative and positive.

Figure 6 shows an original sample of tweets in one of the datasets. It contains information on each of the following fields:

- "target" is the polarity of the tweet;
- "id" is the unique ID of each tweet;
- "date" is the date of the tweet;
- "query_string" indicates whether the tweet has been collected with any particular query keyword (for this column, 100% of the entries labeled are with the value "NO_QUERY");
- "user" is the Twitter handle name of the user who tweeted;
- "text" is the verbatim text of the tweet.

We used the "text" and "target" fields to perform the experiment.

	A	B	C	D	E	F
1	target	id	date	flag	user	text
2	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by texting it
3	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Managed t
4	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
5	4	1882658073	Fri May 22 07:15:43 PDT 2009	NO_QUERY	DanBearUK	That's Flicking her bean and living the dream
6	4	1882658126	Fri May 22 07:15:43 PDT 2009	NO_QUERY	djsteveranford	@thedjbook Brilliant idea!!! Just subscribed to the ma
7	4	1882658150	Fri May 22 07:15:43 PDT 2009	NO_QUERY	jeffjulian	#followfriday fave: @Chicagoist. I'll also plug my comn
8	0	1467811594	Mon Apr 06 22:20:03 PDT 2009	NO_QUERY	coZZ	@LOLTrish hey long time no see! Yes.. Rains a bit ,only
9	4	1692641348	Sun May 03 20:26:45 PDT 2009	NO_QUERY	howl_at_themoon	I'm pretty dang sure that i had a very good day today.
10	4	1692641461	Sun May 03 20:26:46 PDT 2009	NO_QUERY	chellz89	@MiSSJeSS06 yea do that & send me 1
11	0	1467812416	Mon Apr 06 22:20:16 PDT 2009	NO_QUERY	erinx3leannexo	spring break in plain city... it's snowing
12	0	1467812579	Mon Apr 06 22:20:17 PDT 2009	NO_QUERY	pardonlauren	I just re-pierced my ears

Figure 6. Examples of the Sentiment140 dataset.

4.2. Methodological Approach

After reviewing the proposed sentiment analysis methods in Section 3, we identified three popular approaches that have been used frequently in recent studies, namely DNN, CNN, and RNN. These models have been employed in the majority of the 32 reviewed papers, have been widely tested, and provide highly accurate results when working with different types of datasets [75]. However, no comparative study involving those algorithms has been conducted.

The focus of this research was the deep learning approach; therefore, we performed a comparative study of the performance of the three most popular deep learning models (DNN, CNN, and RNN) on eight datasets. Moreover, two text processing techniques (word embedding and TF-IDF) were employed in data preprocessing. The objective of the experiments is to compare the performance of these techniques, contributing in this way to the state-of-the-art literature on sentiment analysis tasks. These algorithms were applied to predict the sentiment polarity of the text and classify it according to that polarity. The performance of those methods was evaluated by means of the most suitable metrics used for classification problems: overall accuracy, recall, F-score, and Area Under Curve (AUC). We used k-fold cross validation with k = 10 in the application of the metrics. More details about the application of DNN, CNN, and RNN algorithms with word embedding and TF-IDF are given in Section 2.

4.3. Sentiment Classification

The process of sentiment analysis is discussed below. Data cleaning and feature extraction were performed in the preprocessing stages. In the training stage, several deep learning models were used. Detailed results are presented in the next section.

The main objective of our study is to evaluate the deep learning models. We used k-fold cross validation with k = 10 to determine the performance of the algorithms. All of them were tested with word embedding and TF-IDF.

Text cleaning is a preprocessing step that removes words or other components that do not contain relevant information, and thus may reduce the effectiveness of sentiment analysis. Text or sentence data include white space, punctuation, and stop words. Text cleaning has several steps for sentence normalization. All datasets were cleaned using the following steps:

- Cleaning the Twitter RTs, @, #, and the links from the sentences;
- Stemming or lemmatization;
- Converting the text to lower case;
- Cleaning all the non-letter characters, including numbers;
- Removing English stop words and punctuation;
- Eliminating extra white spaces;
- Decoding HTML to general text.

A certain processing method was then performed depending on the dataset to facilitate model formation. For example, for the Sentiment140 dataset, we dropped the columns that are not useful

for sentiment analysis purposes: {"id", "date", "query_string", "user"} and converted class label values {4, 0} to {1, 0} (1 = positive, 0 = negative). For the Tweets Airline and Tweets SemEval datasets, we removed all samples labeled "neutral", leaving only two classes for the experiment—positive and negative.

After the datasets were cleaned, sentences were split into individual words, which were returned to their base form by lemmatization. At this point, sentences were converted into vectors of continuous real numbers (also known as feature vectors) by using two methods: word embedding and TF-IDF. Both kinds of feature vectors were the inputs for the deep learning algorithms evaluated in the study. Those algorithms were CNN, DNN, and RNN. Thus, two models were induced per algorithm, one for each type of vector.

4.4. Sentiment Model

Most traditional models use well-known features, such as bag-of words, n-grams, and TF-IDF. Such features do not consider the semantic similarity between words. Currently, many deep learning models in NLP require word embedding results as input features. Figure 7 shows the semantic similarity of the words that are the closest to "iPhone", "Obama", and "university". The words nearest to "Obama" are "president", "leader", and "election". The words nearest to "university" are "students", "education", and "master". Since neural networks can be deployed to solve sentiment classification using word embedding, we use Word2vec to train initial word vectors from the datasets that were described above.

Figure 8 shows word clouds produced from some of the topics of the datasets described in Section 4.1. These datasets were cleaned before being transformed into vectors. The figure demonstrates how topics can be easily identified. The book topic is shown in the top left corner, the movie topic is shown in the top right corner, the left bottom corner shows the music topic, and finally the airplane topic is shown in the bottom right corner.

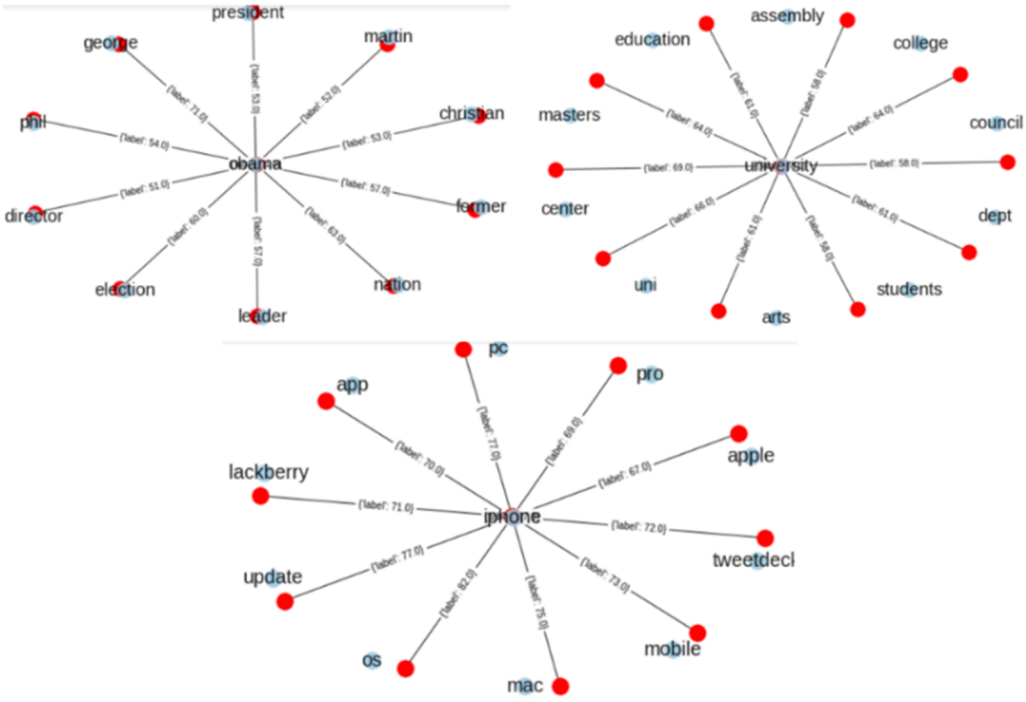


Figure 7. Word embedding after training the dataset with a minimum count of 5000.

Figure 8. Word cloud view of the topics of the datasets.

As stated before, we used k-fold cross validation to determine the effectiveness of different embedding with k = 10. The details are shown in the experimental results section. Figure 9 shows the details of the CNN model, which are explained below.

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 40, 300)	4500300
conv1d_1 (Conv1D)	(None, 40, 64)	57664
conv1d_2 (Conv1D)	(None, 40, 32)	6176
max_pooling1d_1 (MaxPooling1	(None, 13, 32)	0
conv1d_3 (Conv1D)	(None, 13, 16)	1552
conv1d_4 (Conv1D)	(None, 13, 8)	264
global_average_pooling1d_1 (	(None, 8)	0
dense_1 (Dense)	(None, 1)	9

Total params: 4,565,965
Trainable params: 65,665
Non-trainable params: 4,500,300

Figure 9. The details of the CNN model, which was set up for the experiment.

The function embedding is the embedding layer that is initialized with random weights and which will learn the embedding for all words in the training datasets. In our case, the size of the vocabulary is 15,000, the output dim is 300, and the maximum length is 40. The results are in a 40 × 300 matrix.

The first 1D CNN layer defines a filter of kernel size 3. For this, we will define 64 filters. This allows us to train 64 different features on the first layer of the network. Thus, the output of the first neural network layer is a 40 × 64 neuron matrix, and the result from the first CNN will be fed into the second

CNN layer. We will again define 32 different filters to be trained on this level. Following the same logic as the first layer, the output matrix will measure 40 × 32.

The maximum pooling layer is often used after a CNN layer in order to reduce the complexity of the output and prevent overfitting of the data. In our case, we choose a size of three. This means that the size of the output matrix of this layer is 13 × 32.

The third and fourth 1D CNN layers are in charge of learning higher level features. The outputs of those two layers are a 13 × 16 matrix and a 13 × 8 matrix.

The average pooling layer is a pooling layer used to further avoid overfitting. We will use the average value instead of the maximum value because it will give better results in this case. The output matrix has a size of 1 × 8 neurons.

The fully connected layer with sigmoid activation is the final layer that will reduce the vector of height 8 to 1 for prediction ("positive", "negative").

5. Experimental Results

To conduct the tests, we used a GeForce GTX2070 GPU card, and the Keras (https://keras.io) and Tensorflow (https://www.tensorflow.org/) libraries. DNN, CNN, and RNN models were applied to perform experiments with the different datasets described above, in order to analyze the performance of those algorithms using both word embedding and TF-IDF feature extraction.

In all the experiments, we configure the parameter for our code, such as echoes = 5, batch size = 4096, and k-fold = 10.

Accuracy, AUC, and F-score were the metrics used to evaluate the performance of the models through all experiments. Since F-score is derived from recall and precision, we also show these two measures for reference purposes.

Sentiment140 was the first dataset to be processed. Its contents were labeled as positive or negative. Since this dataset contains a much larger number of tweets than the other datasets, we first analyzed the performance of the models induced from different subsets formed with different percentages of the initial data, ranging from 10% to 100%. As shown in Figures 10–15, the combinations of feature extractors and deep learning techniques applied to those subsets produced different results for Sentiment140 data.

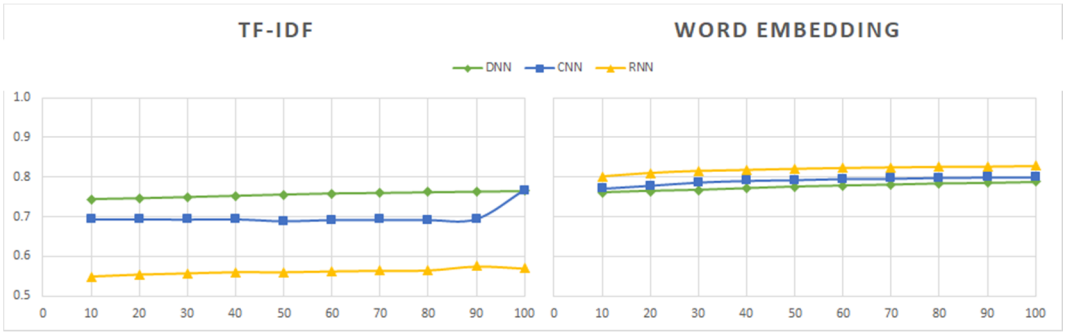


Figure 10. Accuracy values of deep-learning models with TF-IDF and word embedding for different numbers of tweets (percentage of the dataset).

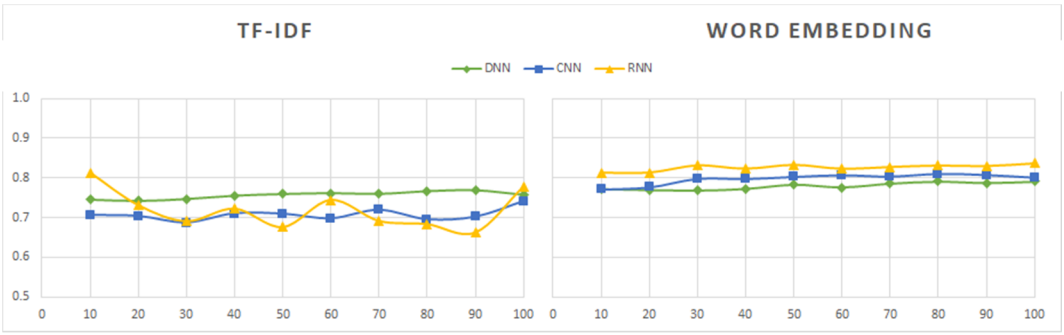


Figure 11. Recall values of DNN, CNN, and RNN models with TF-IDF and word embedding for different numbers of tweets (percentage of the dataset).

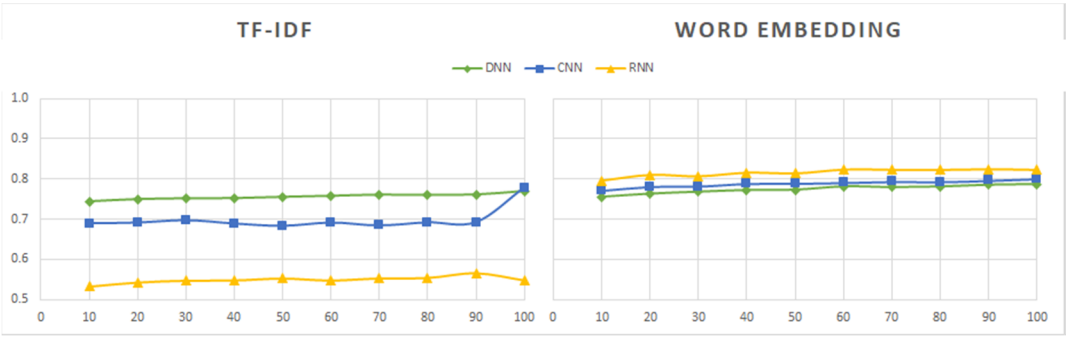


Figure 12. Precision values of DNN, CNN, and RNN models with TF-IDF and word embedding for different numbers of tweets (percentage of the dataset).

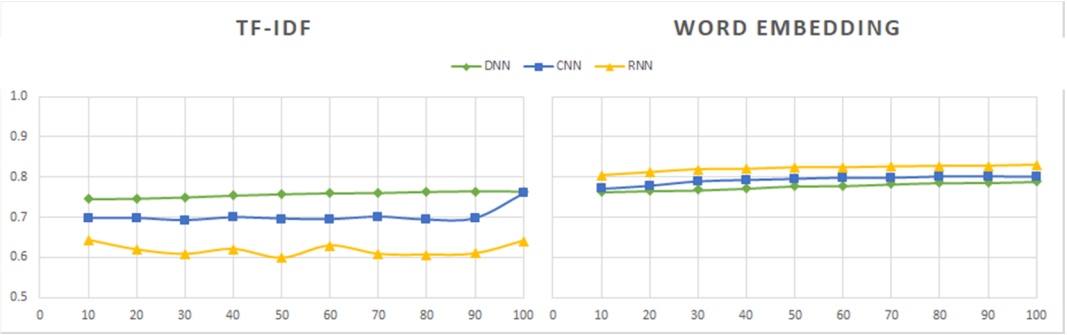


Figure 13. F-score values of DNN, CNN, and RNN models with TF-IDF and word embedding for different numbers of tweets (percentage of the dataset).

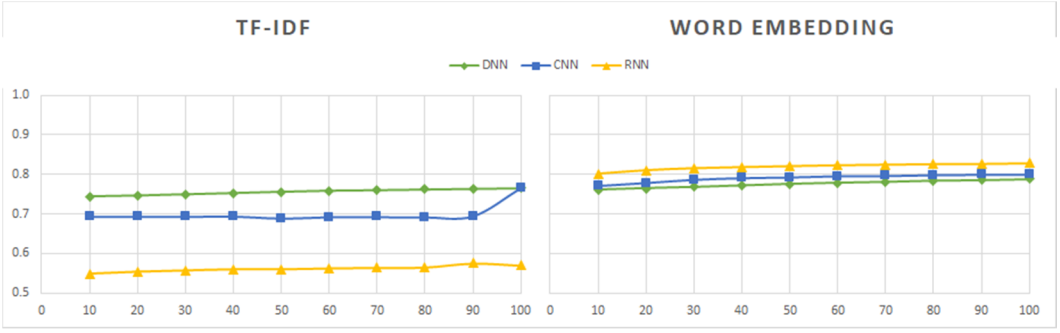


Figure 14. AUC values of DNN, CNN, and RNN models with TF-IDF and word embedding for different numbers of tweets (percentage of the dataset).

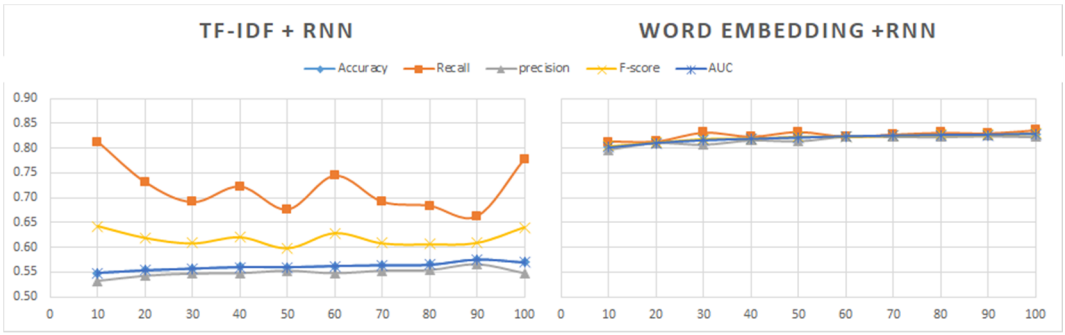


Figure 15. Comparison of all measures for RNN with TF-IDF (left) and word embedding (right).

An observation that is clearly seen in Figures 10–15 is the best behavior of the models when using word embedding against TF-IDF regarding all metrics analyzed. This improvement is especially significant for RNN, which is the method providing the best results when used in conjunction with word embedding. On the contrary, RNN is the worst of the three analyzed methods when used with TF-IDF. Figure 14 shows the metric values yielded by the RNN models.

We can also see in the graphs above that there are no significant differences in the values of the evaluation measures between the three deep learning techniques in the case of word embedding, while in the case of TF-IDF the differences in the results of the three methods are important.

Regarding the dataset size, its influence on the results is minimal for word embedding, being slightly greater and uneven for the TF-IDF method.

Therefore, from the analysis of the results obtained with the Sentiment140 dataset, we can deduce that word embedding is a more robust technique than TF-IDF. In addition, its use would allow us to work with a subset of data containing 50% or 60% of the total sample at a lower computational cost and with hardly any difference in the results.

After this preliminary study, we conducted additional experiments, in which the methods were tested on other datasets. Some of them contain tweets, while others contain different types of reviews, as noted in Section 4.1. These datasets contain significantly fewer examples than Sentiment140's, which has 1.6 million entries, so there was no problem working with the complete datasets.

Tables 2–6 show the results of the datasets; Figures 16–20 illustrate these results. These tables include the results of the complete Sentiment140 dataset, although as we found in the preliminary study, we could have used a subset of it and obtained similar results.

Table 2. Accuracy comparison for datasets with two classes (positive and negative).

Datasets	TF-IDF			Word Embedding		
	DNN	CNN	RNN	DNN	CNN	RNN
Sentiment140	0.76497407	0.76688544	0.56957939	0.78816761	0.80060849	0.82819948
Tweets Airline	0.85936944	0.85451457	0.82809226	0.8979309	0.90373439	0.90451624
Tweets SemEval	0.83674669	0.81377485	0.54857318	0.83674748	0.84313431	0.85172402
IMDB Movie Reviews (1)	0.85232000	0.82300000	0.56392000	0.84572000	0.86072000	0.87052000
IMDB Movie Reviews (2)	0.85512000	0.80628002	0.58724000	0.80252000	0.82624000	0.86688000
Cornell Movie Reviews	0.70437264	0.67867751	0.50787764	0.70221434	0.71365671	0.76693790
Book Reviews	0.75876443	0.72741509	0.5169437	0.74560455	0.76630924	0.73347052
Music Reviews	0.76850000	0.69200000	0.5170000	0.70800000	0.74450000	0.73100000

Table 3. The recall comparison for different datasets.

Datasets	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
Sentiment140	0.75775700	0.74076035	0.77731305	0.79096262	0.80080020	0.83692316
Tweets Airline	0.95565582	0.97003680	0.97417837	0.9577253	**0.95924821**	**0.95086398**
Tweets SemEval	0.80817204	0.7744086	0.09462366	0.80860215	0.81827957	0.83139785
IMDB Movie Reviews (1)	0.84072000	0.80080000	0.46880000	0.84360000	0.84960000	0.86808000
IMDB Movie Reviews (2)	0.87112000	0.75744000	0.56088000	0.78304000	0.83248000	0.88832000
Cornell Movie Reviews	0.71468474	0.67811554	0.84203575	0.70455552	0.72050860	0.80943813
Book Reviews	0.74221810	0.73009689	0.63040610	0.73912595	0.81599670	0.74824778
Music Reviews	0.76500000	0.69700000	0.74200000	0.68600000	0.72900000	0.73600000

Table 4. The precision comparison for different datasets.

Datasets	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
Sentiment140	0.75775700	0.74076035	0.77731305	0.79096262	0.80080020	0.83692316
Tweets Airline	0.88451273	0.86396543	0.83664149	0.91759076	**0.92284682**	**0.93061436**
Tweets SemEval	0.83504669	0.81594219	0.58839133	0.83492767	0.84024502	0.84745555
IMDB Movie Reviews (1)	0.85057402	0.83996428	0.61862397	0.84727512	0.8689903	0.87328478
IMDB Movie Reviews (2)	0.84410853	0.83943612	0.59209526	0.81478398	0.82221871	0.85179503
Cornell Movie Reviews	0.70070694	0.67920909	0.45431496	0.70142346	0.71117779	0.74808808
Book Reviews	0.77071809	0.72645030	0.56145983	0.74877856	0.74335207	0.73283058
Music Reviews	0.77097163	0.69126657	0.46068591	0.71900797	0.75328872	0.73186536

Table 5. The F-score comparison for different datasets.

Datasets	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
Sentiment140	0.76383225	0.75932297	0.64044056	0.78876610	0.80063705	0.82967613
Tweets Airline	0.91863362	0.91385701	0.90011208	0.93720980	**0.94064543**	**0.94059646**
Tweets SemEval	0.82114704	0.79433397	0.13751971	0.82130776	0.82884635	0.83874720
IMDB Movie Reviews (1)	0.85057402	0.81871110	0.46834558	0.84540045	0.85908973	0.87020187
IMDB Movie Reviews (2)	0.85740157	0.79633290	0.57606508	0.79859666	0.82731754	0.86967419
Cornell Movie Reviews	0.70731859	0.67852670	0.59007189	0.70290291	0.71560412	0.77594109
Book Reviews	0.75501388	0.72758940	0.51163296	0.74364502	0.77728796	0.73395298
Music Reviews	0.76770393	0.69126657	0.56736672	0.70080624	0.74026385	0.73207829

Table 6. The AUC comparison for different datasets.

Datasets	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
Sentiment140	0.76499683	0.76535951	0.56950939	0.78816189	0.80062146	0.82818031
Tweets Airline	0.73510103	0.68790047	0.61740993	0.81170789	**0.82367939**	**0.83767632**
Tweets SemEval	0.83484059	0.81115021	0.51834041	0.83487221	0.84147827	0.85037175
IMDB Movie Reviews (1)	0.85232000	0.82300000	0.56392000	0.84572000	0.86072000	0.87052000
IMDB Movie Reviews (2)	0.85512000	0.80628000	0.58724000	0.80252000	0.82624000	0.86688000
Cornell Movie Reviews	0.70437264	0.67867751	0.50787764	0.70221434	0.71365671	0.76693790
Book Reviews	0.75875593	0.72740157	0.51676458	0.74558854	0.76630592	0.73348794
Music Reviews	0.76850000	0.69200000	0.51700000	0.70800000	0.74450000	0.73207829

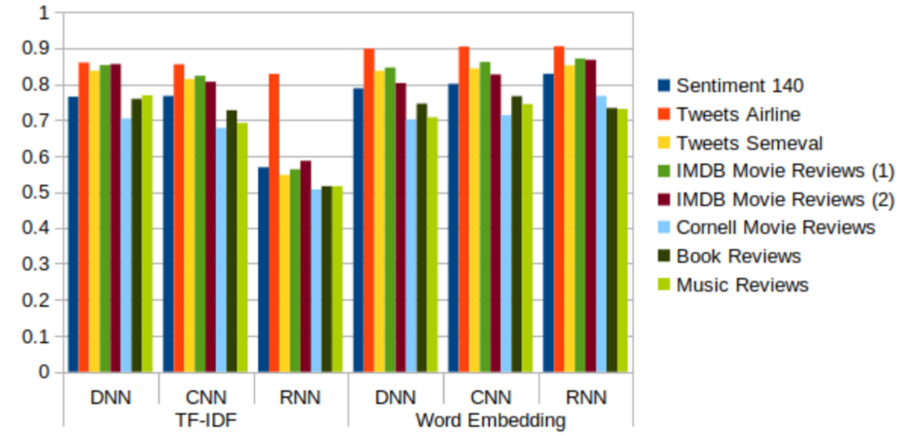


Figure 16. Accuracy values of deep-learning models with TF-IDF and word embedding for different datasets.

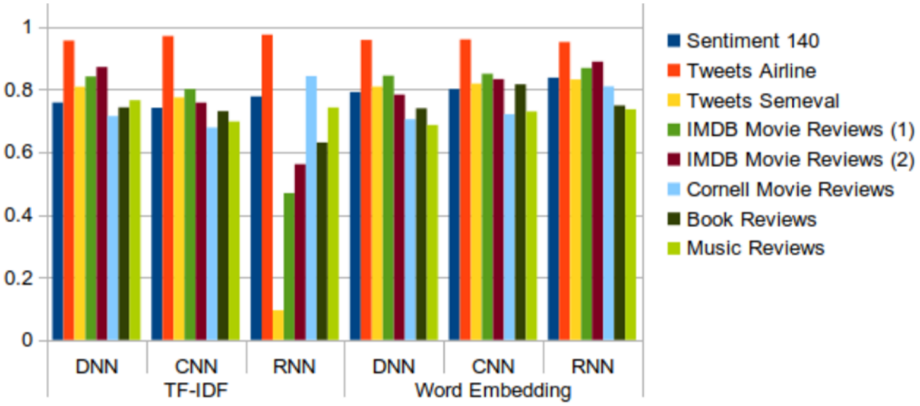


Figure 17. Recall values of DNN, CNN, and RNN models with TF-IDF and word embedding for different datasets.

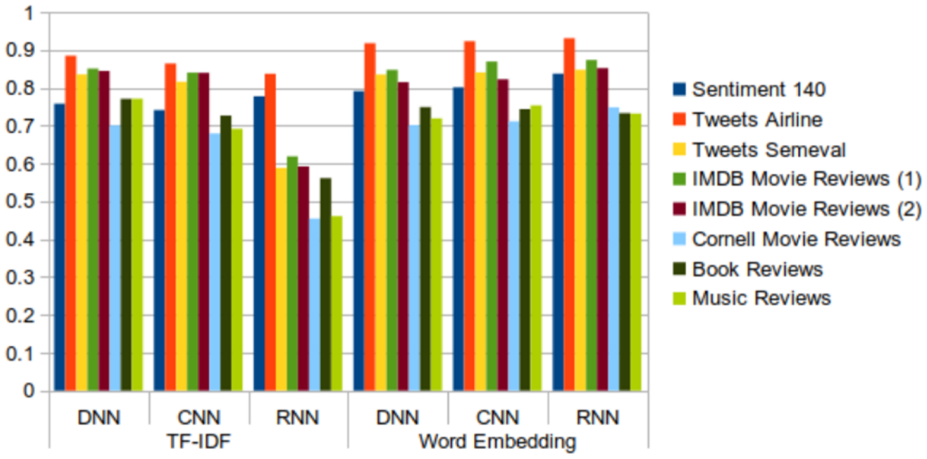


Figure 18. Precision values of DNN, CNN, and RNN models with TF-IDF and word embedding for different datasets.

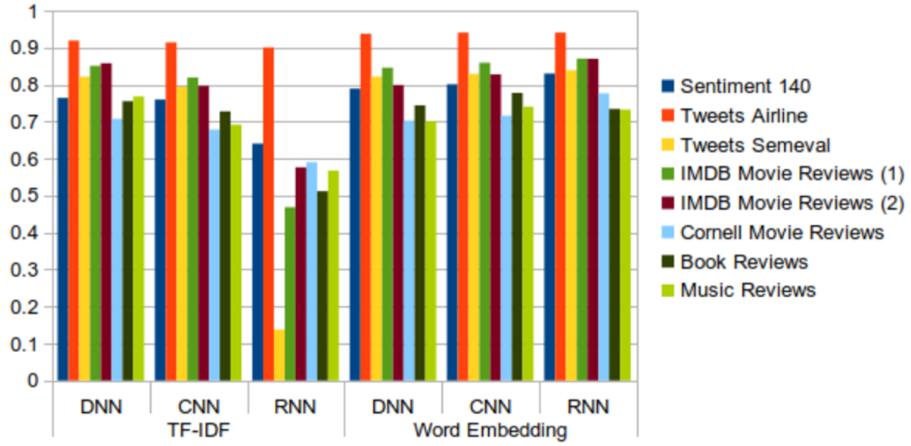


Figure 19. F-score values of DNN, CNN, and RNN models with TF-IDF and word embedding for different datasets.

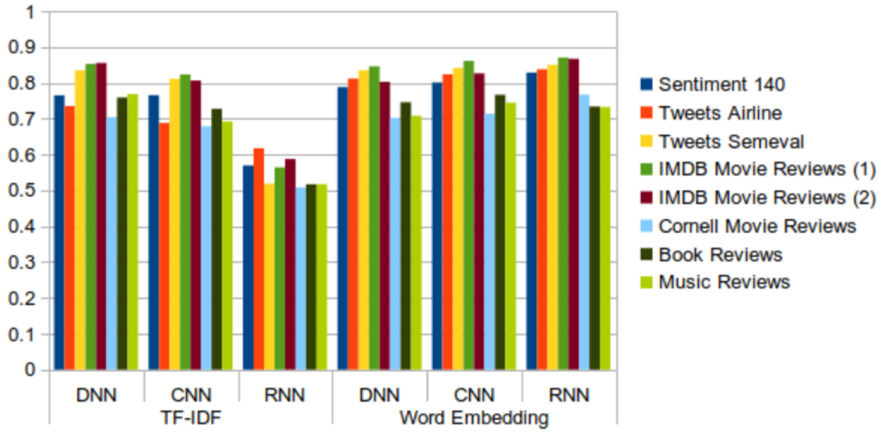


Figure 20. AUC values of DNN, CNN, and RNN models with TF-IDF and word embedding for different datasets.

The results obtained with the new datasets do nothing more than confirm the conclusions obtained after the analysis of the results of the Sentiment140 dataset. In general, the best behavior is shown by the combination of RNN and word embedding, although there are some exceptions. These are produced in the "book reviews" and "music review" datasets, where the values of all the metrics are slightly higher for DNN + TF-IDF than for RNN + word embedding. For "book reviews", the highest values for accuracy, recall, F-score, and AUC were given by CNN + word embedding. In addition, the Tweets Airline dataset is one of the datasets that shows the highest values for all metrics in all cases. We can also highlight that the recall metric shows an uneven behavior, especially for the model that combines RNN and TF-IDF. The same behavior was seen in the preliminary study with the Sentiment140 dataset (Figure 11). Likewise, as in the preliminary study, we can affirm that word embedding is a more appropriate technique than TF-IDF for performing sentiment analysis, despite the slight improvements obtained with TF-IDF for some data sets.

After analyzing the results concerning the quality of the predictions, it is necessary to obtain information on the computational cost associated with the induction of the models, since the differences between the results or some of them are not very significant. The aim is to know the extent to which the best reliability values are obtained at the expense of a higher or lower computational cost.

The CPU times are shown in Tables 7 and 8. Table 7 shows the processing time required to induce the models from the Sentiment140 dataset and its subsets. Table 8 contains the CPU time required for all datasets involved in the experiments.

Table 7. CPU times of experiments with numbers of tweets with a GPU.

Dataset (%)	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
10	1 min 37 s	1 min 14 s	11 min 18 s	25.8 s	39.6 s	4 min 58 s
20	2 min 32 s	2 min 25 s	22 min 14 s	41.3 s	1 min 18 s	11 min 59 s
30	3 min 26 s	3 min 34 s	32 min 56 s	1 min	1 min 53 s	18 min 57 s
40	4 min 19 s	4 min 53 s	44 min 1 s	1 min 21 s	2 min 32 s	25 min 9 s
50	**5 min 12 s**	**6 min 9 s**	**54 min 32 s**	**1 min 44 s**	**3 min 10 s**	**31 min 29 s**
60	6 min 33 s	7 min 23 s	1h 5 min 26 s	2 min 10 s	3 min 52 s	37 min 35 s
70	7 min 47 s	10 min 20 s	1 h15 min5 s	2 min 45 s	4 min 38 s	44 min 16 s
80	9 min 4 s	18 min 32 s	1 h27 min22 s	3 min 19 s	5 min 31 s	50 min 47 s
90	10 min 14 s	29 min 49 s	1 h37 min59 s	3 min 47 s	6 min 12 s	57 min 3 s
100	**11 min 55 s**	**38 min 17 s**	**1 h48 min52 s**	**4 min 18 s**	**7 min 3 s**	**1 h 4 min 16 s**

Table 8. CPU times of experiments with numbers of datasets with a GPU.

Dataset	TF-IDF			Word Embedding		
	DNN	**CNN**	**RNN**	**DNN**	**CNN**	**RNN**
Sentiment140	11 min 55 s	3 8min 17 s	**1h48 min 52 s**	4 min 18 s	7 min 3 s	**1 h 4 min 16 s**
Tweets Airline	1 min	34.41 s	1 h 54 s	30.66 s	1 min 22 s	2 min 41 s
Tweets SemEval	20.53 s	24.5 s	23 min 52 s	26.75 s	1 min 11 s	2 min 43 s
IMDB Movie Reviews (1)	1 min 11 s	1 min 7 s	**1 h25 min 48 s**	21.13 s	32.66 s	7 min 42 s
IMDB Movie Reviews (2)	17.78 s	22.05 s	30 min 21 s	31.32 s	36.81 s	8 min 23 s
Cornell Movie Reviews	23.2 s	16.83 s	31 min 55 s	12.9 s	21.26 s	4 min 40 s
Book Reviews	11.93 s	10.12 s	21 min 9 s	16.21 s	20.6 s	2 min17 s
Music Reviews	26.48 s	17.35 s	29 min 50 s	13.94 s	16.89 s	4 min 42 s

The tables show that the use of TF-IDF, which produces less reliable models, requires longer computational time than the use of word embedding. This is one more reason to consider this last technique as the most recommendable. However, RNN is the most time-consuming algorithm, both with TF-IDF and with word embedding. Given that the improvements of RNN with respect to DNN and CNN are not very significant in the latter case, the use of these two methods could be considered more appropriate when the computational cost needs to be reduced.

Regarding the large Sentiment140 dataset shown in Table 8, if the sample size is reduced by 50%, the evaluation measures are not significantly affected, but the processing time is reduced by 50%.

Moving to a comparison between the DNN and CNN models, CNN has slightly longer processing times, but it also has much better evaluation measures than DNN.

From the analysis of overall accuracy, recall, precision, F-scores, AUC values, and CPU times, we highlight some patterns for high and low performance of the sentiment analysis methods. We are aware that different types of datasets influence the results of a sentiment analysis differently [76].

- The DNN model is simple to implement and provides results within a short period of time—around 1 min for the majority of datasets, except dataset Sentiment140, for which the model took 12 min to obtain the results. Although the model is quick to train, the overall accuracy of the model is average (around 75% to 80%) in all of the tested datasets, including tweets and reviews.
- The CNN model is also fast to train and test, although possibly a bit slower than DNN. The model offers higher accuracy (over 80%) on both tweet and review datasets.
- The RNN model has the highest reliability when word embedding is applied, however its computational time is also the highest. When using RNN with TF-IDF, it takes a longer time than other models and results in lower accuracy (around 50%) in the sentiment analysis of tweet and review datasets.

In comparative studies presented in [5,17,19,20], which were performed by using tweets or reviews datasets, the evaluation of results was made only in terms of accuracy, however the processing time

was not considered. Regarding the continuously expanding size and complexity of big data in the future, it is crucial to consider both reliability and time, especially in critical systems requiring a fast response [77]. In this work, two techniques (TF-IDF and word embedding) are examined on three deep learning algorithms, which give an extended overview of performances of sentiment analysis using deep learning techniques.

Finally, general summaries of the results archived in the experiments referenced earlier are explained below:

- Three deep learning models (DNN, CNN, and RNN) were used to perform sentiment analysis experiments. The CNN model was found to offer the best tradeoff between the processing time and the accuracy of results. Although the RNN model had the highest degree of accuracy when used with word embedding, its processing time was 10 times longer than that of the CNN model. The RNN model is not effective when used with the TF-IDF technique, and its far higher processing time leads to results that are not significantly better. DNN is a simple deep learning model that has average processing times and yields average results. Future research on deep learning models can focus on ways of improving the tradeoff between the accuracy of results and the processing times.
- Related techniques (TF-IDF and word embedding) are used to transfer text data (tweets, reviews) into a numeric vector before feeding them into a deep learning model. The results when TF-IDF is used are poorer than when word embedding is used. Moreover, the TF-IDF technique used with the RNN model takes has a longer processing time and yields less reliable results. However, when RNN is used with word embedding, the results are much better. Future work can explore how to improve these and other techniques to achieve even better results.
- The results from the datasets containing tweets and IMDB movie review datasets are better than the results from the other datasets containing reviews. Regarding tweets data, the models induced from the Tweets Airline dataset, focused on a specific topic, show better performance than those built from datasets about generic topics.

6. Conclusions

In this paper, we described the core of deep learning models and related techniques that have been applied to sentiment analysis for social network data. We used word embedding and TF-IDF to transform input data before feeding that data into deep learning models. The architectures of DNN, CNN, and RNN were analyzed and combined with word embedding and TF-IDF to perform sentiment analysis. We conducted some experiments to evaluate DNN, CNN, and RNN models on datasets of different topics, including tweets and reviews. We also discussed related research in the field. This information, combined with the results of our experiments, gives us a broad perspective on applying deep learning models for sentiment analysis, as well as combining these models with text preprocessing techniques.

After the analysis of 32 papers, DNN, CNN, and hybrid approaches were identified as the most widely used models for sentiment polarity analysis. Another conclusion extracted from the analysis was the fact that common techniques, such as CNN, RNN, and LSTM, are individually tested in these studies on different datasets, however there is a lack of a comparative analysis for them. In addition, the results presented in most papers are given in terms of reliability, without considering computational time.

The experiments conducted in this work were designed to help fill the gaps mentioned above. We studied the impacts of different types of datasets, feature extraction techniques, and deep learning models, with a special focus on the problem of sentiment polarity analysis. The results show that it is better to combine deep learning techniques with word embedding than with TF-IDF when performing a sentiment analysis. The experiments also revealed that CNN outperforms other models, presenting a good balance between accuracy and CPU runtime. RNN reliability is slightly higher than CNN reliability with most datasets but its computational time is much longer. One last conclusion derived from the study is the observation that the effectiveness of the algorithms depends largely on the

characteristics of the datasets, hence the convenience of testing deep learning methods with more datasets in order to cover a greater diversity of characteristics.

In future work, we will focus on exploring hybrid approaches, where multiple models and techniques are combined in order to enhance the sentiment classification accuracy achieved by the individual models or techniques, as well as to reduce the computational cost. The aim is to extend the comparative study to include both new methods and new types of data. Therefore, the reliability and processing time of hybrid models will be evaluated with several types of data, such as status, comments, and news on social media. We will also intend to address the problem of aspect sentiment analysis in order to gain deeper insight into user sentiments by associating them with specific features or topics. This has great relevance for many companies, since it allows them to obtain detailed feedback from users, and thus know which aspects of their products or services should be improved.

Author Contributions: Conceptualization, N.C.D. and M.N.M.-G.; methodology, M.N.M.-G.; software, N.C.D.; validation, M.N.M.-G., and F.D.l.P.; formal analysis, N.C.D., and M.N.M.-G.; investigation, N.C.D.; data curation, N.C.D.; writing—original draft preparation, N.C.D.; writing—review and editing, M.N.M.-G. and F.D.l.P.; visualization, N.C.D.; supervision, M.N.M.-G. and F.D.l.P.; project administration, F.D.l.P.; funding acquisition, F.D.l.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Spanish government and European (Fondo Europeo de Desarrollo Regional) FEDER funds, project InEDGEMobility: Movilidad inteligente y sostenible soportada por Sistemas Multi-agentes y Edge Computing (RTI2018-095390-B-C32).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Pouli, V.; Kafetzoglou, S.; Tsiropoulou, E.E.; Dimitriou, A.; Papavassiliou, S. Personalized multimedia content retrieval through relevance feedback techniques for enhanced user experience. In Proceedings of the 2015 13th International Conference on Telecommunications (ConTEL), Graz, Austria, 13–15 July 2015; pp. 1–8.
2. Thai, M.T.; Wu, W.; Xiong, H. *Big Data in Complex and Social Networks*; CRC Press: Boca Raton, FL, USA, 2016.
3. Cambria, E.; Das, D.; Bandyopadhyay, S.; Feraco, A. *A Practical Guide to Sentiment Analysis*; Springer: Berlin, Germany, 2017.
4. Hussein, D.M.E.-D.M. A survey on sentiment analysis challenges. *J. King Saud Univ. Eng. Sci.* **2018**, *30*, 330–338. [CrossRef]
5. Sohangir, S.; Wang, D.; Pomeranets, A.; Khoshgoftaar, T.M. Big Data: Deep Learning for financial sentiment analysis. *J. Big Data* **2018**, *5*, 3. [CrossRef]
6. Jangid, H.; Singhal, S.; Shah, R.R.; Zimmermann, R. Aspect-Based Financial Sentiment Analysis using Deep Learning. In Proceedings of the Companion of the The Web Conference 2018 on The Web Conference, Lyon, France, 23–27 April 2018; pp. 1961–1966.
7. Keenan, M.J.S. *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets*; Wiley: Hoboken, NJ, USA, 2018.
8. Satapathy, R.; Cambria, E.; Hussain, A. *Sentiment Analysis in the Bio-Medical Domain*; Springer: Berlin, Germany, 2017.
9. Rajput, A. Natural Language Processing, Sentiment Analysis, and Clinical Analytics. In *Innovation in Health Informatics*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 79–97.
10. Qian, J.; Niu, Z.; Shi, C. Sentiment Analysis Model on Weather Related Tweets with Deep Neural Network. In Proceedings of the 2018 10th International Conference on Machine Learning and Computing, Macau, China, 26–28 February 2018; pp. 31–35.
11. Pham, D.-H.; Le, A.-C. Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data Knowl. Eng.* **2018**, *114*, 26–39. [CrossRef]
12. Preethi, G.; Krishna, P.V.; Obaidat, M.S.; Saritha, V.; Yenduri, S. Application of deep learning to sentiment analysis for recommender system on cloud. In Proceedings of the 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), Dalian, China, 21–23 July 2017; pp. 93–97.

13. Ain, Q.T.; Ali, M.; Riaz, A.; Noureen, A.; Kamran, M.; Hayat, B.; Rehman, A. Sentiment analysis using deep learning techniques: A review. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 424.
14. Gao, Y.; Rong, W.; Shen, Y.; Xiong, Z. Convolutional neural network based sentiment analysis using Adaboost combination. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 1333–1338.
15. Hassan, A.; Mahmood, A. Deep learning approach for sentiment analysis of short texts. In Proceedings of the Third International Conference on Control, Automation and Robotics (ICCAR), Nagoya, Japan, 24–26 April 2017; pp. 705–710.
16. Kraus, M.; Feuerriegel, S. Sentiment analysis based on rhetorical structure theory: Learning deep neural networks from discourse trees. *Expert Syst. Appl.* **2019**, *118*, 65–79. [CrossRef]
17. Li, L.; Goh, T.-T.; Jin, D. How textual quality of online reviews affect classification performance: A case of deep learning sentiment analysis. *Neural Comput. Appl.* **2018**, 1–29. [CrossRef]
18. Singhal, P.; Bhattacharyya, P. *Sentiment Analysis and Deep Learning: A Survey*; Center for Indian Language Technology, Indian Institute of Technology: Bombay, Indian, 2016.
19. Alharbi, A.S.M.; de Doncker, E. Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information. *Cogn. Syst. Res.* **2019**, *54*, 50–61. [CrossRef]
20. Abid, F.; Alam, M.; Yasir, M.; Li, C.J. Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter. *Future Gener. Comput. Syst.* **2019**, *95*, 292–308. [CrossRef]
21. Aggarwal, C.C. *Neural Networks and Deep Learning*; Springer: Berlin, Germany, 2018.
22. Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. *WIREs Data Min. Knowl. Discov.* **2018**, *8*, e1253. [CrossRef]
23. Britz, D. Recurrent Neural Networks Tutorial, Part 1–Introduction to Rnns. Available online: http://www.wildml.com/2015/09/recurrent-neural-networkstutorial-part-1-introduction-to-rnns/ (accessed on 12 March 2020).
24. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long time lag problems. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 2–5 December 1996; pp. 473–479.
25. Ruangkanokmas, P.; Achalakul, T.; Akkarajitsakul, K. Deep Belief Networks with Feature Selection for Sentiment Classification. In Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, 25–27 January 2016; pp. 9–14.
26. Socher, R.; Lin, C.C.; Manning, C.; Ng, A.Y. Parsing natural scenes and natural language with recursive neural networks. In Proceedings of the 28th International Conference on Machine Learning (ICML-11), Bellevue, WA, USA, 28 June–2 July 2011; pp. 129–136.
27. Long, H.; Liao, B.; Xu, X.; Yang, J. A hybrid deep learning model for predicting protein hydroxylation sites. *Int. J. Mol. Sci.* **2018**, *19*, 2817. [CrossRef]
28. Vateekul, P.; Koomsubha, T. A study of sentiment analysis using deep learning techniques on Thai Twitter data. In Proceedings of the 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), Khon Kaen, Thailand, 13–15 July 2016; pp. 1–6.
29. Ghosh, R.; Ravi, K.; Ravi, V. A novel deep learning architecture for sentiment classification. In Proceedings of the 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, 3–5 March 2016; pp. 511–516.
30. Bhavitha, B.; Rodrigues, A.P.; Chiplunkar, N.N. Comparative study of machine learning techniques in sentimental analysis. In Proceedings of the 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 10–11 March 2017; pp. 216–221.
31. Salas-Zárate, M.P.; Medina-Moreira, J.; Lagos-Ortiz, K.; Luna-Aveiga, H.; Rodriguez-Garcia, M.A.; Valencia-García, R.J.C. Sentiment analysis on tweets about diabetes: An aspect-level approach. *Comput. Math. Methods Med.* **2017**, *2017*. [CrossRef] [PubMed]
32. Huq, M.R.; Ali, A.; Rahman, A. Sentiment analysis on Twitter data using KNN and SVM. *IJACSA Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 19–25.
33. Pinto, D.; McCallum, A.; Wei, X.; Croft, W.B. Table extraction using conditional random fields. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, Toronto, ON, Canada, 28 July–1 August 2003; pp. 235–242.

34. Soni, S.; Sharaff, A. Sentiment analysis of customer reviews based on hidden markov model. In Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015), Unnao, India, 6 March 2015; pp. 1–5.
35. Zhang, X.; Zheng, X. Comparison of Text Sentiment Analysis Based on Machine Learning. In Proceedings of the 2016 15th International Symposium on Parallel and Distributed Computing (ISPDC), Fuzhou, China, 8–10 July 2016; pp. 230–233.
36. Malik, V.; Kumar, A. Communication. Sentiment Analysis of Twitter Data Using Naive Bayes Algorithm. *Int. J. Recent Innov. Trends Comput. Commun.* **2018**, *6*, 120–125.
37. Mehra, N.; Khandelwal, S.; Patel, P. *Sentiment Identification Using Maximum Entropy Analysis of Movie Reviews*; Stanford University: Stanford, CA, USA, 2002.
38. Wu, H.; Li, J.; Xie, J. Maximum entropy-based sentiment analysis of online product reviews in Chinese. In *Automotive, Mechanical and Electrical Engineering*; CRC Press: Boca Raton, FL, USA, 2017; pp. 559–562.
39. Firmino Alves, A.L.; Baptista, C.d.S.; Firmino, A.A.; Oliveira, M.G.d.; Paiva, A.C.D. A Comparison of SVM versus naive-bayes techniques for sentiment analysis in tweets: A case study with the 2013 FIFA confederations cup. In Proceedings of the 20th Brazilian Symposium on Multimedia and the Web, João Pessoa, Brazil, 18–21 November 2014; pp. 123–130.
40. Pandey, A.C.; Rajpoot, D.S.; Saraswat, M. Twitter sentiment analysis using hybrid cuckoo search method. *Inf. Process. Manag.* **2017**, *53*, 764–779. [CrossRef]
41. Medhat, W.; Hassan, A.; Korashy, H. Sentiment analysis algorithms and applications: A survey. *Ain Shams Eng. J.* **2014**, *5*, 1093–1113. [CrossRef]
42. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the Advances in neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
43. Jain, A.P.; Dandannavar, P. Application of machine learning techniques to sentiment analysis. In Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Karnataka, India, 21–23 July 2016; pp. 628–632.
44. Tang, D.; Qin, B.; Liu, T. Deep learning for sentiment analysis: Successful approaches and future challenges. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2015**, *5*, 292–303. [CrossRef]
45. Sharef, N.M.; Zin, H.M.; Nadali, S. Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data. *JCS* **2016**, *12*, 153–168. [CrossRef]
46. Rojas-Barahona, L.M. Deep learning for sentiment analysis. *Lang. Linguist. Compass* **2016**, *10*, 701–719. [CrossRef]
47. Roshanfekr, B.; Khadivi, S.; Rahmati, M. Sentiment analysis using deep learning on Persian texts. In Proceedings of the 2017 Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, 2–4 May 2017; pp. 1503–1508.
48. Jeong, B.; Yoon, J.; Lee, J.-M. Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis. *Int. J. Inf. Manag.* **2019**, *48*, 280–290. [CrossRef]
49. Gupta, U.; Chatterjee, A.; Srikanth, R.; Agrawal, P. A sentiment-and-semantics-based approach for emotion detection in textual conversations. *arXiv* **2017**, arXiv:1707.06996.
50. Ramadhani, A.M.; Goo, H.S. Twitter sentiment analysis using deep learning methods. In Proceedings of the 2017 7th International Annual Engineering Seminar (InAES), Yogyakarta, Indonesia, 1–2 August 2017; pp. 1–4.
51. Paredes-Valverde, M.A.; Colomo-Palacios, R.; Salas-Zárate, M.D.P.; Valencia-García, R. Sentiment analysis in Spanish for improvement of products and services: A deep learning approach. *Sci. Program.* **2017**, *2017*. [CrossRef]
52. Yang, C.; Zhang, H.; Jiang, B.; Li, K.J. Aspect-based sentiment analysis with alternating coattention networks. *Inf. Process. Manag.* **2019**, *56*, 463–478. [CrossRef]
53. Do, H.H.; Prasad, P.; Maag, A.; Alsadoon, A.J. Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review. *Expert Syst. Appl.* **2019**, *118*, 272–299. [CrossRef]
54. Schmitt, M.; Steinheber, S.; Schreiber, K.; Roth, B. Joint Aspect and Polarity Classification for Aspect-based Sentiment Analysis with End-to-End Neural Networks. *arXiv* **2018**, arXiv:1808.09238.
55. Balabanovic, M.; Shoham, Y. Combining content-based and collaborative recommendation. *Commun. ACM* **1997**, *40*, 66–72. [CrossRef]

56. Wang, Y.; Wang, M.; Xu, W. A sentiment-enhanced hybrid recommender system for movie recommendation: A big data analytics framework. *Wirel. Commun. Mob. Comput.* **2018**, *2018*. [CrossRef]
57. Singh, V.K.; Mukherjee, M.; Mehta, G.K. Combining collaborative filtering and sentiment classification for improved movie recommendations. In Proceedings of the International Workshop on Multi-disciplinary Trends in Artificial Intelligence, Hyderabad, India, 7–9 December 2011; pp. 38–50.
58. Chen, Z.; Liu, B. Lifelong machine learning. *Synth. Lect. Artif. Intell. Mach. Learn.* **2018**, *12*, 1–207. [CrossRef]
59. Stai, E.; Kafetzoglou, S.; Tsiropoulou, E.E.; Papavassiliou, S.J. A holistic approach for personalization, relevance feedback & recommendation in enriched multimedia content. *Multimed. Tools Appl.* **2018**, *77*, 283–326.
60. Wu, C.; Wu, F.; Wu, S.; Yuan, Z.; Liu, J.; Huang, Y. Semi-supervised dimensional sentiment analysis with variational autoencoder. *Knowl. Based Syst.* **2019**, *165*, 30–39. [CrossRef]
61. Zhang, Z.; Zou, Y.; Gan, C. Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression. *Neurocomputing* **2018**, *275*, 1407–1415. [CrossRef]
62. Tang, D.; Zhang, M. Deep Learning in Sentiment Analysis. In *Deep Learning in Natural Language Processing*; Springer: Berlin, Germany, 2018; pp. 219–253.
63. Araque, O.; Corcuera-Platas, I.; Sanchez-Rada, J.F.; Iglesias, C.A. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Syst. Appl.* **2017**, *77*, 236–246. [CrossRef]
64. Liu, J.; Chang, W.-C.; Wu, Y.; Yang, Y. Deep learning for extreme multi-label text classification. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 115–124.
65. Chen, M.; Wang, S.; Liang, P.P.; Baltrušaitis, T.; Zadeh, A.; Morency, L.-P. Multimodal sentiment analysis with word-level fusion and reinforcement learning. In Proceedings of the 19th ACM International Conference on Multimodal Interaction, Glasgow, UK, 13–17 November 2017; pp. 163–171.
66. Al-Sallab, A.; Baly, R.; Hajj, H.; Shaban, K.B.; El-Hajj, W.; Badaro, G. Aroma: A recursive deep learning model for opinion mining in arabic as a low resource language. *ACM Trans. Asian Low-Resour. Lang. Inf. Process. TALLIP* **2017**, *16*, 1–20. [CrossRef]
67. Kumar, S.; Gahalawat, M.; Roy, P.P.; Dogra, D.P.; Kim, B.-G.J.E. Exploring Impact of Age and Gender on Sentiment Analysis Using Machine Learning. *Electronics* **2020**, *9*, 374. [CrossRef]
68. Available online: http://help.sentiment140.com/site-functionality (accessed on 12 March 2020).
69. Available online: https://www.kaggle.com/crowdflower/twitter-airline-sentiment (accessed on 12 March 2020).
70. Available online: http://alt.qcri.org/semeval2017/ (accessed on 12 March 2020).
71. Available online: https://www.kaggle.com/c/word2vec-nlp-tutorial/data (accessed on 12 March 2020).
72. Maas, A.L.; Daly, R.E.; Pham, P.T.; Huang, D.; Ng, A.Y.; Potts, C. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Portland, OR, USA, 19–24 June 2011; pp. 142–150.
73. Available online: http://www.cs.cornell.edu/people/pabo/movie-review-data/ (accessed on 12 March 2020).
74. Blitzer, J.; Dredze, M.; Pereira, F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, 23–30 June 2007; pp. 440–447.
75. Kim, Y.; Sidney, J.; Buus, S.; Sette, A.; Nielsen, M.; Peters, B. Dataset size and composition impact the reliability of performance benchmarks for peptide-MHC binding predictions. *BMC Bioinform.* **2014**, *15*, 241. [CrossRef]
76. Choi, Y.; Lee, H.J. Data properties and the performance of sentiment classification for electronic commerce applications. *Inf. Syst. Front.* **2017**, *19*, 993–1012. [CrossRef]
77. Neppalli, V.K.; Caragea, C.; Squicciarini, A.; Tapia, A.; Stehle, S.J. Sentiment analysis during Hurricane Sandy in emergency response. *Int. J. Disaster Risk Reduct.* **2017**, *21*, 213–222. [CrossRef]

Hindawi
Complexity
Volume 2021, Article ID 9986920, 16 pages
https://doi.org/10.1155/2021/9986920

Research Article

Hybrid Deep Learning Models for Sentiment Analysis

Cach N. Dang,[1,2,3] **María N. Moreno-García**,[2] **and Fernando De la Prieta**[3]

[1]*Department of Information Technology, Ho Chi Minh City University of Transport (UT-HCMC), Ho Chi Minh 70000, Vietnam*
[2]*Data Mining (MIDA) Research Group, University of Salamanca, Salamanca 37007, Spain*
[3]*Biotechnology, Intelligent Systems and Educational Technology (BISITE) Research Group, University of Salamanca, Salamanca 37007, Spain*

Correspondence should be addressed to Cach N. Dang; cach@ut.edu.vn

Received 2 April 2021; Accepted 6 August 2021; Published 13 August 2021

Academic Editor: Tao Jia

Sentiment analysis on public opinion expressed in social networks, such as Twitter or Facebook, has been developed into a wide range of applications, but there are still many challenges to be addressed. Hybrid techniques have shown to be potential models for reducing sentiment errors on increasingly complex training data. This paper aims to test the reliability of several hybrid techniques on various datasets of different domains. Our research questions are aimed at determining whether it is possible to produce hybrid models that outperform single models with different domains and types of datasets. Hybrid deep sentiment analysis learning models that combine long short-term memory (LSTM) networks, convolutional neural networks (CNN), and support vector machines (SVM) are built and tested on eight textual tweets and review datasets of different domains. The hybrid models are compared against three single models, SVM, LSTM, and CNN. Both reliability and computation time were considered in the evaluation of each technique. The hybrid models increased the accuracy for sentiment analysis compared with single models on all types of datasets, especially the combination of deep learning models with SVM. The reliability of the latter was significantly higher.

1. Introduction

Sentiment analysis on information from social networks, such as Twitter or Facebook, is a research topic of growing interest today. Although much work has been done in this area, there are still many challenges to be addressed, including improving model reliability, reducing processing time, and applying techniques developed for specific types of data and specific data domains [1]. In recent years, deep learning models have been extensively applied in the field of sentiment analysis, where their great potential has been demonstrated.

Several studies are focused exclusively on building a single model from a single (or some) dataset(s) in a particular domain, such as marketing strategies [2], financial forecasting [3–5], and medical analysis [6, 7]. For social network applications, sentiment polarity-based deep learning applied to tweets is described thoroughly in [8–14]. Hassan and Mahmood [15] proved that CNN and recurrent neural networks (RNN) models can overcome the shortcoming of short text in deep learning models. Besides, the study by Qian et al. [16] revealed that LSTM behaves efficiently when used on different text levels of weather-and-mood tweets. After reviewing some recent studies [1, 11, 12, 15, 17–20], we found that CNN and RNN are outperforming methods with a relatively high overall accuracy. Both shallow neural networks and deep neural networks are capable of approximating any function. However, when contrasted to shallow neural networks, deep neural networks have the advantage of being able to do the feature extraction in the process of learning on large datasets. This is primarily because the deep models are able to extract/build better features than shallow models, using the intermediate hidden layers to achieve this [21, 22]. For the same level of accuracy, deep neural networks can be much more efficient in terms of computation and number of parameters. Deep neural networks are able to create deep representations; at every layer, the network learns a new, more abstract representation of the input.

Although a single machine learning method is relatively reliable when applied within certain domains, each deep learning approach has its own advantages and disadvantages. LSTM normally yields better results but requires more processing time than CNN, and CNN requires fewer hyperparameters and less supervision. Meanwhile, the LSTM performs more accurately for long sentences but requires a longer time to process [1].

The approach of combining two (or more) methods is introduced [23–25] as a means of incorporating the advantages of both and thus fills some shortcomings of individual methods. Alfrjani et al. [25] combined machine learning and semantic knowledge base for improving accuracy of sentiment analysis on reviews (improvement 1% to 6%). In another case, Gupta and Joshi [23] proposed a hybrid method that combines lexicon and machine learning for sentiment analysis on tweets (improvement 2% to 6%). A hybrid system with collaborative functions, therefore, is better able to address potential pitfalls, if any exist, associated with one single system. The effectiveness of the integrated models may vary based on different tasks. The CNN enhanced by SVM [26–28], CNN with RNN [29–32], and Lexicon-based analysis with machine learning [33, 34] showed an enhanced result. The combination of CNN, LSTM, and SVM aims to take advantage of the two deep network architecture models and SVM algorithms when performing sentiment analysis on different domains and types of datasets. Moreover, there are different types of input data obtained from social networks, such as tweets and reviews. Within and across these types, the input data also contains differences, for example, the distribution of the lengths of the tweets and reviews, the diversity of topics in each dataset, the sample size, and the greater or lesser presence of explicit sentiments and irrelevant information. Some approaches may be unable to perform well in different domains, with inadequate accuracy and performance in sentiment analysis [1, 35]. As a result, certain approaches may be ill-suited and difficult to apply to certain types of input data.

A question raised in our study is whether hybrid models perform better than single models regardless of the characteristics of the datasets. Therefore, our work examines how selected hybrid models behave with different types of datasets from different domains. In this work, we evaluated and validated the combination of three models CNN, LSTM, and SVM. We considered the relationship between models and its advanced capacities to extract characteristics, store past information and nodes, and classify text. First, in the initial stages of the model, two possible variations in the sequence of CNN and LSTM are introduced. Then, for each of these alternatives, two new variations are introduced: the use of CNN with ReLU function or SVM. We applied these models with word embedding on eight datasets, including tweets and reviews. The results of our experiments showed that the combined models increased the accuracy of sentiment analysis.

This paper offers three important contributions to the literature by highlighting four hybrid deep learning models for sentiment analysis that results in improved accuracy regardless of the types of social network datasets; providing an experimental study to evaluate the performance of hybrid deep learning models; and detailing a performance comparison of sentiment analysis methods with some state-of-art methods.

The paper is organized as follows. Section 2 presents an overview of related work; Section 3 describes the methodology in this research area; Section 4 contains the proposed hybrid models; Section 5 describes and discusses the results of our experiments; and Section 6 offers our conclusions.

2. Related Work

The purpose of this study is to build hybrid models for sentiment analysis that can improve accuracy. We have previously examined the methods proposed and applied in other studies, which are discussed as follows.

There are many ways to build hybrid models. In [26–28], the authors combined a CNN model and SVM that can improve the accuracy in image recognition. A convolutional network layer is used for extracting features and SVM functions as a recognizer. Original CNN is used with Softmax functions. Srinidhi et al. [36] proposed a hybrid model that combined LSTM and SVM with a radial basis function kernel for the textual classification of positive and negative sentiments. The hybrid model was evaluated on the IMDb movie review datasets. These models are combined from single deep learning models with SVM for classification. Some of them are applied for image recognition. Our research combines two deep learning models and then uses SVM or ReLU for classification.

Akhtar et al. [37] built a hybrid deep learning architecture, which is highly efficient for sentiment analysis in resource-poor languages. They used CNN for learning sentiment embedded vectors and SVM for sentiment classification. The model was tested on four Hindi datasets covering varied domains. Vo et al. [31] used a multichannel LSTM-CNN model for sentiment analysis on reviews/comments from e-commerce sites. In addition, hybrid CNN–LSTM models are applied for sentiment analysis on movie reviews by Rehman et al. [30]. The same techniques are used in several works, for example, [29, 38–40]. Kaur et al. [41] designed an algorithm called a hybrid heterogeneous support vector machine (H-SVM). They performed sentiment analysis on Twitter data related to COVID-19. Kastrati et al. [42] employed three different deep learning models such as CNN, LSTM, and CNN-LSTM for classifying Facebook comments related to the COVID-19 pandemic. They used pretrained word embedding method called FastText (an extension to Word2vec proposed by Facebook in 2016) and a contextualized word embedding model, BERT, to learn and generate word vector. Both research scored tweet/comment as positive, negative, or neutral. However, these models were individually tested on different datasets in a particular domain or tested on few sample datasets. Therefore, their validity is not generally proven.

A study by Jnoub et al. [19] focused on providing a generalized model for sentiment analysis that combined CNN with their own algorithm to transform reviews to

vectors. The model was evaluated on three different datasets: IMDb, movie reviews, and their own dataset collected from Amazon reviews. Ombabi et al. [43] proposed a hybrid deep learning model that combines CNN and LTSM. In addition, FastText is used for word embedding and SVM for classification in the Arabic language. In our work, both Word2vec and BERT were applied for word embedding. We proposed four types of hybrid deep learning models based on CNN, LSTM, and SVM for classifying both tweets and reviews.

Furthermore, other studies combine Lexicon-based analysis with machine learning [33, 34] or sentiment lexicons and polarity shifting devices [44]. The research by Sánchez-Rada and Iglesias [24] deals with the problem of user and content sentiment classification. They proposed a hybrid model that merges features from different levels of social context. The model is evaluated in different datasets. A study from Wang et al. [45] presented a hybrid approach, in which sentiment analysis of reviews about movies is used to improve a preliminary recommendation list obtained from the combination of collaborative filtering and content-based methods. In the same approach, the use of a sentiment classifier induced from movie reviews as a second filter after collaborative filtering was proposed by Pandey et al. [10]. These research projects use traditional techniques to perform sentiment analysis. Our research applies deep learning techniques for improving the accuracy of sentiment classification.

Recently, transfer learning has been successfully applied in sentiment analysis, in which lower network layers are trained on high-resource supervised datasets, such as BERT (proposed by researchers at Google AI language in 2018 [46]) and XLNET [47]. Examples can be found in [48–51], where BERT and XLNET were applied for sentiment analysis. The evaluation of different datasets and languages provides significant results. However, it also requires sufficiently powerful hardware, large datasets, and long processing times when applying these techniques. For example, BERT-Base model has 110M parameters, and BERT-Large model has 340M parameters: pretraining is fairly expensive, requiring four days on 4 to 16 cloud TPUs.

3. Methodology

Considering all of the advantages and potential of hybrid models and aiming at improving the performance of sentiment analysis techniques, our paper evaluates four hybrid models. The methodology is focused on three main components: the data to be used; process to build the feature vectors; building of hybrid methods for an appropriate sentiment analysis solution. These algorithms are applied to predict the sentiment polarity of the text and classify it according to that polarity.

3.1. Datasets. Our study does not focus on solving a problem in a particular domain but on providing an evaluation for general application models. In this study, we used several public datasets instead of generating and labelling new datasets of a specific application domain. Multiple criteria were considered for the selection including the ability to avoid privacy concerns [52], acceptance in the research community, diversity of sources and topics, and size. The selected datasets enable a comprehensive comparison of the sentiment analysis approaches examined in this paper. The aim of the experiment is to understand whether the models give consistently accurate results regardless of the dataset type and size.

The experiments were conducted using eight datasets. Three datasets contain tweets (Sentiment140, Tweets Airline, and Tweets SemEval) and five datasets contain reviews (IMDb movie reviews (1) and (2) and Cornell movie review). Among the tweets datasets, Sentiment140 [53], the largest, has 1.6 million tweets, each one labelled as either positive or negative sentiment, while the others, Tweets Airline [54] and Tweets SemEval [55], contain 14,640 and 17,750 tweets, respectively, labelled as positive, negative, or neutral. The five review datasets include a total of 125,000 comments from user reviews of movies (IMDb movie reviews (1) [56], IMDb movie reviews (2) [57], and Cornell movie reviews [58]), books, and music (book and music reviews [59]), labelled as either positive or negative sentiments. They are discussed in more detail in [1].

After examining the collected datasets, we saw that six out of eight datasets are initially labelled as positive and negative, and the sample on each label is relatively equal. The two datasets Airline and Tweet SemEval contain not only positive and negative labels but also neutral label. Having a balanced class distribution is important to ensure that prior probabilities are not biased for training models and doing classification [60]. In this research, we focus on polarity sentiment analysis, based on two classes positive and negative. The size of these datasets was reduced by removing the neutral labels. The remaining positive and negative classes are readjusted to be balanced. In addition, we applied k-fold cross-validation to the data in order to evaluate the models. In this way, the tests cover all instances of the datasets avoiding bias towards a particular subset of the data. Table 1 shows the number of samples (positive and negative) taken from each of dataset for performing experiments.

3.2. Preprocessing and Building the Feature Vector. Sentiment classification can be carried out on three levels of extraction: the document, sentence, and aspect or feature [61]. In our experiments, we applied document-based sentiment analysis with word embedding techniques on eight datasets of tweets and reviews. Sentiment analysis requires that text-training data be cleaned before using as input for classification models. Irrelevant information in text or sentence data, including white space, punctuation, and stop words, is removed. Two techniques commonly used for this task are TF-IDF and word embedding. Our proposal uses the latter because it provides better results than TF-IDF [1]. We then used word embedding models, BERT and Word2vec, to build the feature vector.

BERT is a language model for nature language processing, and it was published by researchers at Google AI Language in 2018 [46]. BERT was developed after Word2vec

Table 1: Number samples of datasets.

#	Datasets	Number of samples
1	Sentiment140 (10%)	160.000
2	Tweets Airline	4.726
3	Tweets SemEval	9.300
4	IMDb movie reviews (1)	50.000
5	IMDb movie reviews (2)	25.000
6	Cornell movie reviews	10.662
7	Book reviews	2.000
8	Music reviews	2.000

and includes some advances over Word2vec, such as support for out-of-vocabulary (OOV) words.

Word2vec was published in 2013 by Tomas Mikolov at Google [62]. This unsupervised learning model has trained datasets from a large corpus. The dimension of Word2vec is much less than the dimension of one-hot encoding, with a matrix NxD, with N being the number of documents and D being the dimension of word embedding. Word2vec contains two models: skip-gram and continuous bag-of-words (CBOW). Both models are based on the probability of words occurring in proximity to each other. Skip-gram allows us to start with a word and predict words that likely surround it. However, one of the major drawbacks of using Word2vec is a lack of support for out-of-vocabulary words. To work around this issue, we use the special token [UNK] for words not found in the vocabulary. In addition, we also retrain the Word2vec model according to our vocabulary datasets with all words that appear more than five times, reducing the use of the special token.

One issue in conducting sentiment analysis modelling is the varying length of the samples of the dataset. While deep learning models require fixed input vectors. Figures 1 and 2 show histograms of the datasets of reviews and tweets after they were cleaned. The x-axis represents the length of the data samples, and the y-axis is the frequency of appearance. Some histograms are rather ragged because we chose different types of datasets from different sources. Standardizing data by smoothing outlines based on sample size could well fit the models [63]. In this study, we keep nearly raw data for sentiment analysis with the purpose of creating the necessary conditions to compare the efficiency of other models.

We can see in Figures 1 and 2 that the data samples are quite widely varied in length. Therefore, it is necessary to set the data samples to the same length. The conversion of data samples adjusted to the same length is done as follows.

For each dataset, we select a fixed length called d; for samples shorter than d, we add zeros to the end of the vector. And vice versa, in samples with length greater than d, the back will be cut off. However, truncating the length of the data sample will result in a loss of information used in the classification process, so it is pivotal to choose a fixed length d to minimize truncation of the data samples. In this study, we used both tweets and reviews datasets for our proposed models. We truncated any tweet or review if its length is longer than the length of the feature vector. The length of the feature vector is chosen to be close to the maximum length of tweets and reviews, so very few samples were truncated in the dataset. This is commonly done in other works [30, 64–66].

The fixed length d is selected as follows: datasets related to tweets usually have a small length variation due to the limit of tweets to a maximum of 280 characters; thus, this fixed length d is chosen to be the maximum length of the sample in the dataset. For the remaining datasets, the length d is selected from 300 to 500, based on the histogram of every single dataset. It could be possible to take a fixed length d, instead of different lengths, for tweets and reviews. However, if set length d is larger, it will waste much memory, and if set length d is smaller, it will miss some review data.

3.3. Hybrid Methods. There are numerous methods to build up a hybrid model for sentiment analysis. In this study, we tested the combination of several successful approaches. As shown in Figure 3, we start by using Word2vec or a pretrained BERT model to create the feature vector. We then vary the order of the CNN and LSTM models used in the next stages: Word2vec/BERT - > CNN - > LSTM or Word2vec/BERT - > LSTM - > CNN. We also vary the final stage of the model, using a ReLU function or using an SVM.

Combining these two types of variation yields the four hybrid approaches that we have tested:

(1) Word2vec/BERT - > CNN - > LSTM - > Relu
(2) Word2vec/BERT - > LSTM - > CNN - > Relu
(3) Word2vec/BERT - > CNN - > LSTM - > SVM
(4) Word2vec/BERT - > LSTM - > CNN - > SVM

Two approaches were used in our experiments to create feature vectors. The first approach was Word2vec initialized with random weights to learn the embedding for all words in our training datasets. Because Word2vec does not include contextual analysis to handle complex semantical or polymorphic cases in natural languages, our second approach was BERT. A pretrained BERT model was used in this study. After adjusting the parameters, the BERT model was used as a feature extractor to generate input data for the proposal of hybrid models. The tweets and reviews data were fed into the BERT model to generate the feature vectors, which are the input to the hybrid models that perform the classification.

The next step combines CNN and LSTM deep learning models, which are used because of their good performance on sentiment analysis [1], as well as taking advantage of the two network architectures when performing sentiment analysis on data in different domains. A CNN is a type of feedforward neural network, since it is composed of multiple layers that process and pass information in one direction, from input to output, without cycles. It has a deep neural network architecture [67], typically starting with convolutional and pooling/subsampling layers that transform inputs that feed into a fully connected classification layer. In this research, a single convolutional (1D CNN) was used. LSTM is one of the many variations of the RNN architecture [68]. The LSTM block consists of three so-called gates, the forget gate, input gate, and output gate, in addition to the input and output blocks and the memory cell. CNNs are good at

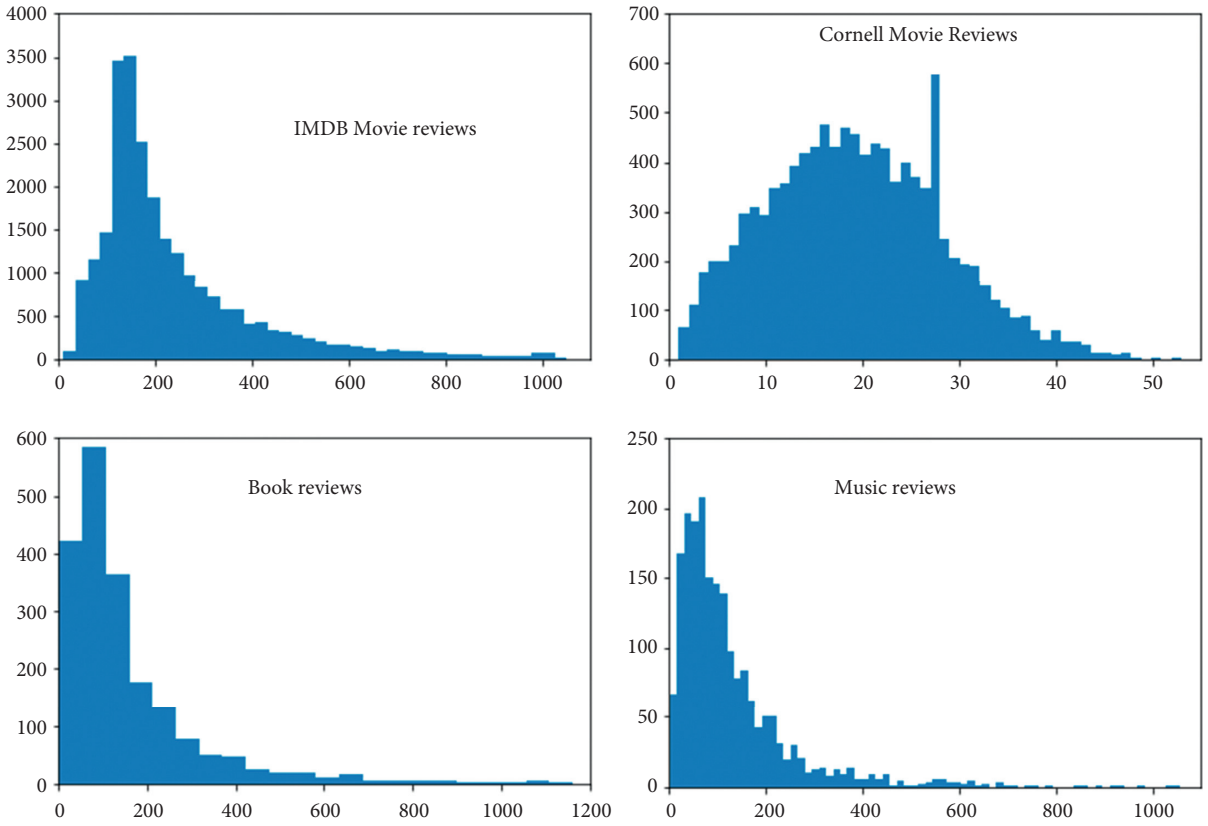


Figure 1: Histograms for different length data samples of reviews datasets.

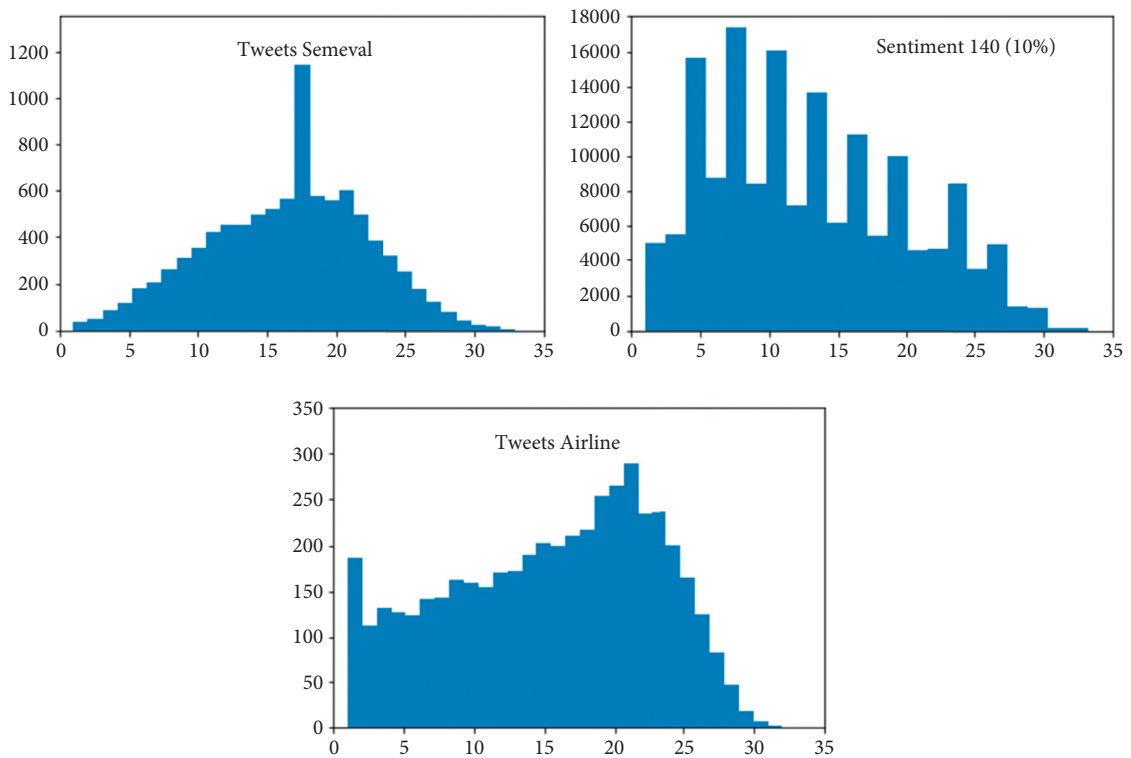


Figure 2: Histograms for different length data samples of tweets datasets.

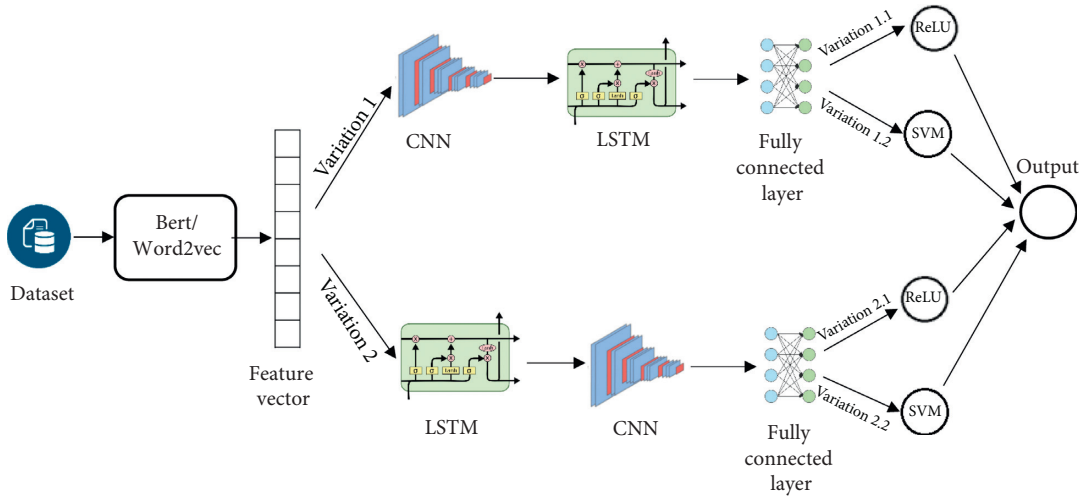


Figure 3: Process of methodology for sentiment analysis.

dealing with spatially related data while the RNNs are good at temporal signals. LSTM can remember forward information of the sequence, and multilayer CNN can catch and learn local information sufficiently. So, the combination makes use of the best of both worlds, the spatial and temporal worlds.

The final stage is classification. We use the activate function of ReLU instead of Sigmoid because of the high convergence. In addition, SVM was chosen for classification because of its efficiency in word processing, especially in high dimensional contexts, such as natural language processing. Support vector machine [69] is a supervised machine learning algorithm that can be used for both classification and regression tasks. It has been widely exploited with positive results in many areas. In our research, we have applied linear SVMs for classification with the proposed hybrid deep learning models. We extracted feature vectors from the top hidden layer and fed it to SVM that will classify for prediction ("positive" and "negative").

4. Proposed Hybrid Models

In this section, we proposed four hybrid deep learning models on variations in the use of CNN and LSTM in deep learning layers and variations of CNN and SVM in the classifier layers. The architecture of these hybrid models is shown in Tables 2 and 3, and the details are discussed as follows.

4.1. Scenario Combination 1. The first hybrid model combines CNN and LSTM models. The visualization of the model connection, the connection process, and the data processing flow are indicated in Table 2.

The function embedding is the embedding layer that is initialized with random weights, which will learn the embedding for all words in the training datasets. The first layer of the hybrid model is the CNN, which receives the vector produced by word embedding. It has three convolution layers consisting of 512, 256, and 128 filters, respectively, with a kernel size = 3, which receive and process data before feeding it into next deep learning layer. The second layer of the hybrid model is the LSTM, which produces a 1×500 matrix that is fed into the classifier. Next, the hybrid model's classifier is composed of two continuous, fully connected layers with 128 nodes and, finally, the output layer with a ReLU activation function.

Table 2: A hybrid CNN-LSTM model.

Layer (type)	Output shape	Param #
embedding_1 (embedding)	(None, 38, 100)	1,808,900
conv1d_1 (Conv1D)	(None, 38, 512)	154,112
conv1d_2 (Conv1D)	(None, 38, 256)	393,472
conv1d_3 (Conv1D)	(None, 38, 128)	98,432
lstm_1 (LSTM)	(None, 500)	1,258,000
dense_1 (dense)	(None, 128)	64,128
dense_2 (dense)	(None, 128)	16,512
dense_3 (dense)	(None, 1)	129
Total params: 3,793,685		
Trainable params: 1,984,785		
Nontrainable params: 1,808,900		

Table 3: A hybrid LSTM-CNN model.

Layer (type)	Output shape	Param #
embedding_2 (embedding)	(None, 38, 100)	1,808,900
lstm_2 (LSTM)	(None, 38, 500)	1,202,000
conv1d_3 (Conv1D)	(None, 38, 512)	768,512
conv1d_5 (Conv1D)	(None, 38, 256)	393,472
conv1d_6 (Conv1D)	(None, 38, 128)	98,432
flatten_1 (flatten)	(None, 4864)	0
dense_4 (dense)	(None, 128)	622,720
dense_5 (dense)	(None, 128)	16,512
dense_6 (dense)	(None, 1)	129
Total params: 4,910,677		
Trainable params: 3,101,777		
Nontrainable params: 1,808,900		

4.2. Scenario Combination 2. The second hybrid model combines LSTM and CNN models. The visualization of the

model connection, the connection process, and the data processing flow are indicated in Table 3.

The input data is preprocessed to reshape data for the embedding matrix. The first layer of the hybrid model is the LSTM layer. That output has a matrix 13×500 and is fed into the second model of the hybrid deep learning model. The next layer of the hybrid model is the CNN. It has three convolution layers consisting of 512, 256, and 128 filters, respectively, with a kernel size = 3, which are in charge of receiving and processing data before feeding it into the next layer. The CNN output is flattened and transferred to a fully connected layer. Finally, the hybrid model's classifier is a CNN composed of two continuous fully connected layers with 128 nodes and the ReLU activation function as the output layer.

4.3. Scenario Combinations 3 and 4. Our final hybrid model is based on the hybrid models from scenarios 1 and 2. We used the deep learning stages from those models (CNN-LSTM and LSTM-CNN) but replaced the classifier. While there are multiple alternatives to the CNN-based ReLU function used, we have chosen to use SVM for the replacement classifier. Scenario 3 is based on CNN-LSTM, and Scenario 4 is based on LSTM-CNN. An architectural overview of the model is shown in Tables 2 and 3.

5. Experimental Results

In this section, we present the experiments conducted to compare the performance of the proposed hybrid models. Moreover, we also examine other common deep learning models (SVM, CNN, and LSTM). All of them were tested with the eight datasets introduced in subsection 3.1 that have been preprocessed with text processing techniques. Accuracy, AUC, and F-score were the metrics used to evaluate the performance of the models through all experiments. Since F-score is derived from recall and precision, we also show these two measures for reference purposes. The results are shown, discussed, and analysed in Sections 5.2 and 5.3.

5.1. Performance Comparison. Before performing the experiments, the configuration of related parameters, hardware devices, and the necessary library facilities were carried out. We used Google Colab Pro with GPU Tesla P100-PCIE-16GB or GPU Tesla V100-SXM2-16GB [70] and the Keras [71] and TensorFlow libraries [72]. In all the experiments, we configured the parameter for our code, such as echoes = 4, k-fold = 10, and batch size = 32 with reviews and 128 with tweets. The common values for K-fold validation method are $k = 3$, $k = 5$, and $k = 10$, and by far, the most popular value used in applied machine learning to evaluate models is $k = 10$. The latter value is used when the dataset is large enough for the subsets to have a significant number of examples. This is the case of the datasets used in this work. Thus, nine parts are used as training set and one as test set in each of the 10 validations. The value of k is chosen to ensure that each train or test sample is large enough to represent the dataset. Furthermore, this procedure ensures that the k models in the cross-validation are induced from training sets of the same size and that the k test sets in all validations are also of the same size. It is recommended to split data into equal samples, so that the performance of the models is equivalent.

5.2. Results. The results of eight sets of experiments are shown: three baseline models (SVM, CNN, and LSTM) and four hybrid models: CNN and LSTM, LSTM and CNN, CNN-LSTM and SVM, LSTM-CNN and SVM referred to as C-LSTM (or C-L), L-CNN (or L-C), CLSTM-SVM (or CL-S), LCNN-SVM (or LC-S), respectively. A comparison analysis between the results obtained from the proposed hybrid methods against the baseline methods is also included.

Our experiments were run twice: once using Word2vec to train word embedding and once using a pretrained BERT model to train word embedding. The results were consistently better when BERT was used, so Tables 4–8 provide details on the experimental results using Word2vec and BERT. Figures 4–8 illustrate the comparative results obtained with Word2vec and BERT using side-by-side bar charts.

The accuracy results shown in Table 4 are very high for all datasets and classification models when using a pretrained BERT model to extract a feature vector, around 90%, especially, 92.9% in Tweets Airline, and 93.4% in IMDb movie reviews (1). Moreover, the results prove that hybrid models show higher (or equal) accuracy than single deep learning models (SVM, CNN, or LSTM) for seven out of eight datasets. Regarding the use of Word2vec in the music review and book review datasets, CNN's accuracy results given in Table 4 are 76.4% and 76.5%, respectively. By comparison, when using the LCNN-SVM model, the results significantly improve to 83.7% and 82.7%, which represent an improvement of 7.3% and 6.2%, respectively.

For the F-score (Table 7), hybrid models provided higher (or equal) values than single deep learning models for seven out of eight datasets. Regarding the AUC value in Table 8, the hybrid models also perform better than the single deep-learning models. The hybrid models using SVM for classification achieved the best results for six out of eight datasets using Word2vec. Among the datasets, the Tweets Airline dataset and IMDb movie reviews (1) are the datasets that show the highest values for all metrics in all cases. Book reviews and music reviews work well with hybrid LSTM-CNN and LCNN-SVM models. The Sentiment140 dataset has low accuracy in all models. In Figures 1 and 2, we can see the distribution of the total number of samples with the length data sample in the dataset. The Sentiment140 dataset is also different from the other datasets. Number samples of length data are so much different.

5.3. Discussion. As seen in Figures 4 to 8, using pretrained BERT produces better results than using Word2vec for sentiment analysis with all models and datasets. Focusing on the results of hybrid models, we see that, for each dataset, the best results are given by a hybrid model. Hybrid models

Table 4: Accuracy comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	79.7	80.3	80.1	79.9	**80.6**	79.9	82.4	84.0	84.1	84.0	**84.1**	83.5	83.9
Tweets Airline	82.2	86.8	87.1	88.0	87.5	**88.6**	87.8	92.0	92.9	92.8	92.8	92.8	**92.9**	92.7
Tweets SemEval	80.5	84.4	**86.4**	85.0	86.2	85.6	85.7	91.2	91.9	91.8	**91.9**	91.8	91.7	91.8
IMDb movie reviews (1)	78.9	87.6	88.5	89.7	90.0	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.8	87.3	85.1	89.2	**89.4**	**89.4**	**89.4**	90.5	90.7	90.6	**90.7**	**90.7**	**90.7**	**90.7**
Cornell movie reviews	67.7	72.4	76.1	73.0	**76.4**	76.2	75.9	85.3	87.0	86.9	86.9	**87.0**	86.9	**87.0**
Book reviews	77.2	76.4	77.2	75.8	83.5	78.7	**83.7**	89.9	90.7	91.0	90.7	**91.1**	90.4	91.0
Music reviews	76.6	76.5	79.6	70.9	82.1	76.6	**82.7**	87.8	89.2	89.2	89.0	89.1	88.8	**89.5**

Table 5: Recall comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	80.6	80.0	79.4	78.9	80.0	**80.5**	**84.7**	84.1	84.2	84.1	84.1	84.2	83.9
Tweets Airline	83.0	86.3	88.2	88.1	87.4	**88.8**	87.7	91.9	92.9	92.8	93.0	92.8	**93.1**	92.5
Tweets SemEval	80.1	84.1	87.4	**88.3**	86.4	85.2	85.0	91.6	92.2	92.1	**92.3**	91.6	91.8	92.1
IMDb movie reviews (1)	78.9	89.2	90.0	88.9	**90.1**	**90.1**	**90.1**	93.2	93.6	93.3	93.1	**93.6**	93.4	93.4
IMDb movie reviews (2)	82.9	86.9	85.4	87.6	**89.6**	**89.6**	89.3	90.6	90.9	90.7	**90.9**	90.8	**90.9**	90.8
Cornell movie reviews	67.1	73.6	75.1	71.1	**77.4**	75.7	75.6	84.5	**87.2**	86.6	86.6	86.8	86.7	87.1
Book reviews	72.6	78.0	78.2	76.3	**83.4**	78.2	**83.4**	90.7	90.8	90.8	90.5	**91.0**	90.9	**91.0**
Music reviews	75.7	75.8	79.4	70.8	80.8	76.5	**82.8**	87.7	88.5	88.8	88.2	88.3	87.9	**89.2**

Table 6: Precision comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.1	78.6	81.1	81.9	**82.0**	81.6	79.1	79.7	83.9	83.9	83.9	83.9	82.8	**84.0**
Tweets Airline	81.0	87.6	85.7	88.1	88.0	**88.4**	88.0	92.3	92.9	92.8	92.7	92.8	92.8	**93.1**
Tweets SemEval	81.1	82.2	83.0	78.3	83.6	83.7	**84.1**	90.7	91.7	91.5	91.5	**91.9**	91.5	91.5
IMDb movie reviews (1)	79	85.6	86.7	**91.0**	90.2	89.9	90.5	93.4	93.2	93.5	**93.7**	93.2	93.4	93.5
IMDb movie reviews (2)	82.7	87.9	84.8	**91.5**	89.3	89.2	89.6	90.3	90.6	90.6	90.5	**90.7**	90.4	90.5
Cornell movie reviews	69.8	70.8	78.4	**82.0**	74.8	77.3	76.3	86.3	86.9	**87.5**	87.4	87.4	87.1	86.9
Book reviews	**88.3**	74.8	76.3	76.3	84.0	79.9	84.0	89.0	90.7	91.2	90.9	**91.4**	89.8	91.1
Music reviews	79.7	81.4	80.1	76.7	**84.5**	77.2	82.7	88.1	90.2	89.9	**90.3**	**90.3**	90.1	90.0

Table 7: F-score comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.1	79.5	80.5	80.4	80.3	**80.8**	79.8	81.8	84.0	**84.1**	84.0	84.0	83.3	83.9
Tweets Airline	82.0	86.9	86.9	87.9	87.6	**88.5**	87.9	92.0	92.9	92.8	92.8	92.8	**92.9**	92.8
Tweets SemEval	80.6	83.1	84.9	82.8	**84.9**	84.4	84.5	91.1	91.9	91.8	**91.9**	91.8	91.6	91.8
IMDb movie reviews (1)	78.9	87.3	88.3	89.8	90.1	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.8	87.4	85.0	**89.5**	89.4	89.4	**89.5**	90.4	90.7	90.6	**90.7**	**90.7**	90.6	**90.7**
Cornell movie reviews	68.3	71.5	**76.6**	75.1	76.0	76.5	75.9	85.4	87.0	87.0	87.0	**87.1**	86.9	87.0
Book reviews	79.5	75.9	76.7	75.6	83.5	79.0	**83.7**	89.8	90.7	91.0	90.7	**91.1**	90.3	91.0
Music reviews	77.3	77.3	79.6	72.5	82.3	76.7	**82.7**	87.8	89.3	89.3	89.1	89.2	88.9	**89.6**

produced better results than single models using either Word2vec or BERT. With the use of Word2vec, the results of accuracy from hybrid models are higher than the ones from single models. Using BERT, the results have also improved although by a smaller amount since these models have reached a relatively high accuracy, mostly more than 90%.

Table 8: AUC comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	79.7	80.3	80.1	79.9	**80.6**	79.9	83.0	84.0	84.1	84.0	**84.1**	83.8	84.0
Tweets Airline	82.3	86.8	87.1	88.0	87.5	**88.6**	87.8	92.1	92.9	92.8	**92.9**	92.8	**92.9**	92.8
Tweets SemEval	80.5	84.3	**86.2**	84.6	86.0	85.5	85.6	91.2	92.0	91.8	**92.0**	91.8	91.7	91.8
IMDb movie reviews (1)	78.9	87.6	88.5	89.7	90.0	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.9	87.3	85.1	89.2	89.4	89.4	**89.5**	90.5	90.7	90.7	**90.7**	**90.7**	**90.7**	**90.7**
Cornell movie reviews	67.8	72.3	76.1	73.0	**76.4**	76.2	75.9	85.3	**87.1**	87.0	86.9	87.0	86.9	87.0
Book reviews	79.0	76.4	77.2	75.8	83.5	78.7	**83.7**	90.0	90.8	91.0	90.7	**91.2**	90.5	91.1
Music reviews	77.2	76.5	79.6	70.9	82.1	76.6	**82.7**	87.9	89.3	89.3	89.2	89.2	88.9	**89.6**

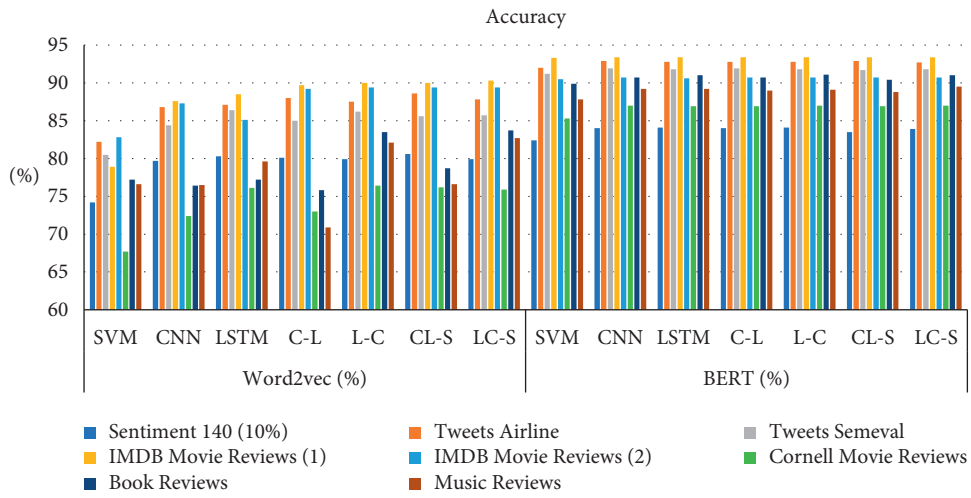


Figure 4: Accuracy values of deep learning models with Word2vec and BERT for different datasets.

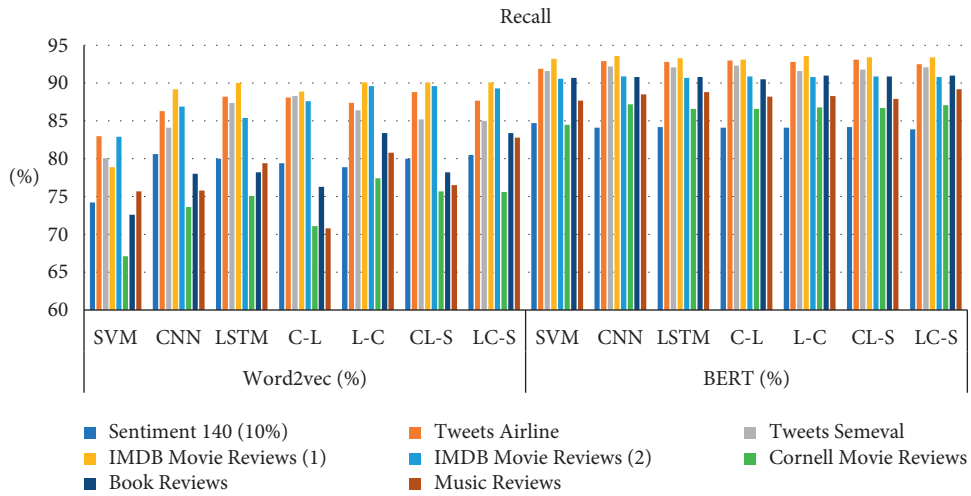


Figure 5: Recall values of deep learning models with Word2vec and BERT for different datasets.

The text in a review is normally longer than the text in a tweet, which suggests that LCNN-SVN performs better than other hybrid models on longer textual sample (Table 4). In selected datasets, when examining the distribution of the textual length of samples, the length of review ranges from 1 to 800 words. However, the Cornell movie reviews range from only 1 to 50 words. Besides, the length of a tweet ranges from 1 to 40 words; however, the distribution of sample length on Sentiment140 dataset is right skewed. It is observed that the results on two datasets, Sentiment140 and Cornell movie reviews, are lower than those of the remaining datasets.

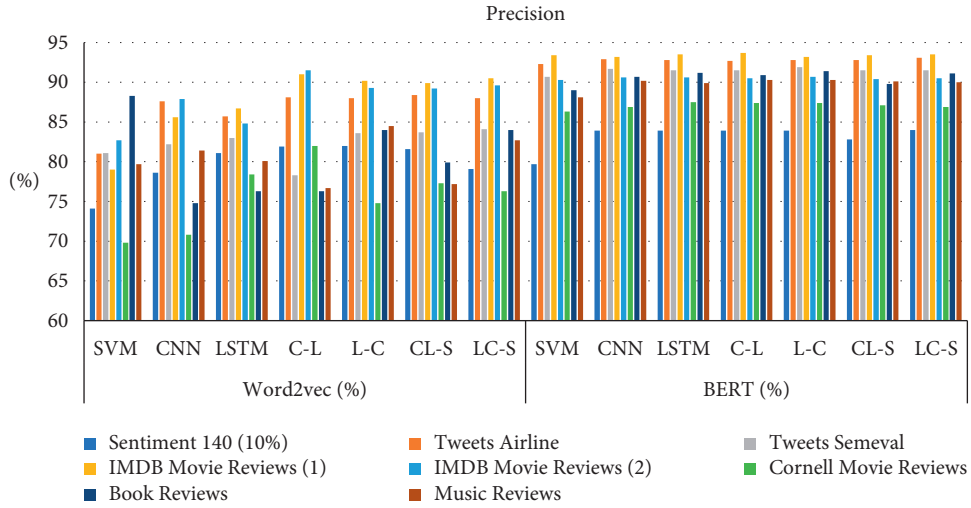


Figure 6: Precision values of deep learning models with Word2vec and BERT for different datasets.

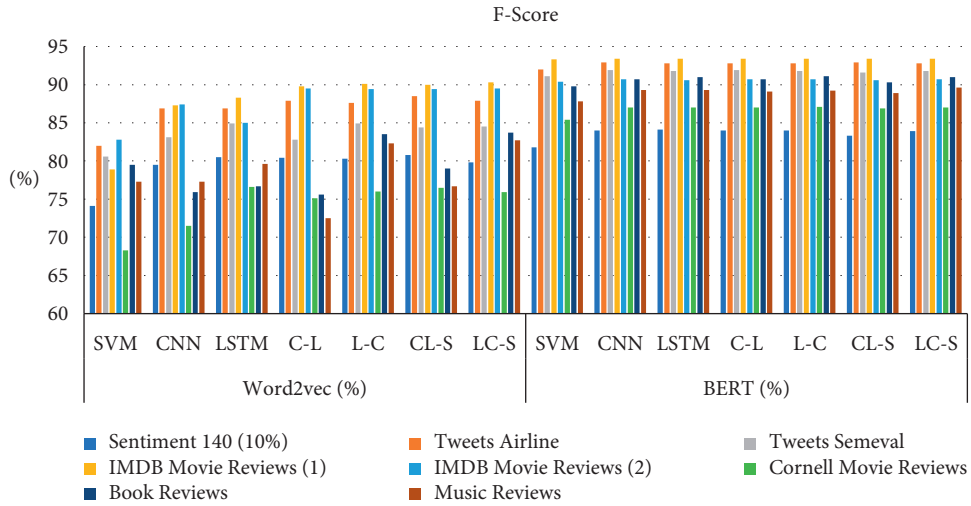


Figure 7: F-score values of deep learning models with Word2vec and BERT for different datasets.

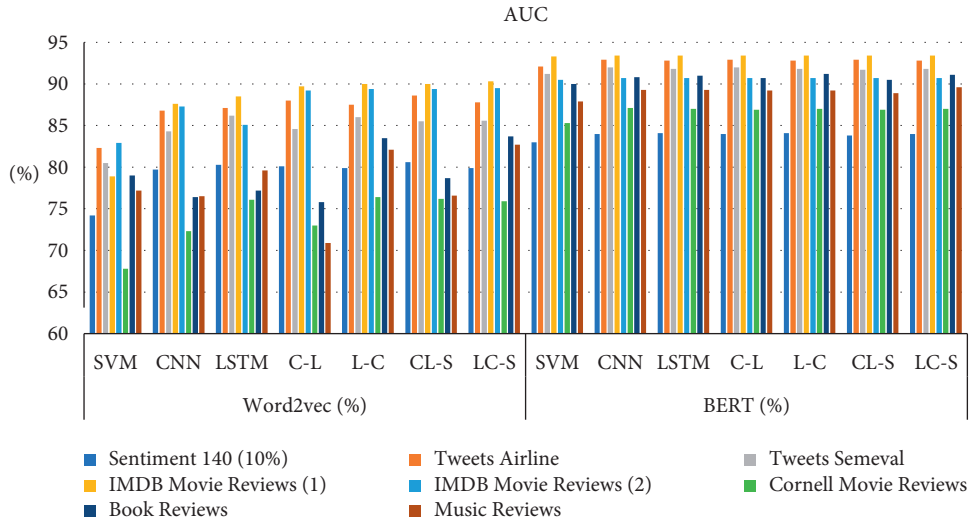


Figure 8: AUC values of deep learning models with Word2vec and BERT for different datasets.

Table 9: A comparison based on the proposed models and state-of-the-art approaches on datasets.

Study	Model	Dataset	Accuracy (%)
Kim and Jeong [18]	CNN	Cornell movie reviews	81
Maulana et al. [75]	SVM-IG	Cornell movie reviews	85.65
Proposed hybrid model	LCNN-SVM	Cornell movie reviews	**87**
Jnoub et al. [19]	SNN/CNN	IMDb	87/81
McCann et al. [20]	Char + CoVe-LSTM	IMDb	92.1
Tang et al. [76]	L-GRNN/Conv-GRNN	IMDb	45.3/42.5
Maltoudoglou et al. [49]	BERT	IMDb	92.28
Yang et al. [47]	XLNET	IMDb	96.21
Proposed hybrid model	CNN-LSTM	IMDb	**93.4**
Baziotis et al. [77]	Bi-LSTM + attention	Tweets SemEval	67.7 (F1)
Cliché [78]	LSTM-CNN	Tweets SemEval	68.5 (F1)
Proposed hybrid model	CNN-LSTM	Tweets SemEval	**91.9** (F1)
Abid et al. [12]	Bi-LSTM/CNN	Sentiment140	87.21/72.42
Han et al. [79]	FK-SVM	Sentiment140	87.2
Proposed hybrid model	LSTM-CNN	Sentiment140	**84.1**
Rane and Kumar [80]	AdaBoost	Tweets Airline	84.5
Duan et al. [81]	SVM and Naive Bayes	Tweets Airline	80
Monika et al. [17]	LSTM	Tweets Airline	80
Proposed hybrid model	CLSTM-SVM	Tweets Airline	**92.9**
Blitzer et al. [82]	SCL-MI	Music reviews/book reviews	79.7
Uribe [83]	Logistic/SVM	Music reviews/book reviews	87/89
Proposed hybrid model	LSTM-CNN/LC-S	Music reviews/book reviews	**91.1/89.5**

Some other studies performing sentiment analysis by using a single dataset of tweets or reviews are presented in [29, 33, 34, 37–39, 73, 74]. Note that the hybrid models provide much improved results in terms of processing time and accuracy. In addition, the overall accuracy of these hybrid models was given with eight different types of datasets, which give an objective view of overall accuracy.

Among the state-of-the-art approaches shown in Table 9, most of our hybrid models proposal got higher accuracy results on six datasets. On Sentiment140, however, Han et al. [79] and Abid et al. [12] achieved a better accuracy of around 87%. The XLNet method for sentiment analysis with IMDb dataset, performed by Yang et al. [47], resulted in 96.21% accuracy. On the other hand, Akhtar et al. [37] tested a hybrid model of combined CNN and SVM on both tweet and review datasets; however, the results showed a lower accuracy in comparison to hybrid methods, which were only tested on a single type of dataset (58.62% accuracy on tweet dataset and 77.16% accuracy with review dataset). The comparison details with the state-of-the-art approaches are shown in Table 9. It includes the authors' names, methods, datasets, and accuracy (or F1 for some studies that only provide the F1 measure).

In addition to the evaluation of the reliability of the models, it is also important to evaluate the performance of the algorithms in terms of resource utilization. There is very little work evaluating the computational complexity of deep learning models although there is some proposal [84] that considers some factors, such as the number of layers, the size of the input matrix, and other factors depending on the specific algorithm. In CNN, the number and size of convolution kernels and the number of output channels of each layer are considered. In view of this, it is clear that the higher reliability of hybrid models comes at the cost of higher complexity. Since time is one of the most valuable resources and the most taken into account when evaluating the performance of algorithms, we include the analysis of the computational time of the models involved in the comparative study, as this is a reflection of the time complexity.

Table 10 contains the time processing required for all datasets involved in the experiments. Processing time is calculated for the entire process of training and testing models using Word2vec and BERT. It includes time for data division and time to create the classification model (initialize the number of layers of the neural network, the number of nodes per layer, etc.) but does not include the time used to display the classification results. When using the hybrid models with the BERT technique for feature extraction, the accuracy generally is higher than with Word2vec, but the processing time is longer.

In general, the hybrid methods provide better results than single deep learning models. Most hybrid networks provide higher (or equal) scores in all datasets. Moreover, from the good result of Maltoudoglou et al. [49] (Table 9), we saw that the feature extraction plays an important role in sentiment classification. We also discussed the importance of feature extraction in [1], where TF-IDF and word embedding techniques for feature extraction were analysed. These improved results are high and stable at the expense of some increase in processing time, as shown in Table 10. The table shows that the hybrid model required longer computational time than the single models, because hybrid models are complex and feature many more parameters than single models. While the computational times are longer, they do not preclude analysis of the trade-offs between processing time and accuracy of results.

Our aim is to build a hybrid deep learning model for sentiment analysis that works well on various datasets of

Table 10: Time processing for experiments using a Google Colab Pro by dataset and model used.

Datasets	Word2vec							BERT						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	1h08m32	15m44	24m19	26m00	26m50	26m13	27m41	5h13m30	5h52m25	5h22m12	5h59m58	6h2m08	6h0m34	6h7m01
Tweets Airline	0m35	0m40	1m20	1m34	1m28	2m56	2m52	9m55	0h11m42	0h9m58	0h11m52	0h10m42	0h13m00	0h11m14
Tweets SemEval	1m18	1m32	2m04	2m30	2m20	3m55	3m06	8m46	0h9m46	0h8m40	0h10m12	0h10m07	0h14m28	0h37m48
IMDb movie reviews (1)	1h16m26	1h24m19	1h43m52	1h40m41	1h00m03	1h55m12	2h03m57	6h45m15	6h25m50	6h22m50	6h27m00	6h26m40	7h10m15	6h37m40
IMDb movie reviews (2)	39m42	48m04	0h54m03	58m27	1h03m23	1h0m42	1h05m15	3h24m20	3h11m00	3h9m30	3h11m40	3h11m25	3h36m15	3h28m05
Cornell movie reviews	2m16	2m46	3m42	4m49	4m19	6m04	5m45	16m12	0h15m26	0h14m10	0h15m55	0h15m49	0h24m03	0h19m41
Book reviews	2m23	5m31	6m46	9m40	8m35	10m16	9m21	32m22	0h33m11	0h32m37	0h33m23	0h33m49	0h33m59	0h58m18
Music reviews	1m43	2m41	4m36	5m53	5m21	5m21	5m45	18m23	0h18m44	0h18m17	0h18m55	0h19m03	0h20m14	1h5m57

domains. However, when building the classification models, there are many parameters that must be defined before, so they can be suitable for a given dataset but not for others. Therefore, the results obtained are positive and highly reliable because they have been evaluated on many datasets with different topics. Finally, general summaries of the results achieved in the experiments referenced earlier are discussed as follows:

(i) The hybrid models increased the accuracy for sentiment analysis compared with a single model performance on all types of datasets, although the computation time of SVM models is longer.

(ii) The combination helped to take advantage of the strengths of CNN, LSTM, and SVM, where CNN has the capability to extract characteristics, LSTM has capability to store past information at the state nodes (cell state), and SVM has capability to classify.

(iii) Using SVM as the classification method improved the results of both L-CNN and C-LSTM. SVM is effective in multidimensional data stratification and helps minimize local minima of neural networks.

6. Conclusions

In this paper, we proposed the use of hybrid deep learning models for sentiment analysis from social network data. We tested the performance of mixing SVM, CNN, and LSTM, using two-word embedding techniques, Word2vec and BERT, on eight textual datasets of tweets and reviews. Afterwards, we compared four generated hybrid models with single models. These experiments are conducted to understand the adaptability of hybrid models, whether hybrid approaches can adapt in a wide range of dataset types and sizes. We studied the influence of different types of datasets, feature extraction techniques, and deep learning models on reliability of sentiment polarity analysis.

Our experiments reveal that the reliability of hybrid models outperformed among all tested models for sentiment polarity analysis. Combining deep learning models with the SVM technique yields better results than using an individual model for performing sentiment analysis. In most of the tested datasets, the reliability of hybrid models using SVM is higher than that of the ones not using it; however, the computational time is much longer for the ones with SVM. We also observed that the effectiveness of the algorithms depends largely on the characteristics and quality of the datasets.

We are aware that the context of the dataset has a large impact on the choice of sentiment analysis models. We intend to study the performance of hybrid approaches for sentiment analysis on hybrid datasets and multiple or hybrid contexts in order to gain deeper insight in a specific topic, such as business, marketing, or medicine. Its application derives from associating sentiments to relevant context in order to provide detailed personal feedback and recommendation for users.

Data Availability

The datasets used to support the findings of this study are available from the direct link in the dataset citations.

Disclosure

The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Spanish Government and European (Fondo Europeo de Desarrollo Regional) FEDER funds, project InEDGEMobility: Movilidad inteligente y sostenible soportada por Sistemas Multi-agentes y Edge Computing (RTI2018-095390-B-C32).

References

[1] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment analysis based on deep learning: a comparative study," *Electronics*, vol. 9, no. 3, p. 483, 2020.

[2] M. J. S. Keenan, *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets: Bridging Fundamental and Technical Analysis*, Wiley, Hoboken, NJ, USA, 2nd edition, 2018.

[3] S. Sohangir, D. Wang, A. Pomeranets, and T. M. Khoshgoftaar, "Big data: deep learning for financial sentiment analysis," *Journal of Big Data*, vol. 5, no. 1, p. 3, 2018.

[4] H. Jangid, S. Singhal, R. R. Shah, and R. Zimmermann, "Aspect-based financial sentiment analysis using deep learning," in *Companion Proceedongs of the Web Conference 2018*, International World Wide Web Conferences Steering Committee, Lyon, France, April 2018.

[5] G. Wang, G. Yu, and X. Shen, "The effect of online investor sentiment on stock movements: an LSTM approach," *Complexity*, vol. 2020, Article ID 4754025, 11 pages, 2020.

[6] R. Satapathy, E. Cambria, and A. Hussain, *Sentiment Analysis in the Bio-Medical Domain*, Springer Interntional Publishing AG, Basel, Switzerland, 2017.

[7] A. Rajput, "Natural language processing, sentiment analysis, and clinical analytics," in *Innovation in Health Informatics*, pp. 79–97, Academic Press, Cambridge, MA, USA, 2020.

[8] V. Malik and A. Kumar, "Sentiment analysis of twitter data using Naive Bayes algorithm," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 6, no. 4, pp. 120–125, 2018.

[9] P. Vateekul and T. Koomsubha, "A study of sentiment analysis using deep learning techniques on Thai Twitter data," in *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, IEEE, Khon Kaen, Thailand, July 2016.

[10] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing & Management*, vol. 53, no. 4, pp. 764–779, 2017.

[11] A. S. M. Alharbi and E. de Doncker, "Twitter sentiment analysis with a deep neural network: an enhanced approach using user behavioral information," *Cognitive Systems Research*, vol. 54, pp. 50–61, 2019.

[12] F. Abid, M. Alam, M. Yasir, and C. Li, "Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter," *Future Generation Computer Systems*, vol. 95, pp. 292–308, 2019.

[13] A. M. Ramadhani and H. S. Goo, "Twitter sentiment analysis using deep learning methods," in *2017 7th International Annual Engineering Seminar (InAES)*, IEEE, Yogyakarta, Indonesia, August 2017.

[14] A. M. Khattak, R. Batool, F. A. Satti et al., "Tweets classification and sentiment analysis for personalized tweets recommendation," *Complexity*, vol. 2020, Article ID 8892552, 11 pages, 2020.

[15] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Third International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, Nagoya, Japan, April 2017.

[16] J. Qian, Z. Niu, and C. Shi, "Sentiment analysis model on weather related tweets with deep neural network," in *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, ACM, Zhuhai, China, February 2018.

[17] R. Monika, S. Deivalakshmi, and B. Janet, "Sentiment analysis of US airlines tweets using LSTM/RNN," in *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, IEEE, Tiruchirappalli, India, December 2019.

[18] H. Kim and Y.-S. Jeong, "Sentiment classification using convolutional neural networks," *Applied Sciences*, vol. 9, no. 11, p. 2347, 2019.

[19] N. Jnoub, F. Al Machot, and W. Klas, "A domain-independent classification model for sentiment analysis using neural models," *Applied Sciences*, vol. 10, no. 18, p. 6221, 2020.

[20] B. McCann, J. Bradbury, C. Xiong, and R. Socher, "Learned in translation: contextualized word vectors," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, December 2017.

[21] H. Mhaskar, Q. Liao, and T. Poggio, "When and why are deep networks better than shallow ones?" in *Proceedings of the 2017 AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2017.

[22] A. Schindler, T. Lidy, and A. Rauber, "Comparing shallow versus deep neural network architectures for automatic music genre classification," in *Proceedings of the 9th Forum Media Technology (FMT2016)*, FMT, Poelten, Austria, 2016.

[23] I. Gupta and N. Joshi, "Enhanced twitter sentiment analysis using hybrid approach and by accounting local contextual semantic," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 1611–1625, 2019.

[24] J. F. Sánchez-Rada and C. A. Iglesias, "CRANK: a hybrid model for user and content sentiment classification using social context and community detection," *Applied Sciences*, vol. 10, no. 5, p. 1662, 2020.

[25] R. Alfrjani, T. Osman, and G. Cosma, "A hybrid semantic knowledgebase-machine learning approach for opinion mining," *Data & Knowledge Engineering*, vol. 121, pp. 88–108, 2019.

[26] D.-X. Xue, R. Zhang, H. Feng, and Y.-L. Wang, "CNN-SVM for microvascular morphological type recognition with data augmentation," *Journal of Medical and Biological Engineering*, vol. 36, no. 6, pp. 755–764, 2016.

[27] M. Elleuch, R. Maalej, and M. Kherallah, "A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition," *Procedia Computer Science*, vol. 80, pp. 1712–1723, 2016.

[28] Y. Tang, "Deep learning using linear support vector machines," 2013, http://arxiv.org/abs/1306.0239.

[29] T. Chen, R. Xu, Y. He, and X. Wang, "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN," *Expert Systems with Applications*, vol. 72, pp. 221–230, 2017.

[30] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, "A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis," *Multimedia Tools and Applications*, vol. 78, no. 18, pp. 26597–26613, 2019.

[31] Q.-H. Vo, H.-T. Nguyen, B. Le, and M.-L. Nguyen, "Multi-channel LSTM-CNN model for Vietnamese sentiment analysis," in *2017 9th international conference on knowledge and systems engineering (KSE)*, IEEE, Hue, Vietnam, October 2017.

[32] C. A. Martín, J. M. Torres, R. M. Aguilar, and S. Diaz, "Using deep learning to predict sentiments: case study in tourism," *Complexity*, vol. 2018, Article ID 7408431, 9 pages, 2018.

[33] K. Elshakankery and M. F. Ahmed, "HILATSA: a hybrid Incremental learning approach for Arabic tweets sentiment analysis," *Egyptian Informatics Journal*, vol. 20, no. 3, pp. 163–171, 2019.

[34] S. J. Putra, I. Khalil, M. N. Gunawan, R. I. Amin, and T. Sutabri, "A hybrid model for social media sentiment analysis for Indonesian text," in *Proceedings of the 20th International Conference on Information Integration and Web-Based Applications & Services*, Yogyakarta, Indonesia, November 2018.

[35] P. Astya, "Sentiment analysis: approaches and open issues," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, IEEE, Greater Noida, India, May 2017.

[36] H. Srinidhi, G. Siddesh, and K. Srinivasa, "A hybrid model using MaLSTM based on recurrent neural networks with support vector machines for sentiment analysis," *Engineering and Applied Science Research*, vol. 47, no. 3, pp. 232–240, 2020.

[37] M. S. Akhtar, A. Kumar, A. Ekbal, and P. Bhattacharyya, "A hybrid deep learning architecture for sentiment analysis," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, December 2016.

[38] S. Al-Azani and E.-S. M. El-Alfy, "Hybrid deep learning for sentiment polarity determination of Arabic microblogs," in *International Conference on Neural Information Processing*, Springer, Guangzhou, China, November 2017.

[39] G. Liu, X. Xu, B. Deng, S. Chen, and L. Li, "A hybrid method for bilingual text sentiment classification based on deep learning," in *Proceedings of the 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, IEEE, Shanghai, China, May 2016.

[40] Q. Zhang, Z. Zhang, M. Yang, and L. Zhu, "Exploring co-evolution of emotional contagion and behavior for microblog sentiment analysis: a deep learning architecture," *Complexity*, vol. 2021, Article ID 6630811, 10 pages, 2021.

[41] H. Kaur, S. U. Ahsaan, B. Alankar, and V. Chang, "A proposed sentiment analysis deep learning algorithm for analyzing COVID-19 tweets," *Information Systems Frontiers*, pp. 1–13, 2021.

[42] Z. Kastrati, L. Ahmedi, A. Kurti et al., "A deep learning sentiment analyser for social media comments in low-resource languages," *Electronics*, vol. 10, no. 10, p. 1133, 2021.

[43] A. H. Ombabi, W. Ouarda, and A. M. Alimi, "Mining, "deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks," *Social Network Analysis and Mining*, vol. 10, no. 1, pp. 1–13, 2020.

[44] G. Yoo and J. Nam, "A hybrid approach to sentiment analysis enhanced by sentiment lexicons and polarity shifting devices," in *The 13th Workshop on Asian Language Resources*, Miyazaki, Japan, May 2018.

[45] Y. Wang, M. Wang, and W. Xu, "A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 8263704, 9 pages, 2018.

[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: pre-training of deep bidirectional transformers for language understanding," 2018, http://arxiv.org/abs/04805.

[47] Z. Yang, Z. Dai, Y. Yang et al., "Xlnet: generalized autoregressive pretraining for language understanding," 2019, http://arxiv.org/abs/1906.08237.

[48] A. Tela, A. Woubie, and V. Hautamaki, "Transferring monolingual model to low-resource language: the case of tigrinya," 2020, http://arxiv.org/abs/2006.07698.

[49] L. Maltoudoglou, A. Paisios, and H. Papadopoulos, "BERT-based conformal predictor for sentiment analysis," in *Proceedings of the 2020 Conformal and Probabilistic Prediction and Applications*, PMLR, Verona, Italy, September 2020.

[50] X.-R. Gong, J.-X. Jin, and T. Zhang, "Sentiment analysis using autoregressive language modeling and broad learning system," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, San Diego, CA, USA, November 2019.

[51] B. Myagmar, J. Li, and S. Kimura, "Cross-domain sentiment classification with bidirectional contextualized transformer language models," *IEEE Access*, vol. 7, pp. 163219–163230, 2019.

[52] S. Kumar, M. Gahalawat, P. P. Roy, D. P. Dogra, and B.-G. Kim, "Exploring impact of age and gender on sentiment analysis using machine learning," *Electronics*, vol. 9, no. 2, p. 374, 2020.

[53] "Sentiment140 - a twitter sentiment analysis tool," Available from: (accessed on 10 December 2020), http://help.sentiment140.com/site-functionality.

[54] "Twitter US Airline Sentiment," Available from: (accessed on 10 December 2020), https://www.kaggle.com/crowdflower/twitter-airline-sentiment.

[55] "International Workshop on Semantic Evaluation 2017, Available from: (accessed on 10 December 2020), http://alt.qcri.org/semeval2017/.

[56] "Large movie review dataset," Available from: (accessed on 10 December 2020), http://ai.stanford.edu/~7amaas/data/sentiment/.

[57] "Bag of words meets bags of popcorn," Available from: (accessed on 10 December 2020), https://www.kaggle.com/c/word2vec-nlp-tutorial/data?select=labeledTrainData.tsv.zip.

[58] "Cornell CIS computer science," Available from: (accessed on 10 December 2020), http://www.cs.cornell.edu/people/pabo/movie-review-data/.

[59] "Multi-domain sentiment dataset," Available from: (accessed on 10 December 2020), http://www.cs.jhu.edu/~mdredze/datasets/sentiment/.

[60] Y. Wan and Q. Gao, "An ensemble sentiment classification system of twitter data for airline services analysis," in *Proceedings of the 2015 IEEE International Conference On Data Mining Workshop (ICDMW)*, IEEE, Atlantic City, NJ, USA, November 2015.

[61] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: a survey," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 4, Article ID e1253, 2018.

[62] T. Mikolov, K. Chen, G. S. Corrado, and J. A. Dean, "Computing numeric representations of words in a high-dimensional space," Google Patents, 2015.

[63] N. Banić and N. Elezović, "TVOR: finding discrete total variation outliers among histograms," *IEEE Access*, vol. 9, pp. 1807–1832, 2020.

[64] B. Jang, M. Kim, G. Harerimana, S.-U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: combining Word2vec CNN and attention mechanism," *Applied Sciences*, vol. 10, no. 17, p. 5841, 2020.

[65] A. Jacovi, O. S. Shalom, and Y. Goldberg, "Understanding convolutional neural networks for text classification," 2018, http://arxiv.org/abs/1809.08037.

[66] H. T. Nguyen and M. Le Nguyen, "An ensemble method with sentiment features and clustering support," *Neurocomputing*, vol. 370, pp. 155–165, 2019.

[67] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[68] S. Hochreiter and J. Schmidhuber, "LSTM can solve hard long time lag problems," in *Proceedings of the 9th International Conference on Neural Information Processing Systems*, Denver, CO, USA, December 1997.

[69] M. M. Adankon, M. Cheriet, and A. Biem, "Semisupervised least squares support vector machine," *IEEE Transactions on Neural Networks*, vol. 20, no. 12, pp. 1858–1870, 2009.

[70] "Making the most of your Colab subscription," Available from: (accessed on 22 January 2021), https://colab.research.google.com/notebooks/pro.ipynb.

[71] "Keras: The Python deep learning API," Available from: (accessed on 10 December 2020, https://keras.io/.

[72] "TensorFlow," Available from: (accessed on 10 December 2020), https://www.tensorflow.org/.

[73] K. Ghasedi and H. Huang, "Sentiment analysis via deep hybrid textual-crowd learning model," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*, New Orleans, Louisiana, February 2018.

[74] M. U. Salur and I. Aydin, "A novel hybrid deep learning model for sentiment classification," *IEEE Access*, vol. 8, pp. 58080–58093, 2020.

[75] R. Maulana, P. A. Rahayuningsih, W. Irmayani, D. Saputra, and W. E. Jayanti, "Improved accuracy of sentiment analysis movie review using support vector machine based information gain," in *Proceedings of the International Conference on Advanced Information Scientific Development (ICAISD)*, IOP Publishing, West Java, Indonesia, August 2020.

[76] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference On Empirical Methods In Natural Language Processing*, Lisbon, Portugal, September 2015.

[77] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop On Semantic Evaluation (SemEval-2017)*, Vancouver, Canada, August 2017.

[78] M. Cliche, "Bb_twtr at semeval-2017 task 4: twitter sentiment analysis with cnns and lstms," 2017, http://arxiv.org/abs/1704.06125.

[79] K.-X. Han, W. Chien, C.-C. Chiu, and Y.-T. Cheng, "Application of support vector machine (SVM) in the sentiment analysis of twitter DataSet," *Applied Sciences*, vol. 10, no. 3, p. 1125, 2020.

[80] A. Rane and A. Kumar, "Sentiment classification system of Twitter data for US airline service analysis," in *Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, IEEE, Tokyo, Japan, July 2018.

[81] X. Duan, T. Ji, and W. Qian, *Twitter US Airline Recommendation Prediction*, p. cs229, Stanford University, Stanford, CA, USA, 2016.

[82] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification," in *Proceedings of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, June 2007.

[83] D. Uribe, "Domain adaptation in sentiment classification," in *2010 Ninth International Conference on Machine Learning and Applications*, IEEE, Washington, DC, USA, December 2010.

[84] H. Xie, M. Zhang, J. Ge, X. Dong, and H. Chen, "Learning air traffic as images: a deep convolutional neural network for airspace operation complexity evaluation," *Complexity*, vol. 2021, Article ID 6457246, 16 pages, 2021.

sensors

MDPI

Article

An Approach to Integrating Sentiment Analysis into Recommender Systems

Cach N. Dang [1,2,3,*], María N. Moreno-García [2] and Fernando De la Prieta [3]

[1] Department of Information Technology, HoChiMinh City University of Transport (UT-HCMC), Ho Chi Minh 70000, Vietnam
[2] Data Mining (MIDA) Research Group, University of Salamanca, 37007 Salamanca, Spain; mmg@usal.es
[3] Biotechnology, Intelligent Systems and Educational Technology (BISITE) Research Group, University of Salamanca, 37007 Salamanca, Spain; fer@usal.es
* Correspondence: cach@ut.edu.vn; Tel.: +84-919-101-086

Abstract: Recommender systems have been applied in a wide range of domains such as e-commerce, media, banking, and utilities. This kind of system provides personalized suggestions based on large amounts of data to increase user satisfaction. These suggestions help client select products, while organizations can increase the consumption of a product. In the case of social data, sentiment analysis can help gain better understanding of a user's attitudes, opinions and emotions, which is beneficial to integrate in recommender systems for achieving higher recommendation reliability. On the one hand, this information can be used to complement explicit ratings given to products by users. On the other hand, sentiment analysis of items that can be derived from online news services, blogs, social media or even from the recommender systems themselves is seen as capable of providing better recommendations to users. In this study, we present and evaluate a recommendation approach that integrates sentiment analysis into collaborative filtering methods. The recommender system proposal is based on an adaptive architecture, which includes improved techniques for feature extraction and deep learning models based on sentiment analysis. The results of the empirical study performed with two popular datasets show that sentiment–based deep learning models and collaborative filtering methods can significantly improve the recommender system's performance.

Keywords: sentiment analysis; deep learning; recommender system; natural language processing

Citation: Dang, C.N.; Moreno-García, M.N.; Prieta, F.D.l. An Approach to Integrating Sentiment Analysis into Recommender Systems. *Sensors* **2021**, *21*, 5666. https://doi.org/10.3390/s21165666

Academic Editor: Wataru Sato

Received: 20 July 2021
Accepted: 19 August 2021
Published: 23 August 2021

1. Introduction

With the explosion of blogs, forums, and online social networks, differing opinions about a particular topic can be easily found from millions of users. For example, users can discuss their current experiences, share their points of view on a specific fact, or praise or complain about a product that they have just bought. With a vast amount of available online data, sentiment analysis—a method to categorize text-based opinions to determine a user's attitude—can help gain better understanding of the attitudes, opinions and emotions of the public in several domains such as business, government, and biomedicine. Several studies are summarized and discussed in [1] regarding the benefits of sentiment analysis in obtaining feedbacks and determining the interests and opinions of customers.

Recommender systems, first developed in the mid-1990s and based on users' ratings and preferences, have expanded widely in recent decades. They are now especially important in the realms of e-commerce, media, banking, and utilities. This type of system is used by Amazon to suggest preferred products for customers, by YouTube to suggest related videos on the auto-play function, and by Facebook to recommend people and webpages to connect and follow.

Sentiment analysis can be beneficial to recommender systems. A sample of this can be found in the work of Preethi et al. [2], in which a cloud-based recommender system uses recursive neural networks to analyze sentiments of reviews in order to improve

and validate restaurant and movie recommendations. Along with behavioral analysis, sentiment analysis is also an efficient tool for commodity markets [3].

Social media data has been exploited in different ways to address some problems, especially associated with collaborative filtering approaches [4]. Methods in recommender systems are based on information filtering, and they can be classified into three categories: content-based; collaborative filtering (CF); and hybrid. Sparsity and gray-sheep problems are two of the main reasons CF methods do not provide the reliability required in some recommender systems [5]. In particular, when only sparse ratings data is available, sentiment analysis can play a key role in improving recommendation quality. This is because recommendation algorithms mostly rely on users' ratings to select the items to recommend. Such ratings are usually insufficient and very limited. On the other hand, sentiment-based ratings of items that can be derived from reviews or opinions given through online news services, blogs, social media or even the recommender systems themselves, are seen as capable of providing better recommendations to users. Sentiment-based models have been exploited in recommender systems to overcome the data-sparsity problem that exists in conventional recommender systems. Hence, integrating sentiment in recommender systems may significantly enhance the recommendation quality.

In this study, we propose a recommendation method that combines sentiment analysis and collaborative filtering. The method is implemented in an adaptive recommender system architecture in which techniques for feature extraction and deep learning-based sentiment analysis is included. The results of the empirical study performed with two popular datasets show that combining deep learning-based sentiment analysis and collaborative filtering methods significantly improve the recommender system's performance.

The rest of this paper is organized as follows. Section 2 presents background information and provides a literature review in this research area. Section 3 describes the methodology for recommender systems. Section 4 outlines the results and discussion, and Section 5 offers the main conclusion.

2. Background and Related work

Sentiment analysis is very useful in a wide range of application domains, including business, government, and education. Application of sentiment analysis in recommender system has also been the focus of extensive research. In this section, we start by presenting background information and reviewing the literature to offer an up-to-date overview of how sentiment analysis has been applied in recommender systems.

2.1. Sentiment Analysis

Sentiment analysis can be performed on three levels of extraction: the sentence level; the document level; and the aspect or feature level. It is a process of extracting information about an entity and automatically identifying any of the subjectivities of that entity. The aim is to determine whether text generated by users conveys their positive, negative, or neutral opinions. Three approaches currently exist to address the problem of sentiment analysis [6]: lexicon-based techniques; machine-learning-based techniques; and hybrid approaches. Lexicon-based techniques are divided into two approaches: dictionary-based and corpus-based [7]. They were the first to be used for sentiment classification. Machine learning-based techniques [8] that have been proposed for sentiment analysis include traditional techniques and deep learning techniques. The hybrid approaches is the combination of machine learning and lexicon-based approaches [9]. Sentiment lexicons regularly play a key role in most of these strategies. Figure 1 illustrates a taxonomy of deep learning-based methods for sentiment analysis.

Deep learning techniques can provide better results than traditional techniques. Different kinds of deep learning models can be used for sentiment classification, including CNN, DNN, and RNN. These models address classification problems at the document level, sentence level, or aspect level. In addition, some approaches that combine two models are

introduced [10–16]. The CNN enhanced by SVM [10–12], CNN with RNN [13–16] showed enhanced results.

The hybrid models can increase the accuracy for sentiment analysis in comparison to a single model performance. In this study, we combine deep learning techniques for sentiment analysis. The resulting hybrid deep-learning models for sentiment analysis, which combine LSTM networks [17] and CNN [18], are built and tested on two datasets containing reviews.

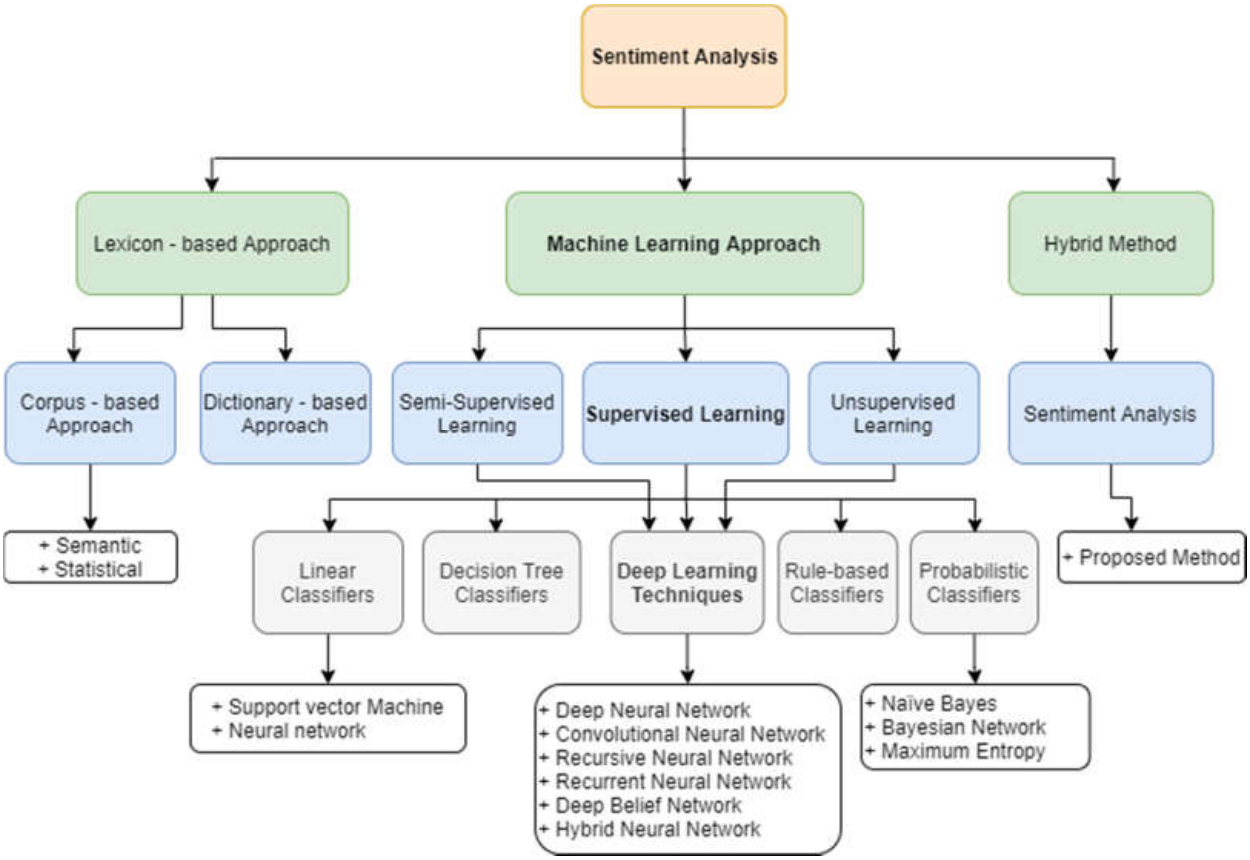


Figure 1. Taxonomy of sentiment analysis techniques. Source: [6,19].

2.2. Recommender Systems

A recommender system intends to provide personalized recommendations about products or services to support decision making in the continuous increase of online information. Several systems have been developed and applied in three main domains: business, government, and education, across eight categories: e-government, e-business, e-commerce/e-shopping, e-library, e-learning, e-tourism, e-resource services and e-group activities [20]. E-commerce has widely applied recommender systems to suggest additional products for customers to choose from among the multiple products available. A filtering technique has improved systems for presenting personalized choices [21].

The most common methods used for recommender systems may be grouped into three categories: content-based; collaborative filtering (CF) and hybrid recommender systems [22]. These techniques vary depending on the types of social media data that are used. Lu et al. [20] analyzed typical recommender systems and effectively identifies the specific requirements for recommendation techniques in the domain. This work also directly motivates and supports researchers and practitioners to promote the popularization and application of recommender systems in different domains.

Content-based recommender systems: Content-based methods make use of characteristics of items and users' profiles. User profiles are created by mining content information

about items accessed over the web by users, such as product attributes. Content-based recommender systems filter items based on the content-based similarity measures between items in the catalog and items that users have previously consumed, accessed, or rating positively. Therefore, a user receives recommendations of items like those that previously have been of interest. The utility of an item for a user can be a derivative done after a quantitative analysis of the metadata of the item.

Collaborative filtering-based recommender systems: Collaborative filtering is a technique that can filter out items that a user might like based on reactions by similar users. It works by searching in a large group of people and finding a smaller set of users with tastes like those of a particular user. It looks at the items they like and combines them to create a ranked list of suggestions. We need data that contains a set of items and a set of users to perform with recommender algorithms. While working with such data, the matrix consists of the reactions given by a set of users to certain items from within a set of items. Each row would contain the ratings given by a user, and each column would contain the ratings received by an item.

Hybrid recommender systems: Hybrid approaches take advantage of any kind of item and user information that can be extracted or inferred from web systems, social media, or other sources. Hybrid approaches are implemented by deployment individually as well as by accumulating rankings and predictions and then building a general consolidative model that resolves the common problems in recommender systems.

Each recommendation approach has advantages and limitations; for example, Collaborative Filtering has sparseness, scalability and cold-start problems [5,23,24]. A sparseness problem occurs when we have a vast amount of data. A scalability problem occurs when the rating data is missing. When a user or an item is added to the system the cold-start problem appears. Combining sentiment analysis with recommendation methods can help solve these problems. Figure 2 shows the categories of deep learning applied to information retrieval and recommender system research.

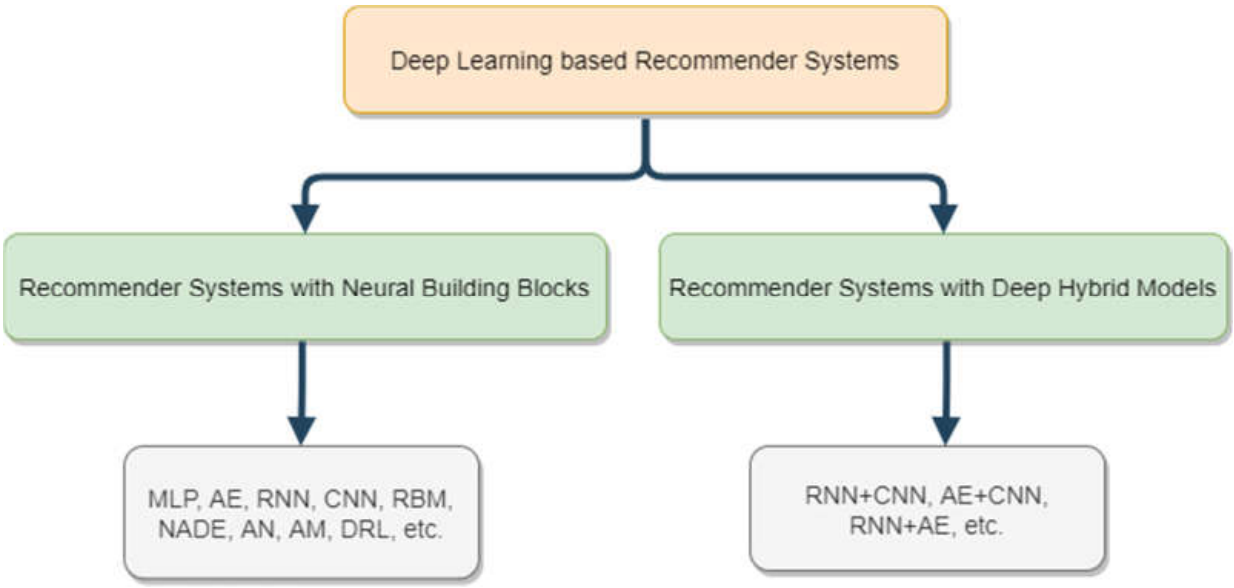


Figure 2. Categories of deep neural network-based recommendation models [25]. Multilayer Perceptron (MLP); Auto Encoder (AE); Convolutional Neural Network (CNN); Recurrent Neural Network (RNN); Restricted Boltzmann Machine (RBM); Neural Autoregressive Distribution Estimation (NADE); Adversarial Networks (AN); Attentional Models (AM); Deep Reinforcement Learning (DRL).

2.3. Related Work

Recommender systems can be improved in a variety of ways. In [4], social tag embedding is used in a collaborative filtering approach in which user similarities based on both

tag embedding and ratings are combined to generate the recommendations. Recommender systems have also benefited from sentiment analysis. An example of this can be found in the work of Preethi et al. [2], where recursive neural networks were applied to analyze sentiments in reviews. The output was used to improve and validate restaurant and movie recommendations of a cloud-based recommender system. Along with behavioral analysis, sentiment analysis is also an efficient tool for commodity markets [3]. Wang et al. [26] combined a hybrid recommender system and sentiment analysis to optimize the preliminary list and obtain the final recommendation list. Kumar et al. [27] proposed a hybrid recommender system by combining collaborative filtering and content-based filtering with the use of sentiment analysis of movie tweets to boost up the recommender system.

Rao et al. [28] designed a recommender system that contains the user list and item list with user reviews. Using the sentiment dictionaries, the researchers divided the items into three categories: brand, quality, and price. They leveraged sentiment dictionaries to calculate sentiment of a particular user on item/product. Gurini et al. [29] adopted a different approach to describe a user recommender system for Twitter. Their work emphasized the use of implicit sentiment analysis in order to improve the performance of the recommendation process. They defined a novel weighting function that considers sentiment, volume, and objectivity related to the users' interests.

In yet another approach, Osman et al. [30] presented an electronic product recommender system based on contextual information from sentiment analysis. Because ratings are usually insufficient and very limited, they constructed a contextual information sentiment model for a recommender system by making use of user comments and preferences. In a similar way, Contratre et al. [31] also proposed a recommender process that includes sentiment analysis of textual data extracted from Facebook and Twitter in order to increase conversion by matching product offers and consumer preferences. We can find similar combinations in other studies [32–34].

In addition, Rosa et al. [35] used a sentiment intensity metric to build a music recommender system. Users' sentiments are extracted from sentences posted on social networks and the recommendations are made using a framework of low complexity that suggests songs based on the current user's sentiment intensity. The research by Osman, Nurul Aida, and Shahrul [36] addressed the data-sparsity problem of recommender systems by integrating a sentiment-based analysis. Their work was applied to the Internet Movie Dataset (IMDb) and Movie Lens datasets, but improvements in sentiment analysis have been made since the paper was published. Rayan et al. [37] also tried to improve recommendations by addressing the data-sparsity problem. They proposed a smart recommender system based on methods of hybrid learning that integrate the most effective and efficient learning algorithms. These methods switch among content-based and collaborative filtering, identify the user context with the integration of dynamic filtering, and finally learn the profiles.

Several research teams [26,27,33,38–40] introduced the techniques for applying sentiment analysis in recommender systems. The techniques that are applicable for performing the analysis of sentiments include support vector machines (SVM), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and deep neural networks (DNN).

Recommender systems rely on explicit user ratings, but this is not feasible in an increasing number of domains. Moreover, when explicit ratings are available, the trust and reliability of the ratings may limit the recommender system. When we have a large number of reviews and comments on these items, analyzing sentiments in that text to obtain implicit feedback in addition to traditional ratings for items, is useful and helps to improve the recommendations to users. The above studies use sentiment analysis in recommendation methods, but most studies have used traditional sentiment techniques or a sole deep learning model.

In this study, we will apply new feature extraction techniques and hybrid deep-learning methods for sentiment analysis exploiting the advantages of BERT, in order to incorporate sentiments into recommendation methods as additional feedback and thus improve the performance and the reliability of recommender systems.

3. Methodology

In this section, the proposed recommender system is presented. It is based on a recommendation method that combines collaborative filtering and sentiment analysis. The aim is to improve reliability of the recommendations to the user by combining sentiment analysis of reviews or comments of users with traditional recommendation methods. The architecture of this system is illustrated in Figure 3. The architecture makes it easy to configure the modules and their interactions, allowing the application to be composed by choosing from supported techniques and methods.

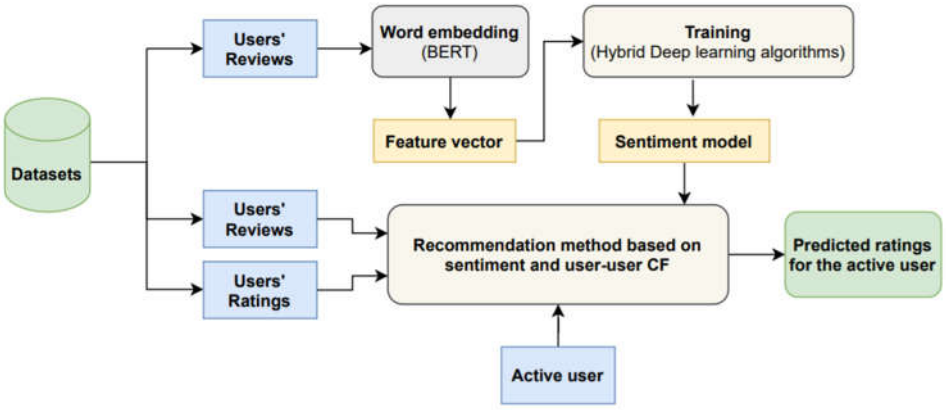


Figure 3. An architecture application in a recommender system.

The architecture has two separate parts, one part in charge of generating the sentiment models and the other part to provide recommendations to a given user making use of the models previously generated. The reviews' data were preprocessed and used to conduct and train sentiment-based hybrid deep-learning model. Then, a user-based (user-user) collaborative filtering method is combined with sentiment-based models for rating prediction.

3.1. Input Data and Preprocessing

Sentiment analysis requires that the text training data are cleaned before being used to induce the classification model. Text cleaning is a preprocessing step that removes words or other components that lack relevant information, and thus may reduce the effectiveness of sentiment analysis. After cleaning, the text data can be split into individual words, which are transformed into their base form by lemmatization, and then converted into numerical vectors by using methods such as word embedding or TF-IDF. Both word embedding and TF-IDF are used as input features of deep learning algorithms in nature language processing [41].

For the deep learning approaches, word embedding representations have performed significantly better than the TF-IDF representation of all features and feature selection algorithms [1,42]. In this research, we used BERT to transform text data to word embedding. Word embedding [43] is a type of word representation that maps each word into a vector of real values in such a way that words with similar meanings have a similar representation. Value learning can be done using neural networks. BERT is a language model for nature language processing, and it was published by researchers at Google AI Language in 2018 [44]. BERT was developed after Word2vec, and includes some advances over Word2vec, such as support for out-of-vocabulary (OOV) words.

3.2. Conduct and Train Sentiment-Based Hybrid Deep-Learning Models

We used the combination of several successful approaches. We start by using a pre-trained BERT model to create the feature vectors. We then vary the order of the CNN and LSTM models used in the next stages: BERT → CNN → LSTM or BERT → LSTM →

CNN. The final stage of the model uses a ReLu activation function. We labeled the reviews with one value of an ordinal scale of five classes (very negative; negative; neutral; positive; and very positive), analogous to the explicit ratings, to train and validate the result of sentiment analysis.

Figure 4 visualizes the process of the hybrid methodology for sentiment analysis. A pre-trained BERT model was used in our experiments as a feature extractor to generate input data for the proposal of hybrid models. The reviews data were fed into the BERT model to generate the feature vectors, which are then input to the hybrid models that perform the classification. The next step combines CNN and LSTM deep learning models, which are used because of their good performance on sentiment analysis [1], as well as to take advantage of the two network architectures when performing sentiment analysis on data in different domains. The final stage is classification. We use the activate function of Relu instead of Sigmoid because of the high convergence.

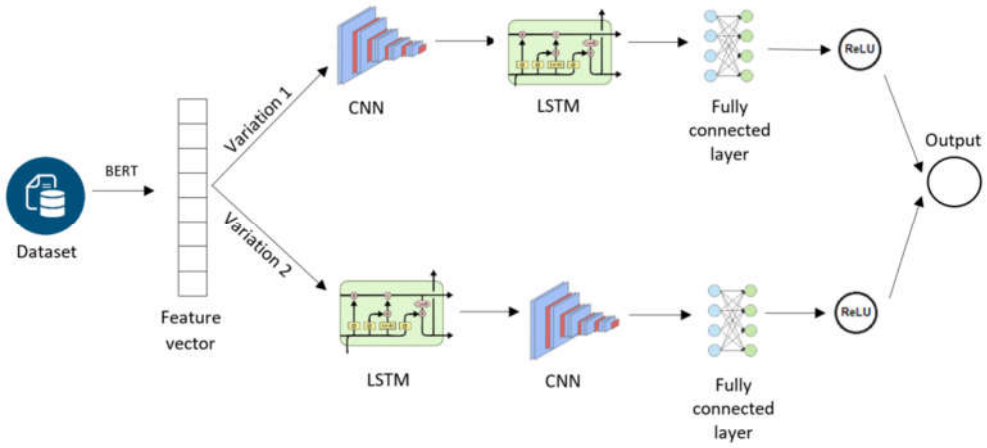


Figure 4. Process of hybrid methodology for sentiment analysis.

3.3. Proposed Recommendation Method

The proposed recommendation method is a user-based collaborative filtering approach that considers explicit ratings and sentiment analysis extracted from users' reviews. We tested Singular Value Decomposition (SVD), Non-Negative Matrix Factorization (NMF), and SVD++ (a derivative of SVD) as collaborative filtering methods. The objective is to achieve better predictive accuracy because of the addition of implicit feedback information provided by the sentiment.

Results from the CF recommendation method and sentiment analysis were combined to generate a rating and used to create a list of recommendations.

Given a rating matrix $R_{m \times n}$ $(\mathbb{N})$ for training, where m is the number of users and n is the number of items, $r_{ij} \in R_{m \times n}$ denotes the rating of user u_i on item i_j.

The rating of user u_a on item i_j in the test set is predicted as follows:

$$pr_{aj} = \beta * pr\_mf_{aj} + (1 - \beta) * pr\_sent_{aj} \quad (1)$$

where:

- pr_mf_{aj}: Rating for user u_a and item i_j predicted by Matrix Factorization methods (SVD, SVD++, and NMF) without using sentiments.
- pr_sent_{aj}: Rating for user u_a and item i_j predicted by using the sentiment model.
- β parameter used to adjust the importance of each term of the equation.

As shown in Algorithm 1, we used pseudocode to describe how to compute pr_sent_{aj}. As mentioned above, hybrid sentiment models are used for classifying each "review" in one of five possible classes. These classes are converted into sentiment scores from 1 to 5 analogous to ratings. First, for each user u_a, we find all items that user u_a already rated and

the sentiment score of the corresponding review matches the explicit rating. And second, for each item i_j, we also find all users who already rated item i_j and item i_k (found in the first step) in the training set and their review scores also match the explicit ratings.

Algorithm 1. Rating prediction based on sentiment for user u_a and item i_j.

```
Function sentiment_ratingPred (user u_a, item i_j) {
//This function is used to obtain the pr_sent_aj term of Equation (1)
  //Step 1:
  FOR each item i_k in the training set:
    IF user u_a already rated item i_k AND review score matches rating THEN
      Add i_k to list of items I;
  //The result of this step is a set of m items I = {i_1, i_2, ... i_m}
  //Step 2:
  FOR each user u_b in the training set:
    FOR each item i_k in the set of items I:
      IF user u_b already rated item i_j AND user u_b already rated item i_k
AND their review scores match ratings
        Add user u_b to list of users U;
  //The results of this step is a set of n users U = {u_1, u_2, ... u_n}
  //Step 3:
  IF length(U)>0 THEN
    FOR each user u_i in the set of user U:
      Compute s_a,i= sim (user u_a, user u_i) by applying cosine metric;
      Add s_a,i to S;
    //The result is a set of n similarity values S = {s_a,i}
    Set the K value to select the K nearest neighbors using S;
    Compute the predicted rating pr_aj by applying the Equation (2);
     pr_sent_aj = pr_aj
    Return pr_sent_aj;
  ELSE
    Return 0;
}
```

Next, two lists of data, including items and users which are created from step 1 and step 2, are used for predicting user u_a rating on each item i_j. To do that, we compute the similarity between users by applying the cosine metric. Then, we apply Equation (2) for rating prediction based on user similarity. The ratings of the k most similar users are used to estimate the preferences of the active user u_a about the item i_j that he/she has not rated.

$$pr_{aj} = \overline{r_a} + \frac{\sum_{i=1}^{K} Sim(u_a, u_i)\left(r_{ij} - \overline{r_i}\right)}{\sum_{i=1}^{K}\left|Sim(u_a, u_i)\right|} \tag{2}$$

where r_{ij} is the rating that user u_i gives to item i_j respectively; $\overline{r_a}$ and $\overline{r_i}$ are the average ratings of user u_a and user u_i, respectively; and $Sim(u_a, u_i)$ is the similarity between the active user u_a and his neighbor user u_i, which would be obtained by using the cosine metric (Equation (3)). In our case, the neighbors of user u_a are users who have rated the same items as user u_a in a similar way or the score of their reviews on the same items are similar.

$$Sim(u_a, u_i) = \frac{\sum_{j=1}^{n} r_{aj}\, r_{ij}}{\sqrt{\sum_{j=1}^{n} r_{aj}^2}\sqrt{\sum_{j=1}^{n} r_{ij}^2}} \tag{3}$$

4. Experiments and Results

In this section, we present the experiments conducted to evaluate the performance of the proposed approach to recommender systems. In particular, we used two well-known datasets, Amazon Fine Food Reviews and Amazon Movie Reviews, in order to validate the

proposal. The results are shown and discussed in Section 4.2. The metrics used to evaluate the reliability of rating predictions were Root-Mean-Square Error (RMSE), Mean Absolute Error (MAE) and Normal MAE (NMAE). In addition, Mean Reciprocal Rank (MRR), Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG) were used for evaluating top-N recommendations. Accuracy, Area Under Curve (AUC), and F-score were the metrics used to evaluate the performance of the two hybrid deep-learning models for sentiment analysis through all experiments. Because the F-score is the average of the F-score of each class, with weighting depending on the average parameter in the multi-class and multi-label case, we used the average parameter with 'weighted' value to calculate the metrics for each label and to find their average weighted by support.

The configuration of related parameters, hardware devices, and the necessary library facilities was carried out before performing the experiments, such as echo = 5, and k-fold = 5. In particular, we used Google Colab Pro with GPU Tesla P100-PCIE-16GB or GPU Tesla V100-SXM2-16GB [45], Keras [46] and Tensorflow [47] libraries. We also used the implementation of the SVD, NMF, and SVD++ algorithms provided by the Surprise library (http://surpriselib.com/, accessed on 10 December 2020).

4.1. Dataset

We chose the datasets based on availability and accessibility criteria. Moreover, we considered that they are widely accepted by the research community. These datasets are shown in Table 1 and described below:

- Amazon Fine Foods Reviews comprise reviews of fine foods from Amazon [48]. Each review includes product and user information, as well as the rating, and the plaintext review given by each user to each product he/she rated. The data span a period of more than 10 years, including 568,454 reviews with 256,059 users and 74,258 products up to October 2012.
- Amazon Movie Reviews consists of movie reviews from Amazon [48]. Each review also includes product and user information, ratings, and plaintext reviews. It covers a period of more than 10 years as well, including 7,911,684 reviews with 889,176 users and 253,059 products up to October 2012.

Table 1. Statistics of the datasets.

#	Amazon Fine Foods Reviews	Amazon Movie Reviews
Number of reviews	568,454	7,911,684
Number of users	256,059	889,176
Number of products	74,258	253,059
Users with > 50 reviews	260	16,341
Average no. of words per review	56	101
Timespan	October 1999–October 2012	August 1997–October 2012

4.2. Results and Discussion

We performed experiments with two different settings without/with sentiment analysis. In the former, recommendations are based on recommender system methods without sentiment while in the second, the result of performing sentiment analysis on the reviews is incorporated to the recommendation process. We tested two hybrid deep-learning models for sentiment analysis: CNN and LSTM as well as LSTM and CNN, referred to as C-LSTM, L-CNN, respectively.

As presented in Figure 4, we adopt a pre-trained BERT model to vectorize each plaintext review. The obtained vector is then fed into C-LSTM or L-CNN followed by the fully connected layer. Finally, ReLU is stacked on the top of the classifier. The output of the sentiment classifier is exploited for recommendation. Table 2 and Figure 5 present the experimental results of sentiment classification. The results show that the performances of the hybrid models are encouraging, with accuracy and F-score over 80% and AUC over

84%, respectively. These models will be applied to predict sentiment rating before being combined with recommendation methods.

Table 2. Sentiment performance of hybrid deep-learning models.

Measures	Amazon Fine Foods Reviews		Amazon Movie Reviews	
	L-CNN	C-LSTM	L-CNN	C-LSTM
Accuracy	80.04%	79.95%	82.27%	82.27%
F-Score	80.24%	80.00%	82.49%	82.46%
AUC	84.22%	84.36%	86.07%	86.17%

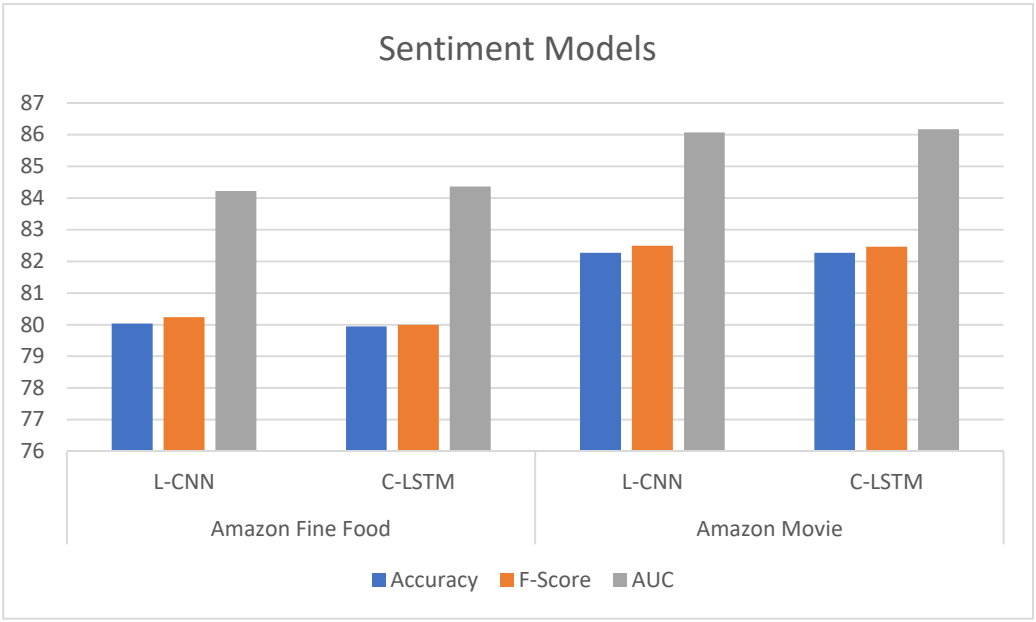


Figure 5. Sentiment performance of hybrid deep-learning models using BERT.

To validate our recommendation approach, we compared the performance of three widely used CF recommendation methods in their traditional form as baseline and the same methods improved with our proposal involving use of sentiment analysis of reviews. The comparative study was conducted for both rating prediction and item recommendation (recommendation of top-N lists).

Tables 3–5 show the results of MAE, RMSE and NMAE measures for rating prediction on both dataset food and movie reviews. They were calculated based on SVD, NMF and SVD++ algorithm with and without using sentiment analysis. Beta (β) parameter is used to adjust the importance of the recommendation result without and with sentiment in the Equation (1). Figures 6–8 illustrate the comparative results obtained from the recommender with sentiment analysis on different values of the β parameter against those obtained from the recommender without sentiment analysis.

Table 3. MAE values without and with L-CNN sentiment analysis model.

#	Amazon Fine Foods Reviews			Amazon Movie Reviews		
	SVD	NMF	SVD++	SVD	NMF	SVD++
Without sentiment	0.9706	0.9608	0.9540	0.8644	0.8087	0.8266
With sentiment (β = 0.3)	0.8365	0.8762	0.8263	0.5943	0.5936	0.5770
With sentiment (β = 0.5)	0.8634	0.8846	0.8470	0.6268	0.5976	0.5959
With sentiment (β = 0.7)	0.8933	0.8964	0.8707	0.6701	0.6125	0.6253

Table 4. RMSE values without and with L-CNN sentiment analysis model.

#	Amazon Fine Foods Reviews			Amazon Movie Reviews		
	SVD	**NMF**	**SVD++**	**SVD**	**NMF**	**SVD++**
Without sentiment	1.2076	1.2312	1.1831	0.9960	0.9464	0.9376
With sentiment (β = 0.3)	1.1338	1.2103	1.1292	0.8732	0.9112	0.8577
With sentiment (β = 0.5)	1.1442	1.2102	1.1356	0.8851	0.9041	0.8598
With sentiment (β = 0.7)	1.1633	1.2150	1.1493	0.9166	0.9110	0.8791

Table 5. NMAE values without and with L-CNN sentiment analysis model.

#	Amazon Fine Foods Reviews			Amazon Movie Reviews		
	SVD	**NMF**	**SVD++**	**SVD**	**NMF**	**SVD++**
Without sentiment	0.2427	0.2402	0.2385	0.2161	0.2022	0.2066
With sentiment (β = 0.3)	0.2091	0.2191	0.2066	0.1486	0.1484	0.1443
With sentiment (β = 0.5)	0.2158	0.2211	0.2117	0.1567	0.1494	0.1490
With sentiment (β = 0.7)	0.2233	0.2241	0.2177	0.1675	0.1531	0.1563

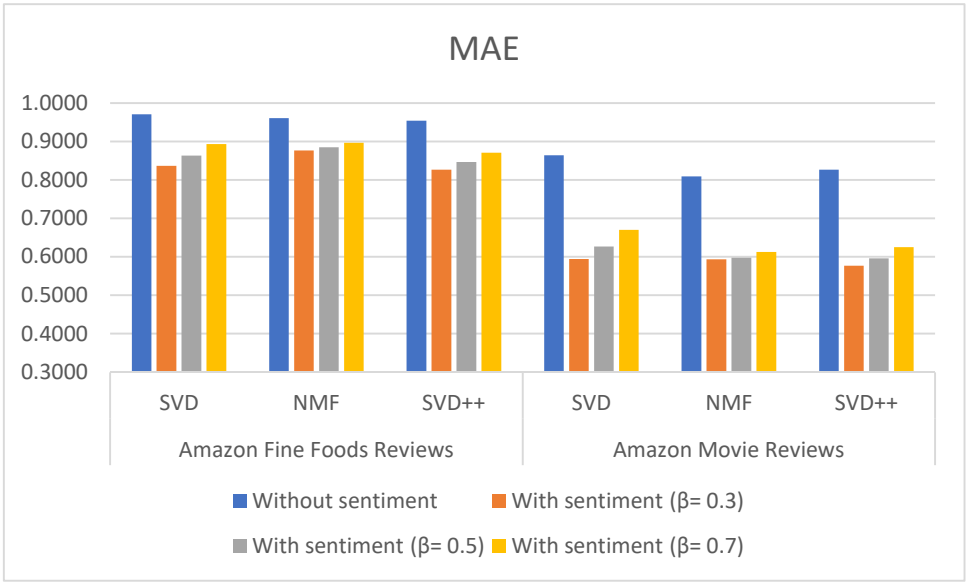


Figure 6. MAE measures comparison for different types of method and datasets using L-CNN sentiment model.

The results show that RSME, MAE, and NMAE yielded by the approach that combines CF with sentiment analysis are better than the error rates yielded by traditional CF methods without sentiment on all algorithm in all β values. We found that the best results of the proposal are obtained with β = 0.3.

Regarding the type of datasets, Amazon Movie Reviews provided better results than those of Amazon Fine Foods reviews. For example, MAE measured with SVD++ is 0.577 with β = 0.3; RMSE measured with SVD++ is 0.8577 with β = 0.3; and NMAE measured with SVD++ is 0.1443 with β = 0.3.

Figures 9 and 10 illustrate the comparison of the sentiment-based methods with the L-CNN and the C-LCTM against non-sentiment-based methods on Amazon Fine Foods Reviews and Amazon Movie Reviews. The values with sentiment are obtained with β = 0.3. We found that C-LSTM and L-CNN provide similar results. In addition, the sentiment-based approach provides better results on Amazon Movie reviews.

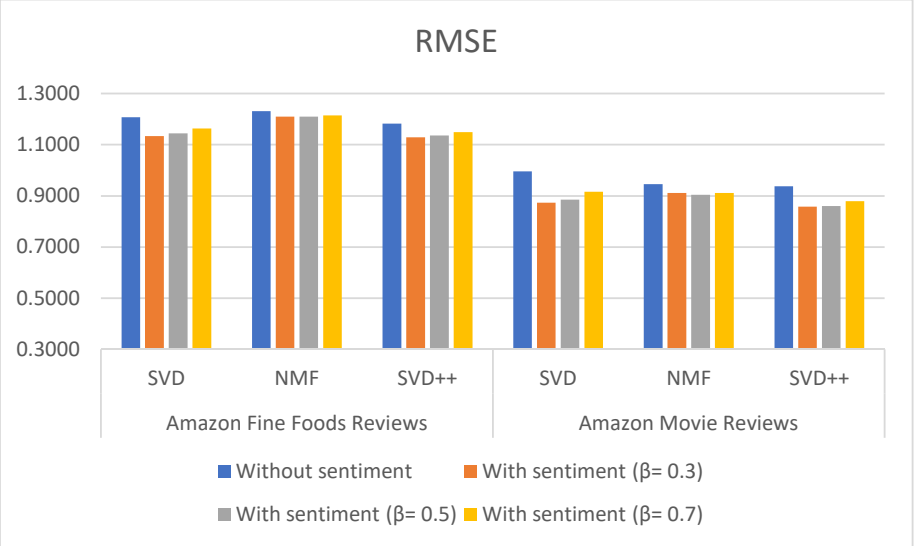


Figure 7. RMSE measures the comparison for different types of methods and datasets using L-CNN sentiment model.

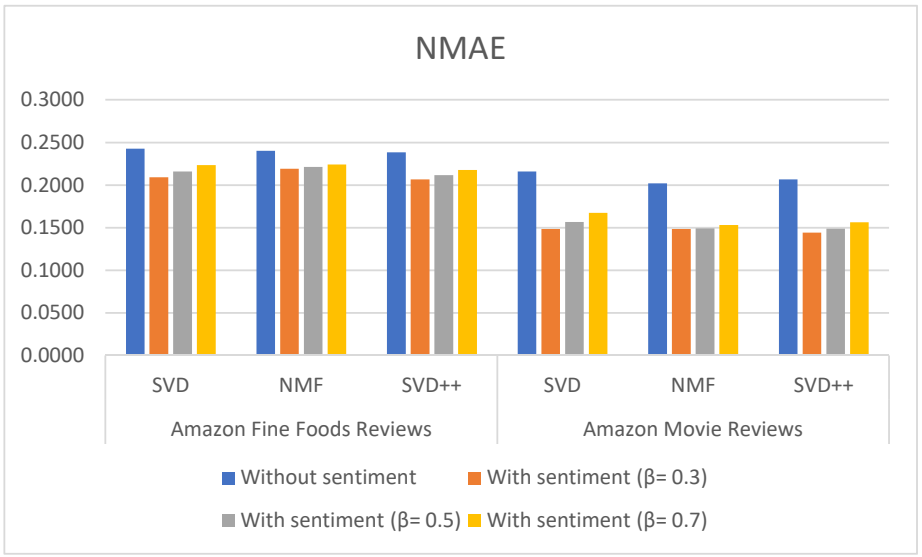
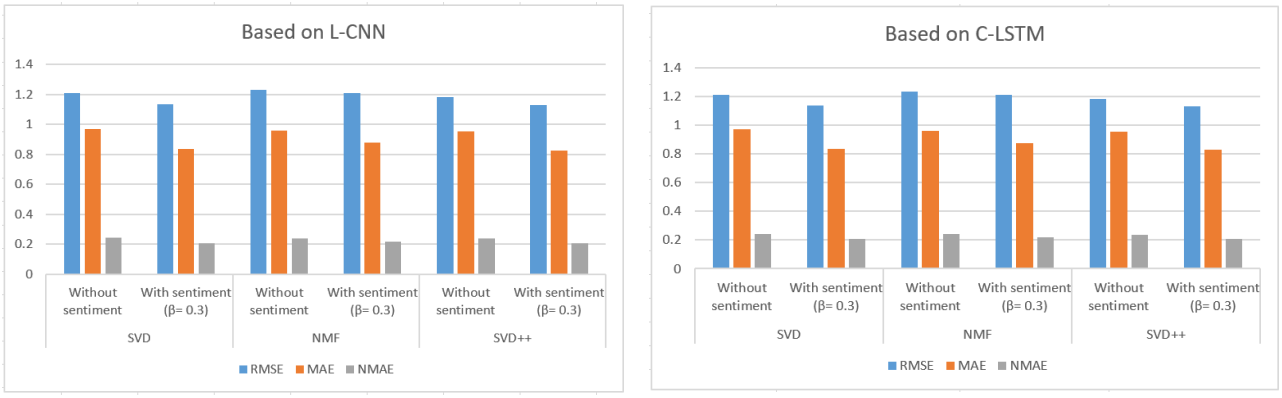


Figure 8. NMAE measures the comparison for different types of methods and datasets using L-CNN sentiment model.

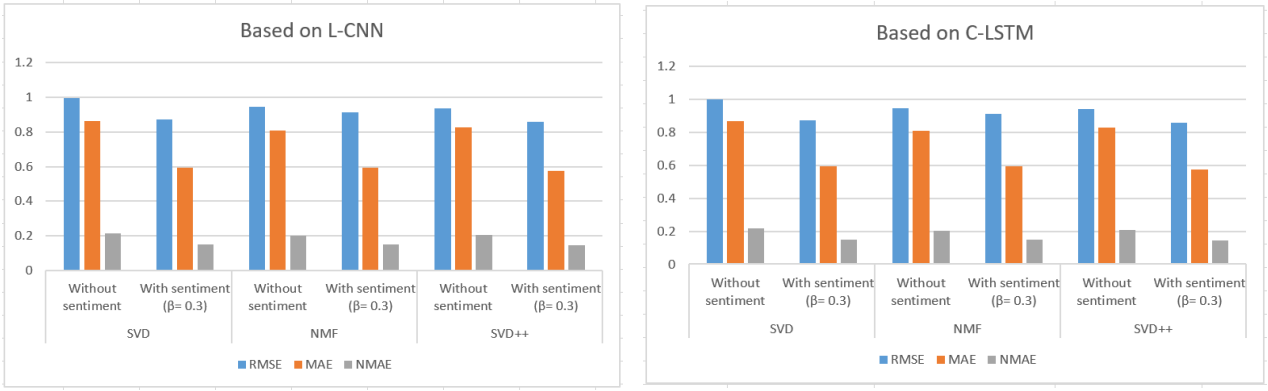
For all algorithms applied to two datasets, the combined proposal provides lower error rates. For example, the sentiment-based approach on Amazon Movie Reviews with β = 0.3 when L-CNN is used for sentiment analysis provided the following percentage improvement with SVD: 12.29% in RMSE; 27.01% in MAE; and 6.75% in NMAE. With the above results, we see that the sentiment model helps to improve the predicted ratings. Instead of just using explicit rating, the predictive model now considers the aspect of analyzing reviews of related items and users. Because more information is available in the new recommendation method, we get better than usual results.

In addition to proving that the proposed method performs better in predicting ratings, we also checked the performance for top-N recommendations. MRR, MAP, and NDCG rank-based metrics have been computed. The results obtained for N = 5 are given in Tables 6 and 7, and in Figures 11 and 12. SVD, NMF, and SVD++ values, respectively, with L-CNN and C-LSTM sentiment models are obtained when applied on β = 0.7.



(**a**) (**b**)

Figure 9. Comparison of the sentiment-based methods with the L-CNN model (**a**) and the C-LSTM (**b**) and $\beta = 0.3$ against non-sentiment-based methods on Amazon Fine Foods Reviews.



(**a**) (**b**)

Figure 10. Comparison of the sentiment-based methods with the L-CNN model (**a**) and the C-LSTM (**b**) and $\beta = 0.3$ against non-sentiment-based methods on Amazon Movie Reviews.

Table 6. MRR, MAP and NDCG values without and with L-CNN sentiment model on Amazon Fine Foods Reviews with $\beta = 0.7$.

#	SVD		NMF		SVD++	
	Without Sentiment	**With Sentiment ($\beta = 0.7$)**	**Without Sentiment**	**With Sentiment ($\beta = 0.7$)**	**Without Sentiment**	**With Sentiment ($\beta = 0.7$)**
MRR	83.92%	84.09%	82.98%	83.19%	84.24%	84.24%
MAP	73.06%	73.67%	72.97%	73.26%	73.48%	73.82%
NDCG	86.53%	86.78%	86.67%	86.84%	86.84%	86.89%

Table 7. MRR, MAP, and NDCG values without and with C-LSTM sentiment model on Amazon Fine Foods Reviews with $\beta = 0.7$.

#	SVD		NMF		SVD++	
	Without Sentiment	**With Sentiment ($\beta = 0.7$)**	**Without Sentiment**	**With Sentiment ($\beta = 0.7$)**	**Without Sentiment**	**With Sentiment ($\beta = 0.7$)**
MRR	83.92%	84.16%	82.98%	83.23%	84.24%	84.33%
MAP	73.06%	73.64%	72.97%	73.23%	73.48%	73.84%
NDCG	86.53%	86.78%	86.67%	86.82%	86.84%	86.89%

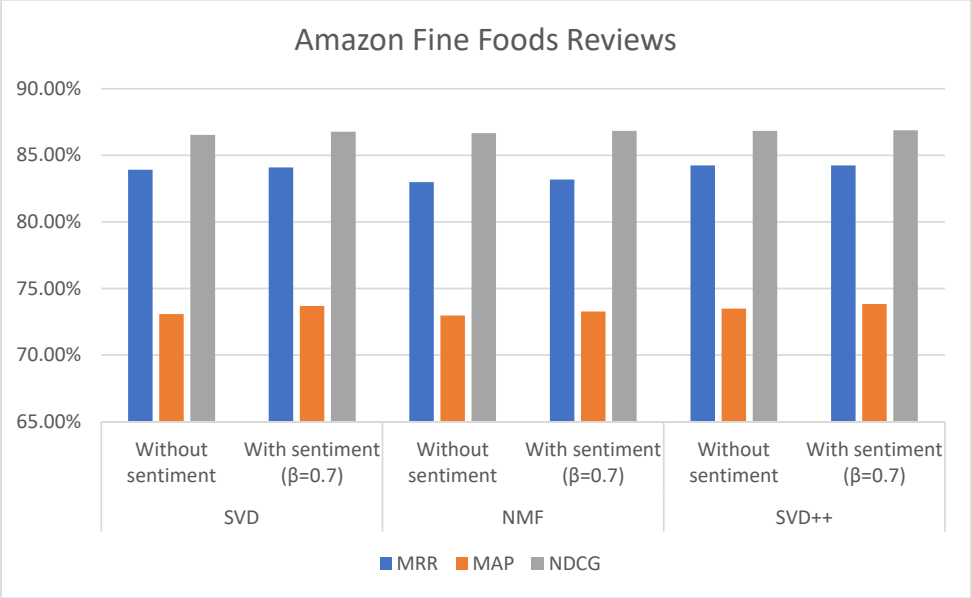


Figure 11. MRR, MAP, and NDCG values without and with L-CNN sentiment model on Amazon Fine Foods Reviews with $\beta = 0.7$.

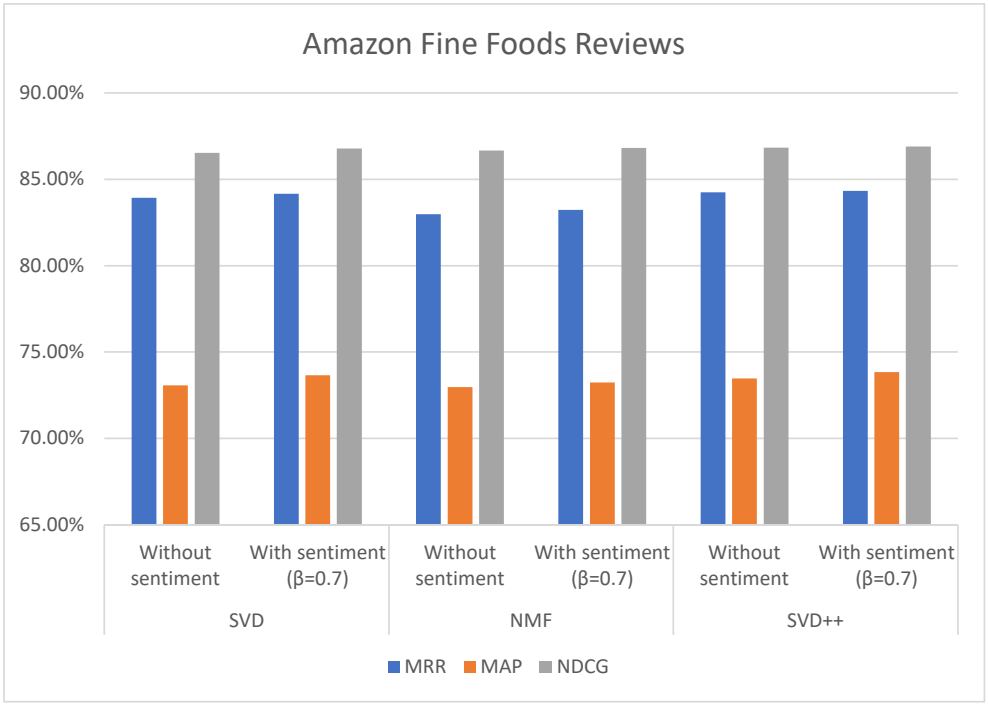


Figure 12. MRR, MAP, and NDCG values without and with C-LSTM sentiment model on Amazon Fine Foods Reviews with $\beta = 0.7$.

The values of MRR, MAP, and NDCG show that the proposed method also improve top-N recommendations. In the case of the Amazon Foods Review with $\beta = 0.7$ and C-LSTM sentiment model, the increase in MAP was 0.58% (SVD), 0.26% (NMF), and 0.36% (SVD++) over without sentiment. Regarding NDCG, the increase was 0.25% (SVD), 0.15% (NMF), and 0.06% (SVD++) over without sentiment. The value of MRR was increased on 0.24% (SVD), 0.25% (NMF), and 0.09% (SVD++).

Three algorithms (SVD, NMF, and SVD++) were tested in two ways, with explicit ratings only, and combining explicit ratings with sentiment extracted from reviews. In

most cases, the combined approach with sentiments from two classification models (C-LSTM and L-CNN) on food and movie reviews datasets gave better results. However, the improvement for top-N recommendation is not as significant than the achieved for rating prediction.

In general, the sentiment-based methods proposed in this work provide better results than those based only on explicit ratings. These improvements have occurred in both data sets used in the study. General summaries of the results achieved in the experiments referenced earlier are discussed below:

- We presented and evaluated a recommendation approach that integrates sentiment analysis and collaborative filtering methods.
- Two datasets, Amazon Fine Foods Review and Amazon Movie Review, are used for evaluation. Each plaintext review is vectorized by using the pre-trained BERT model.
- Two hybrid sentiment classification models, CNN-LSTM and LSTM-CNN, are used for extracting sentiments from reviews, which are incorporated as implicit feedback into the recommender system models.
- We applied SVD, NMF, and SVD++ recommendation methods following the user-based CF approach.
- Accuracy, F-score, and AUC were computed for validating the sentiment classification models.
- The evaluation of the recommendation method was performed for rating prediction and top-N recommendation. RMSE, MAE, and NMAE were the metrics used in the first case, and MRR, MAP and NDCG were the metrics used in the second case.
- The sentiment-based proposal increased the recommendation reliability in comparison to traditional, rating-based recommendation methods on the two datasets.

5. Conclusions

In this paper, we have proposed an application of sentiment analysis in recommender systems that is based on hybrid deep-learning models and collaborative filtering on online social networks. The system architecture presented in this work, can integrate a variety of techniques that have been proposed to perform recommendations, including the pre-processing strategy, hybrid deep-learning models for sentiment analysis and methods for recommender systems. The architecture can be used to develop a recommender system in the context of social networks that take advantage of sentiment analysis performed on user opinions and reviews in the network. We conducted experiments with reviews of food and movies. Based on such experiments, we demonstrate the utility and applicability of our approaches in producing personalized recommendations on online social networks.

The results show that the joint use of deep learning-based sentiment analysis and collaborative filtering methods significantly improves the performance last ones. This is achieved through the exploitation of additional information from user reviews/comments data. Its integration into the traditional recommendation methods makes the recommender system more reliable and capable of providing better recommendations to users.

As a future work, we plan to explore other application domains to ensure that the proposed architecture can be generalized to efficiently solve similar problems. We will also consider researching new sentiment analysis techniques, such as graph convolutional networks, for a potential improvement of this aspect.

Author Contributions: Conceptualization, C.N.D. and M.N.M.-G.; methodology, M.N.M.-G. and F.D.l.P.; software, C.N.D.; validation, M.N.M.-G. and F.D.l.P.; formal analysis, C.N.D., and M.N.M.-G.; investigation, C.N.D.; data curation, C.N.D.; writing—original draft preparation, C.N.D.; writing—review and editing, M.N.M.-G. and F.D.l.P.; visualization, C.N.D.; supervision, M.N.M.-G. and F.D.l.P.; project administration, M.N.M.-G.; funding acquisition, M.N.M.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by the Junta de Castilla y León, Spain, grant number SA064G19.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Dang, N.C.; Moreno-García, M.N.; De la Prieta, F. Sentiment analysis based on deep learning: A comparative study. *Electronics* **2020**, *9*, 483. [CrossRef]
2. Preethi, G.; Krishna, P.V.; Obaidat, M.S.; Saritha, V.; Yenduri, S. Application of Deep Learning to Sentiment Analysis for Recommender System on Cloud. In Proceedings of the 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), Dalian, China, 21–23 July 2017; IEEE: New York, NY, USA, 2017; pp. 93–97.
3. Keenan, M.J.S. *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets: Bridging Fundamental and Technical Analysis*, 2nd ed.; Wiley: Chichester, UK, 2018.
4. Sánchez-Moreno, D.; Moreno-García, M.N.; Mobasher, B.; Sonboli, N.; Burke, R. Using Social Tag Embedding in a Collaborative Filtering Approach for Recommender Systems. In Proceedings of the 2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, Melbourne, Australia, 14–17 December 2020; IEEE: New York, NY, USA, 2021.
5. Sánchez-Moreno, D.; Batista, V.F.L.; Vicente, M.D.M.; González, A.B.G.; Moreno-García, M.N. A session-based song recommendation approach involving user characterization along the play power-law distribution. *Complexity* **2020**, *2020*, 1–13. [CrossRef]
6. Bhavitha, B.; Rodrigues, A.P.; Chiplunkar, N.N. Comparative Study of Machine Learning Techniques in Sentimental Analysis. In Proceedings of the 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 10–11 March 2017; IEEE: New York, NY, USA, 2017; pp. 216–221.
7. Salas-Zárate, M.D.P.; Medina-Moreira, J.; Lagos-Ortiz, K.; Luna-Aveiga, H.; Rodriguez-Garcia, M.A.; Valencia-García, R. Sentiment analysis on tweets about diabetes: An aspect-level approach. *Comput. Math. methods Med.* **2017**, *2017*, 1–9. [CrossRef] [PubMed]
8. Zhang, X.; Zheng, X. Comparison of Text Sentiment Analysis Based on Machine Learning. In Proceedings of the 2016 15th International Symposium on Parallel and Distributed Computing (ISPDC), Fuzhou, China, 8–10 July 2016; IEEE: New York, NY, USA, 2016; pp. 230–233.
9. Pandey, A.C.; Rajpoot, D.S.; Saraswat, M. Twitter sentiment analysis using hybrid cuckoo search method. *Inf. Process. Manag.* **2017**, *53*, 764–779. [CrossRef]
10. Xue, D.-X.; Zhang, R.; Feng, H.; Wang, Y.-L. CNN-SVM for microvascular morphological type recognition with data augmentation. *J. Med Biol. Eng.* **2016**, *36*, 755–764. [CrossRef]
11. Elleuch, M.; Maalej, R.; Kherallah, M. A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition. *Procedia Comput. Sci.* **2016**, *80*, 1712–1723. [CrossRef]
12. Tang, Y. Deep learning using linear support vector machines. *arXiv* **2013**, arXiv:preprint/1306.0239.
13. Chen, T.; Xu, R.; He, Y.; Wang, X. Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Syst. Appl.* **2017**, *72*, 221–230. [CrossRef]
14. Rehman, A.U.; Malik, A.K.; Raza, B.; Ali, W. A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis. *Multimed. Tools Appl.* **2019**, *78*, 26597–26613. [CrossRef]
15. Vo, Q.-H.; Nguyen, H.-T.; Le, B.; Nguyen, M.-L. Multi-Channel LSTM-CNN Model for Vietnamese Sentiment Analysis. In Proceedings of the 2017 9th International Conference on Knowledge and Systems Engineering (KSE), Hue, Vietnam, 19–21 October 2017; IEEE: New York, NY, USA, 2017; pp. 24–29.
16. Martín, C.A.; Torres, J.M.; Aguilar, R.M.; Diaz, S. Using deep learning to predict sentiments: Case study in tourism. *Complexity* **2018**, *2018*, 1–9. [CrossRef]
17. Hochreiter, S.; Schmidhuber, J. LSTM can solve hard long time lag problems. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, MA, USA, 3 December 1996; pp. 473–479.
18. Yamashita, R.; Nishio, M.; Do, R.K.G.; Togashi, K. Convolutional neural networks: An overview and application in radiology. *Insights Imaging* **2018**, *9*, 611–629. [CrossRef] [PubMed]
19. Medhat, W.; Hassan, A.; Korashy, H. Sentiment analysis algorithms and applications: A survey. *Ain Shams Eng. J.* **2014**, *5*, 1093–1113. [CrossRef]
20. Lu, J.; Wu, D.; Mao, M.; Wang, W.; Zhang, G. Recommender system application developments: A survey. *Decis. Support Syst.* **2015**, *74*, 12–32. [CrossRef]
21. Betru, B.T.; Onana, C.A.; Batchakui, B. A Survey of State-of-the-art: Deep Learning Methods on Recommender System. *Int. J. Comput. Appl.* **2017**, *162*, 17–22.
22. Kardan, A.A.; Ebrahimi, M. A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. *Inf. Sci.* **2013**, *219*, 93–110. [CrossRef]
23. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [CrossRef]

24. Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S. Collaborative filtering recommender systems. In *The Adaptive Web*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 291–324.
25. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv.* **2019**, *52*, 1–38. [CrossRef]
26. Wang, Y.; Wang, M.; Xu, W. A sentiment-enhanced hybrid recommender system for movie recommendation: A big data analytics framework. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 1–9. [CrossRef]
27. Kumar, S.; De, K.; Roy, P.P. Movie recommendation system using sentiment analysis from microblogging data. *IEEE Trans. Comput. Soc. Syst.* **2020**, *7*, 915–923. [CrossRef]
28. Rao, K.Y.; Murthy, G.; Adinarayana, S. Product recommendation system from users reviews using sentiment analysis. *Int. J. Comput. Appl.* **2017**, *975*, 8887.
29. Gurini, D.F.; Gasparetti, F.; Micarelli, A.; Sansonetti, G. A Sentiment-Based Approach to Twitter User Recommendation. *RSWeb@ RecSys* **2013**, *1066*, 1–4.
30. Osman, N.; Noah, S.; Darwich, M. Contextual sentiment based recommender system to provide recommendation in the electronic products domain. *Int. J. Mach. Learn. Comput.* **2019**, *9*, 425–431. [CrossRef]
31. Contratres, F.G.; Alves-Souza, S.N.; Filgueiras, L.V.L.; DeSouza, L.S. Sentiment Analysis of Social Network Data for Cold-Start Relief in Recommender Systems. In Proceedings of the World Conference on Information Systems and Technologies, Naples, Italy, 27–29 March 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 122–132.
32. Nabil, S.; Elbouhdidi, J.; Yassin, M. Recommendation System Based on Data Analysis-Application on Tweets Sentiment Analysis. In Proceedings of the 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), Marrakech, Morocco, 21–27 October 2018; IEEE: Marrakech, Morocco, 2018; pp. 155–160.
33. Ziani, A.; Azizi, N.; Schwab, D.; Aldwairi, M.; Chekkai, N.; Zenakhra, D.; Cheriguene, S. Recommender system through sentiment analysis. In Proceedings of the 2nd International Conference on Automatic Control, Telecommunications and Signals, Annaba, Algeria, 11–12 December 2017.
34. Abbasi, F.; Khadivar, A.; Yazdinejad, M. A Grouping Hotel Recommender System Based on Deep Learning and Sentiment Analysis. *J. Inf. Technol. Manag.* **2019**, *11*, 59–78.
35. Rosa, R.L.; Rodriguez, D.Z.; Bressan, G. Music recommendation system based on user's sentiments extracted from social networks. *IEEE Trans. Consum. Electron.* **2015**, *61*, 359–367. [CrossRef]
36. Osman, N.A.; Noah, S.A.M. Sentiment-Based Model for Recommender Systems. In Proceedings of the 2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP), Kota Kinabalu, Malaysia, 26–28 March 2018; IEEE: Kota Kinabalu, Malaysia, 2018; pp. 1–6.
37. Nouh, R.M.; Lee, H.-H.; Lee, W.-J.; Lee, J.-D. A smart recommender based on hybrid learning methods for personal well-being services. *Sensors* **2019**, *19*, 431. [CrossRef]
38. Devipriya, K.; Prabha, D.; Pirya, V.; Sudhakar, S. Deep learning sentiment analysis for recommendations in social applications. *Int. J. Sci. Technol. Res.* **2020**, *9*, 3812–3815.
39. Singh, V.K.; Mukherjee, M.; Mehta, G.K. Combining Collaborative Filtering and Sentiment Classification for Improved Movie Recommendations. In Proceedings of the International Workshop on Multi-disciplinary Trends in Artificial Intelligence, Hyderabad, India, 7–9 December 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 38–50.
40. Nimirthi, P.; Krishna, P.V.; Obaidat, M.S.; Saritha, V. A framework for sentiment analysis based recommender system for agriculture using deep learning approach. In *Social Network Forensics, Cyber Security, and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 59–66.
41. Dessi, D.; Helaoui, R.; Kumar, V.; Recupero, D.R.; Riboni, D. Tf-IDF vs word embeddings for morbidity identification in clinical notes: An initial study. *arXiv* **2021**, arXiv:preprint/09632.
42. Kumar, V.; Recupero, D.R.; Riboni, D.; Helaoui, R. Ensembling classical machine learning and deep learning approaches for morbidity identification from clinical notes. *IEEE Access* **2020**, *9*, 7107–7126. [CrossRef]
43. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
44. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:preprint/04805.
45. Making the Most of Your Colab Subscription. Available online: https://colab.research.google.com/notebooks/pro.ipynb (accessed on 22 January 2021).
46. Keras: The Python Deep Learning API. Available online: https://keras.io/ (accessed on 10 December 2020).
47. TensorFlow. Available online: https://www.tensorflow.org/ (accessed on 10 December 2020).
48. McAuley, J.J.; Leskovec, J. From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; pp. 897–908.

Framework for Retrieving Relevant Contents Related to Fashion from Online Social Network Data

**Nhan Cach Dang, Fernando De la Prieta,
Juan Manuel Corchado and María N. Moreno**

Abstract Nowadays, online social networks such as Facebook and Twitter become increasingly popular. These social media channels allow people to create, share, and comment on information about anything related to their real-life. Such information is very useful for various application domains, e.g., decision support systems or online advertising.

In this paper, we propose a comprehensive framework for retrieving relevant contents from online social network data. Our approach is proposed on the basic of the Vector Space Model and Support Vector Machine to process and classify raw text data. Our experiments demonstrate the utility and accuracy of the framework in retrieving fashion related contents from Twitter and Facebook.

Keywords Text mining · TF-IDF · Vector Space Model · Support Vector Machine

1 Introduction

Fashion, especially to young people, is more and more interesting and widely shared in the form of user-generated contents on social media channels. Such data contain opinions of users, about some topics, fashion trends for the next season, such as fashion events and so on. Automatically collecting and analyzing that

N.C. Dang(✉)
HoChiMinh City University of Transport (UT-HCMC), Ho Chi Minh City, Vietnam
e-mail: tucach@hcmutrans.edu.vn
http://www.hcmutrans.edu.vn/en/

F. De la Prieta · J.M. Corchado · M.N. Moreno
University of Salamanca, Salamanca, Spain
e-mail: {fer,corchado,mmg}@usal.es
http://www.usal.es/

F. de la Prieta et al. (eds.), *Trends in Pract. Appl. of Scalable Multi-Agent Syst., the PAAMS Collection*, Advances in Intelligent Systems and Computing 473,
DOI: 10.1007/978-3-319-40159-1_28

information is very helpful for recommender system, online advertising or for improving the quality of services and refining fashion designs. Typically, such information is obtained and analyzed manually through survey. This type of method exposes disadvantages such as high cost, low accuracy and especially not real-time update.

This paper presents a comprehensive framework for the purpose of automatically retrieving contents for a specific topic. Our approach is proposed on the basis of the Vector Space Model and Support Vector Machine (SVM) [1, 2] to process and classify raw text data. Although SVM has been studied for problems of text classification recently [3], applying it to retrieving and analyzing relevant content related to fashion from online social network data is still not much focused. Our experiments on two popular social networks, Facebook and Twitter, demonstrate the utility and accuracy of the framework in the extracted information on the fashion.

The rest of the paper is organized as follows. Section 2 introduces a background and related work of this trend. Section 3, presents the Framework for retrieving useful information related to fashion from online social network data. Section 4 provides some datasets for experiment this framework. Section 5 summarizes experiments and results, following by the conclusion in section 6.

2 Background and Related Work

2.1 Vector Space Model

Vector space model or term vector model [4] is an algebraic model for representing text documents as vectors of identifiers. The elements of this vector expresses the relevance ranking of words or some word frequency function as the appearance or absence of each word in the document.

This model presents text documents as points in n-dimensional Euclidean space. Each unique term in the document collection corresponds to a dimension in the space. Documents are viewed as points in a hyperspace whose axes are the terms used in the document vectors. The location of a document in the space is determined by the degree to which the terms are presented in a document. The similarity between two documents is defined as the distance between the points corresponding to them or the angle of the vectors. Show in Fig. 1.

The measure TF-IDF (Term Frequency - Inverse Document Frequency) is often used because of its effectiveness in the field of text mining. This is a common method to evaluate and rank the importance of a word in a document. Details of this measure are shown in the following sections.

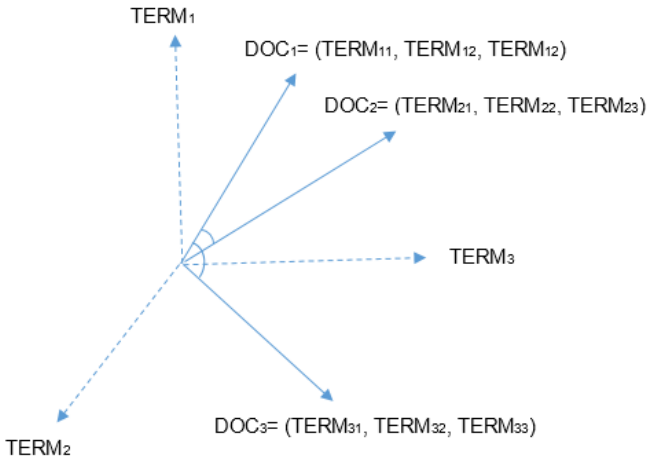


Fig. 1 Presentation vector space model.

2.2 TF-IDF Measure

TF-IDF, the short term of frequency–inverse document frequency, is a statistical measure reflecting how important a word is to a document in a collection or corpus [5]. Just like the name, TF–IDF is the product of two statistics, Term Frequency (TF) and Inverse Document Frequency (IDF).

TF (Term Frequency) – the number of times the words appears in the Document. It is measured which raw frequency divided by the maximum raw frequency of any term in the document:

$$tf(t,d) = \frac{f(t,d)}{\max\{f(w,d) : w \in d\}}$$

Where:

- f(t,d) is the frequency, that is ,the number of times the word t appears in the Document d,
- max {f(w,d):w∈d} is maximum raw frequency of any term in the document.

IDF (Inverse Document Frequency) is a reciprocal of the number of Documents in which the word occurs. The inverse document frequency is a measure of whether the term is common or rare across all documents. For example, in a corpus of fashion documents, the term “fashion” or “model” will appear all over the corpus. When it is already a popular term, it will not provide much information. IDF is calculated as:

$$idf(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Where:

- $|D|$: total number of documents in the corpus D,
- $|\{d \in D : t \in d\}|$:number of documents where the term t appear(it means $tf(t,d) \neq 0$).

If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to $1+|\{d \in D : t \in d\}|$.

Mathematically, the base of the log function does not matter and constitutes a constant multiplicative factor towards the overall result. In other words, the change in the base of the log function will not change the ratio between IDF results.

$$tfidf(t,d,D) = tf(t,d) \times idf(f,D)$$

A high weight in TF-IDF is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms, and keep the importance term (or keyword).

For the purpose of retrieving useful information related to fashion from online social network data, we use TF-IDF measure for transforming data to space vector model. However, we propose a technique that reduces the dimensional specifications of this Vector Space Model to train an SVM (Support Vector Machine) classifier.

2.3 Validation Measuring for Retrieved Documents

Precision, recall, and the F measure [6, 7] are set-based measures. They are computed by using unordered sets of documents. In a ranked retrieval context, appropriate sets of retrieved documents are naturally given by the top k retrieved documents. For each such set, precision and recall values can be plotted to give a precision-recall curve, such as the one shown in Fig. 2.

For classification tasks, the terms true positives, true negatives, false positives, and false negatives (see Fig. 3) compare the results of the classifier under test with trusted external judgments. The terms positive and negative refer to the classifier's prediction, and the terms true and false refer to whether that prediction corresponds to the external judgment. They are defined as an experiment from P positive instances and N negative instances for some conditions. The four outcomes can be formulated as showed in Fig. 3 in the form of a confusion matrix.

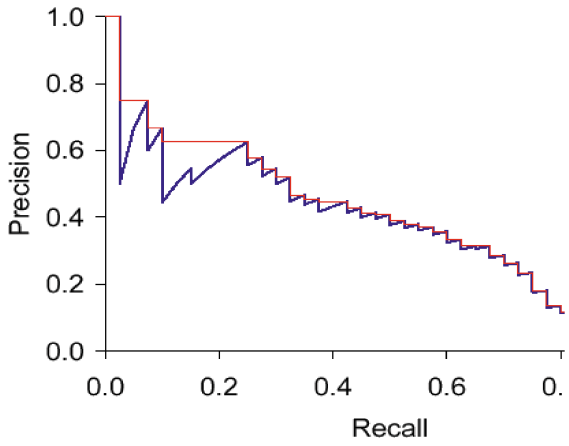


Fig. 2 Precision/Recall graph[1].

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	True positive (tp)	False positive (fp)
	Predicted condition negative	False negative (fn)	True negative (tn)

Fig. 3 P positive instances and N negative instances for some condition.

Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. It is usually expressed as a percentage.

$$precision = \frac{tp}{tp + fp}$$

Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. It is usually expressed as a percentage.

$$recall = \frac{tp}{tp + fn}$$

[1] http://nlp.stanford.edu/IR-book

Accuracy is the proximity of measurement results to the true value.

$$Accuracy = \frac{tp + tn}{(tp + tn + fp + fn)}$$

F measure is a measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score is the harmonic mean of precision and recall:

$$F = \frac{recall \times precision}{(recall + precision)/2}$$

Commonly used evaluation measures including Recall, Precision, F-Measure and Rand Accuracy are applied due to their origin in Information Retrieval. In fact, sometimes we can not directly use these measures to compare two lists of ordered documents returned because of independence of the internal order of the documents [8]. To measure the quality of an ordered list of documents, the average precision of all the relevant documents in the ordered list can be calculated.

2.4 Text Classification

Text (or Document) classification [3, 9-11] is a problem belonging to data mining, but focus on unstructured or semi-structured data [12]. The problem of text classification can be found in applications in a wide variety of domains in text mining. Some examples of domains in which text classification is commonly used are: news filtering and organization (text filtering); document organization and retrieval [13]; opinion mining (sentiment); email classification and spam filtering.

Simply, text classification process includes some steps: (1) data preprocessing; (2) model machine learning; (3) classification processing in training model and (4) result interpretation and reporting. Data pre-processing is an important step in the data mining process. Data pre-processing includes cleaning, normalization, transformation, feature extraction, selection and transforming the text data into space vector model. Machine learning focuses on prediction, based on known properties learned from the training data. Some key methods which are commonly used for machine learning are decision trees; pattern (rule)-based classifiers; support vector machine (SVM) classifiers; neural network classifiers; Bayesian classifiers. Some of the techniques have been converted into software that can be used such as BOW toolkit [14], Mallot [15], WEKA[2], and LingPipe[3]. In this research, we use support vector machines for model training because of advantages [2].

The prediction process is used through the training model in the new text data and a label to unclassified instances is assigned. Finally, the results are interpreted

[2] http://www.cs.wailato.ac.nz/ml/weka
[3] http://alias-i.com/lingpipe

and reported. Based on this general process, we propose a comprehensive framework for retrieving useful information related to fashion from online social network data. It will be discussed in further details in the next session.

3 Framework for Retrieving Information from Social Network

Process of our framework is as follow, we use a measure to determine the importance of a word in the text, called TF-IDF [4] (Term Frequency – Inverse Document Frequency) as described in session 2.2 Based on the TF-IDF measure, we transform the document text of information into Vector Space Model (VSM) [1, 4]. This model allows the text to be represented by the vector in n-dimensional space, each dimension corresponding to the index. In this space, each component of the text vector represents weighted measure of the index corresponding in that text. First, based on the vectors representation in the document, text classification model is built from a training set by means of a Support Vector Machine algorithm. To build the training sample, we use the common account that provides

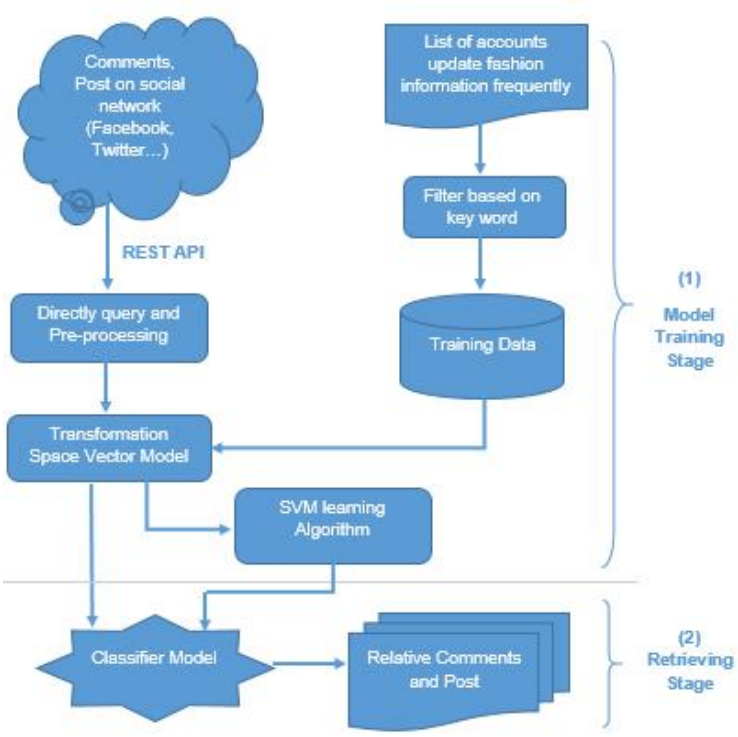


Fig. 4 Framework for model training and retrieving fashion data

updates about fashion (Ex accounts of www.Fashionista.com andwww.elle.com in Twitter and Facebook). Experiments on two popular social networks Facebook and Twitter, have been chosen to see the efficiency and accuracy of the model in the extracted information about fashion.

Fig. 4 presents our framework that automatically trains a model for retrieving fashion data from Facebook and Twitter. Basically, there are two stages for retrieving information in this framework (1) model training and (2) retrieving fashion contents using the induced classification model.

In the stage of model training, social network data (comments, post…) are real-time collected by using REST and Graph protocol (as REST APIs in Twitter and Graph API in Facebook). Some popular accounts or fan pages providing status of fashion (Ex accounts of www.Fashionista.com and www.elle.com on Twitter and Facebook) are used to train a SVM model. Text data are transformed into Space Vector Model by using IF-IDF measure. Using the model trained from training data, new information from social network is classified and related fashion information is retrieved and saved. In the next stage, we also collect data from account in Facebook and Twitter that are not related to fashion such as: economic; weather; traffic; technology...These data are merged with related fashion data to test the accuracy of the SVM model. In the next session, we present in a more detailed way the data used in the experiments carried out to test and evaluate the effectiveness of this framework.

4 Data Model Scenario

To perform the experiments on the proposed framework, we use real-time text data of two popular social networks including Twitter and Facebook. The data are accessed through the APIs provided, that is, REST APIs in Twitter and Graph API in Facebook. Per time launch framework to test, the framework collects 1000 posts or comments from Twitter and Facebook in the real-time.

Training data are built based on data related to fashion provider. That is, we use comments and posts from account on Facebook and Twitter of website www.Fashionista.com and www.elle.com. In addition, we use some other accounts which are not related to fashion. Such as economic data[4]; weather data[5]; traffic data[6]; technology data[7]. Then, we merge them to evaluate the trained model. We use both accounts on Facebook and Twitter of those websites for this task.

Fig. 5 and Fig. 7 present important keywords for two datasets related to fashion and related to traffic as discussed above. We can see the related keywords in Fig. 5 such as fashion, style new… Otherwise, some words as: accident, roadway, lane… appear in the Fig. 7, which it is reasonable.

4 http://www.cnbc.com/economy/
5 http://www.weather.com/
6 https://tfl.gov.uk/traffic/status/?cid=trafficnews
7 http://techcrunch.com/

Fig. 5 Visualization of fashion related data retrieved from Facebook account "Fashionista_com". The more important a word is the larger its size is [8].

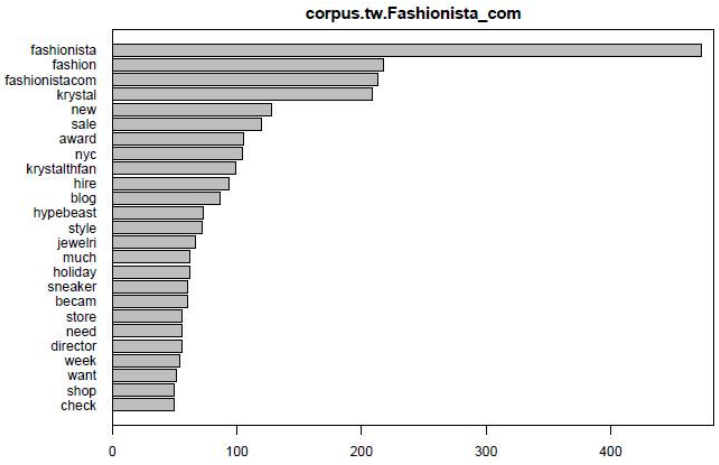


Fig. 6 Frequency of 1000 twitters sample get from "Fashionista_com" account

[8] Using Wordcloud library in R language

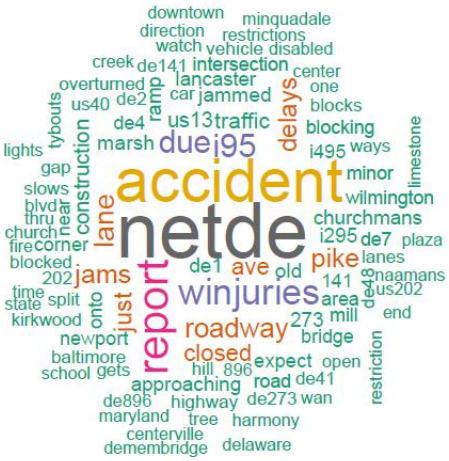


Fig. 7 Visualization traffic related data retrieved from Facebook account "traffic". The more important a word is, the larger the text size.

5 Experimental Results

With the data source described in the previous section, we perform several experiments to test the effectiveness and accuracy of the proposed framework. From each source (Facebook and Twitter), we extract 6 datasets (2 datasets related to fashion and 4 datasets not related to fashion). We use both Accuracy and F measure to evaluate the accuracy of filtering. Because F measure is derived from Recall and Precision, we also show the two measures for reference purpose.

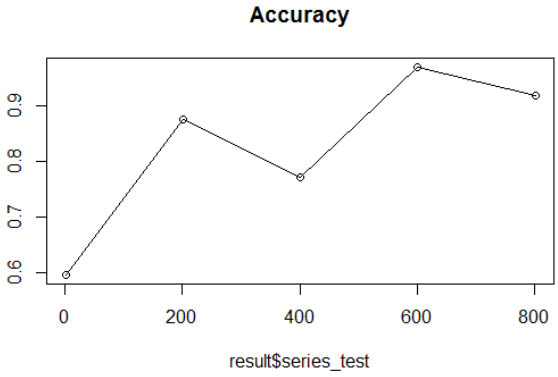


Fig. 8 Experiments with the parameter minfreq to reduce data dimensionality. The highest accuracy was achieved for the value 600.

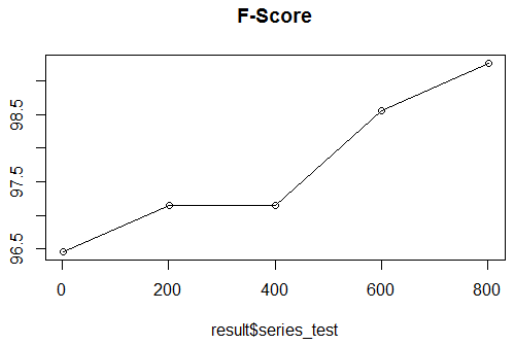


Fig. 9 Experiments with the parameter series_test to reduce data dimensionality. The highest accuracy was achieved for the value 800.

For each run, we need two datasets, one for fashion-related and another one for fashion-unrelated contents. Each dataset contains 1000 posts (comments) obtained from Twitter (Facebook). Then we merge them for training and testing. After that, we transform the text into Vector Space Model based on the TF-IDF measure and reduce the dimensionality of data collection. That can be done by analyzing the frequency of terms that appear in the datasets. To prevent bias, we create several sub dataset from original. That is, each original dataset is partitioned into 5 sub dataset, each one contains 200 posts. Then we train and test with these types of data. Fig. 7 and Fig. 8 show that the highest accuracy was achieved for words appearing more than 500 times in the dataset using as vocabulary.

The experimental results that show our filters have high accuracy (> 91%) in the experimental cases. More detailed reference in Table 1 and Table 3.

Table 1 Experimental result based on data retrieved from Twitter with 5 sub-datasets identified by indices:

Test	Index	Recall	Precision	F measure	Accuracy
1	1	100.00	55.28	**55.28**	**0.59**
2	201	100.00	80.00	**80.00**	**0.87**
3	401	100.00	68.69	**68.69**	**0.77**
4	601	98.53	95.71	**95.71**	**0.97**
5	801	98.53	87.01	**87.01**	**0.92**

Table 2 Experimental result based on data retrieved from Facebook with 5 sub-datasets identified by indices:

Test	Index	Recall	Precision	F measure	Accuracy
1	1	100.00	93.15	**96.45**	**0.96**
2	201	100.00	94.44	**97.14**	**0.97**
3	401	100.00	94.44	**97.14**	**0.97**
4	601	100.00	97.14	**98.55**	**0.98**
5	801	100.00	98.55	**99.27**	**0.99**

The chart in the Fig. 10 shows experimental result with 6 datasets. For each run, we use two datasets. One is related to fashion and the other is not. The F measure in this chart is computed from tests performed on Twitter datasets.

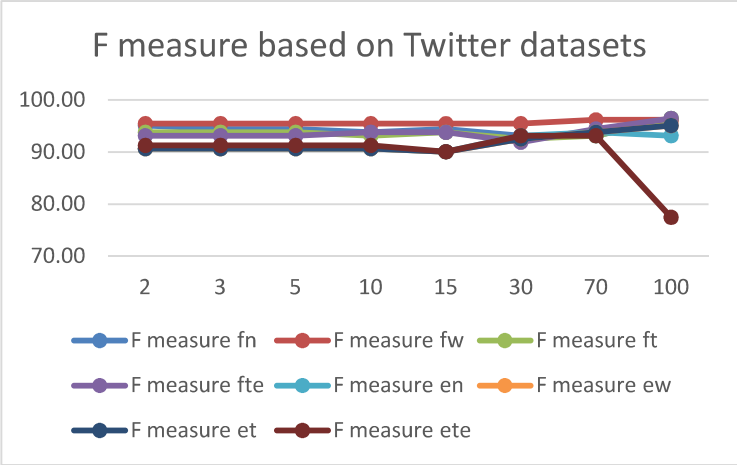


Fig. 10 F measure with 8 times running base series_test<- c (2, 3, 5, 10, 15, 30, 70, 100) partitioned into 8 subdatasets.

6 Conclusion

In this research, we develop a framework to automatically retrieve relevant contents related to fashion from online social network data. We employ machine learning, which automatically train the model to classify fashion contents. The framework tested with datasets from Facebook and Twitter, shows a good performance.

The information about fashion is very useful for various application domain, e.g., decision support systems; automatic surveillance systems; creating suggestion and recommendation systems. The future work for this approach research that

uses data retrieving for useful related applications can be the prediction of fashion opinion of users in the social network. We also intend to extend the framework with the use of more reliable data mining based techniques.

References

1. Ikonomakis, M., Kotsiantis, S., Tampakas, V.: Text classification using machine learning techniques. WSEAS Transactions on Computers **4**(8), 966–974 (2005)
2. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys (CSUR) **34**(1), 1–47 (2002)
3. Aggarwal, C.C., Zhai, C.: A survey of text classification algorithms. In: Mining text data, pp. 163–222. Springer (2012)
4. Turney, P.D., Pantel, P.: From frequency to meaning: Vector space models of semantics. Journal of Artificial Intelligence Research **37**(1), 141–188 (2010)
5. Rajaraman, A., Ullman, J.D., Ullman, J.D.: Mining of massive datasets, vol. 77. Cambridge University Press, Cambridge (2012)
6. Powers, D.M.: Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation (2011)
7. Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters **27**(8), 861–874 (2006)
8. Han, J., Kamber, M., Pei, J.: Data mining: concepts and techniques: concepts and techniques. Elsevier (2011)
9. Berry Michael, W.: Automatic Discovery of Similar Words. Survey of Text Mining: Clustering, Classification and Retrieval, vol. 200, pp. 24–43. Springer Verlag (2004)
10. Kroeze, J.H., Matthee, M.C., Bothma, T.J.D.: Differentiating between data-mining and text-mining terminology. South African Journal of Information Management **6**(4) (2004)
11. Nalini, K., Sheela, L.J.: Survey on Text Classification (2014)
12. Berson, A., Smith, S.J.: Data warehousing, data mining, and OLAP. McGraw-Hill, Inc. (1997)
13. Grimmer, J., Stewart, B.M.: Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political Analysis, p. mps028 (2013)
14. McCallum, A.K.: Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering (1996)
15. McCallum, A.K.: MALLET: A Machine Learning for Language Toolkit (2002)