# FOUNDATIONS OF UNCERTAINTY MANAGEMENT FOR TEXT-BASED SENTIMENT PREDICTION

**Sayyed-Ali Hossayni**

Universitat de Girona

DOCTORAL THESIS

# FOUNDATIONS OF UNCERTAINTY MANAGEMENT FOR TEXT-BASED SENTIMENT PREDICTION

Sayyed-Ali Hossayni

2018

Universitat de Girona

DOCTORAL THESIS

# FOUNDATIONS OF UNCERTAINTY MANAGEMENT FOR TEXT-BASED SENTIMENT PREDICTION

Sayyed-Ali Hossayni

2018

Doctoral Program in Technology

Supervisor: Jose Luis De la Rosa Esteva

Co-supervisor: Esteban Fermin Del Acebo Peña

2$^{nd}$ Co-supervisor: Mohammad-Reza Akbarzadeh-Totonchi

Presented in partial fulfilment of the requirements for a doctoral degree from the University of Girona

### The obtained publications

- Sayyed-Ali Hossayni, Mohammad-R Akbarzadeh-T, Diego Reforgiato Recupero, Aldo Gangemi, and Josep Lluís de la Rosa i Esteva. **Fuzzy Synsets, and Lexicon-based Sentiment Analysis**, published in In EMSA-RMed, edited by Dragoni M. Deng Y., Denecke K., Declerck T., Recupero D.R. Vol. 1613, CEUR Workshop Proceedings (Scopus code 21100218356), 2016.
- Sayyed-Ali Hossayni, Mohammad-R Rajati, Esteve Del Acebo, Diego Reforgiato Recupero, Aldo Gangemi, Mohammad-R Akbarzadeh-T, Josep Lluis de La Rosa i Esteva, **Towards Interval Version of Fuzzy Synsets**, published in Frontiers in Artificial Intelligence and Applications (2016 Vol. 288), ISSN: 0922-6389, Scopus cites/doc (2-year): 0.346.
- Sayyed Ali Hossayni, Hadi Kalamati, Albert Trias i Mansilla, Ehsan Mollazadeh-A, Mohammad-R Akbarzadeh-T, Esteve Del Acebo, Josep Lluís de la Rosa i Esteva, **Optimizing Significance Weights in Linear Time-Complexity: Toward Adaptive Similarity Measures in Collaborative Filtering,** submitted for journal consideration.
- Sayyed-Ali Hossayni, Yousef Alizadeh, Mohammad-R Akbarzadeh-T, Diego Reforgiato Recupero, Aldo Gangemi, Behrouz Minaei-Bidgoli, Esteve Del Acebo, Josep Lluís de la Rosa i Esteva, **An Algorithm for Fuzzification of WordNets and its Application in Sentiment Analysis,** submitted for journal consideration.
- Sayyed-Ali Hossayni, Vahid Tavana, Seyed M. Hosseini Nejad, Mohammad-R Akbarzadeh-T, Esteve Del Acebo, Josep Lluis De la Rosa i Esteva, Enrico Grosso, Massimo Tistarelli, **Toward Real-time Text-based Multibiometric Forensic Document Analysis,** submitted for journal consideration.
- IET Biometrics, ISSN 2047-4938, IF 1.382.
- Sayyed-Ali Hossayni, Yousef Alizadeh, Vahid Tavana, Mohammad-R Akbarzadeh-T, Josep Lluís de la Rosa i Esteva, Aldo Gangemi, Esteve Del Acebo, Diego Reforgiato Recupero, Behrouz Minaei Bidgoli, **Probabilistic and Possibilistic Weibull Naïve Bayes Text Classifiers and Their Application in Uncertainty-handling in Authorship Attribution,** submitted for journal consideration.

## List of Abbreviations

| | |
|---|---|
| BoFS | Bag of Fuzzy Synsets |
| BoFWS | Bag of Fuzzy Wordsenses |
| BoS | Bag of Synsets |
| BoWS | Bag of Wordsenses |
| CF | Collaborative Filtering |
| FWN | Fuzzy WordNet |
| GSW | Generalized SW |
| MAE | Mean Square Error |
| MOA | Meta-heuristic Optimization Algorithm |
| NLP | Natureal Language Processing |
| PNB | Poisson Naïve Bayes |
| RMSE | Root Mean Square Error |
| SM | Surrogate Model |
| sn-gram | Syntactic n-gram |
| SW | Significance Weight |
| WLD | WordNet-like Lexical Databases |
| WNB | Weibull Naïve Bayes |
| WNPC | Weibull Naïve Possibilistic Classifier |
| WSD | Word Sense Disambiguation |

## Index of Figures

## Index of Tables

*To anyone in the world who is calling the free people of the world*
*to rescue them from oppression.*
*We are the children of Hussain, who sacrificed his life to propagate justice in the hearts of tens of millions of his lovers, who visit his shrine in Karbala, annually.*
*We assure the under-oppression people of the world that the future is for us.*

*Then, to my family*
*who suffered the hardest situations of being far from me, more than one year*
*just because they are kind, they are with my heart, and they are justice supporters…*

**Acknowledgments**

This research has partly been supported by AGAUR 2013 DI 012; AHD 13960101-01, 13960101-02, 13960101-03, 13960101-04, 13960101-05; IN2013-48040-R (QWAVES); QWAVES- RTC-2014-2576-7; Nuevos métodos de automatización de la búsqueda social basados en waves de preguntas; ANSwER: ANálisis de SEntimiento y segmentación de las Redes Sociales para la generación de leads y el análisis de complementariedad de marcas; RTC-2015-4303-7; Fidelización Cooperativa- RTC-2016-4836-7, Smart Data Discovery y Natural Language Generation para el Rendimiento de los Medios Digitales, and the grup de recerca consolidat CSI-ref. 2017 SGR 1648, MARIO EU Proj.; IDENTITY– n.690907 H2020-MSCA-RISE-2015; ANSwER- RTC-2015-4303-7; CSI-2014 SGR 1469; NIR-VANA– n. 681787-2; H2020-INNOSUP-2015-2. Convenzione triennale tra la Fondazione di Sardegna e gli Atenei Sardi Regione Sardegna - L.R. 7/2007 annualit\`{a} 2016 - DGR 28/21 del 17.05.201, CUP: F72F16003030002.

Table of Contents

**Resum**

L’anàlisi del sentiment dels usuaris de les xarxes socials és una tasca interessant, molt estudiada per la comunitat científica d'anàlisi de sentiment. De la mateixa manera, realitzar prediccions sobre l'opinió dels usuaris a les xarxes socials o a les plataformes de comerç electrònic és una altra tasca interessant, estudiada per la comunitat científica especialista en sistemes recomanadors. D'altra banda, hi ha un camp d'estudi nou que aprofita els anteriors camps de recerca per predir les opinions "no expressades" pels usuaris, basant-se en els seus sentiments escrits o en la seva semblança. Tot i que les dades extretes de les xarxes socials tenen un alt nivell d'incertesa, cap ni un de la dotzena d'estudis portats a terme en el camp de l'anàlisi de sentiment fins avui posa el focus en l'estudi d'aquesta incertesa. Aquesta tesi introdueix els fonaments bàsics per construir un sistema de predicció del sentiment que té en compte la incertesa, per mitjà de teoria de la possibilitat, teoria difusa i teoria de la probabilitat. A més, es defineix un projecte-comunitat internacional anomenat probabilistic/possibilistic *Text-based Emotion Rating* (pTER) per omplir i enriquir el, fins ara buit, camp de l'anàlisi de la incertesa en predicció de sentiment. pTER inclou dos subprojectes: pTER escalar i pTER intervalar. Aquesta tesiaporta cinc estudis de recerca fonamentals en el subprojecte pTER escalar. Tot i que aquests estudis són suficients per als objectius plantejats, hem deixat que el projecte escalar pTER sigui disseminat gràcies també a la difusió que estan fent altres investigadors de l’esmentadacomunitat. A part dels resultats presentats de pTER escalar, també proposem un projecte de recerca en l'apartat de pTER intervalar que va un pas més enllà en el tractament de la incertesa i té en compte els errors de mesura del pTER escalar. Els resultats presentats en pTER escalar i intervalar estan distribuïts en tres fases: (I) plataforma NLP per al tractament de la incertesa, (II) tractament de la incertesa per a l'anàlisi de sentiment i (III) filtre col·laboratiu per al tractament de la incertesa. Els experiments realitzats en aquesta tesi proven la superioritat de la nostra aproximació al tractament de la incertesa en aquestes tres fases, en comparació amb l'estat de l'art.



**Paraules clau**: Predicció del sentiment, tractament de la incertesa, anàlisi de sentiment, atge col·laboratiu, teoria de la possibilitat, teoria difusa, teoria de la probabilitat.

**Resumen**

Analizar el sentimiento de los usuarios de las redes sociales es una tarea atractiva, bien cubierta por la investigación de Análisis de Sentimiento. Además, predecir la calificación / opinión de los usuarios en redes sociales o plataformas de comercio electrónico es otra tarea atractiva, cubierta por la investigación en *Recommender Systems*. Sin embargo, hay un campo de estudio bastante nuevo que aprovecha ambos ámbitos mencionados para predecir la opinión "no expresada" de los usuarios, en función de sus sentimientos escritos y su similitud. Aunque de la red social se extraen datos (debido a la escasez de los elementos tratados por diferentes usuarios), abarcando e un alto volumen de incertidumbre, ninguna de las pocas docenas de estudios realizados en el campo Predicción del sentimiento se centra en la gestión de la incertidumbre mencionada. En esta disertación, presentamos los fundamentos necesarios para construir un sistema de Predicción del Sentimiento de manejo de la incertidumbre, mediante la teoría de la posibilidad, la teoría difusa y la teoría de la probabilidad. Por otra parte, definimos un proyecto internacional llamado Probabilistic / Possibilistic basado en el Emotion Rating (pTER) de textos para llenar y luego enriquecer el área de investigación de la gestión de la incertidumbre en *Sentiment Prediction*. pTER comprende dos subproyectos: pTER escalar e pTER Intervalar. Esta disertación proporciona cinco estudios de investigación fundamentales en el pTER escalar. Aunque los estudios mencionados son suficientes para el sistema objetivo, dejamos que el sistema escalar pTER, en sí mismo, se disemine solo después que pueda usar toda su potencia contando con los proyectos de investigación en curso de los otros investigadores que se encargan del proyecto pTER, definidos por esta disertación. Además de los estudios escalar-pTER presentados, también proponemos un estudio de investigación en el proyecto de intervalar pTER que va un paso más allá en el manejo de la incertidumbre y tiene en cuenta los errores de medición de los subsistemas pTER escalares. Los estudios presentados los pTER escalar e intervalar pertenecen a tres fases: (I) plataforma de PNL para el manejo de la incertidumbre, (II) análisis de la confianza para el manejo de la incertidumbre y (III) manejo de la incertidumbre para el filtrado colaborativo. Los experimentos realizados en esta disertación demuestran la superioridad de nuestros enfoques de manejo de la incertidumbre en todas estas fases, en comparación con el estado del arte correspondiente.

**Palabras clave:** Predicción del sentimiento, Manejo de la incertidumbre, Análisis del sentimiento, Filtrado colaborativo, Teoría de la posibilidad, Teoría borrosa, Teoría de la probabilidad.

**Abstract**

Analyzing the sentiment of Social Networks users is an attractive task, well-covered by the Sentiment Analysis research communities. Alongside, predicting the rating/opinion of users in Social Networks or e-commerce platforms is another attractive task covered by the Recommender Systems research communities. However, there is a rather new field of study that takes advantage of both of the mentioned scopes to predict the "unexpressed" opinion of users, based on their written sentiments and their similarity. Although the Social Network extracted data (due to the sparsity of the addressed items by different users) deals with high volumes of uncertainty, none of the few dozens of conducted studies in the Sentiment Prediction field focuses on managing the mentioned uncertainty. In this dissertation, we introduce the necessary foundations for constructing an Uncertainty-handling Sentiment Prediction system, by means of possibility theory, fuzzy theory, and probability theory. Moreover, we define an international project called probabilistic/possibilistic Text-based Emotion Rating (pTER) to fill and then enrich the gap of uncertainty management in Sentiment Prediction. pTER comprises two sub-projects: Scalar and Interval pTER. This dissertation provides five foundational research studies in the scalar pTER. Although the mentioned studies are sufficient for the targeted system, we let the scalar pTER system, itself, to be disseminated only after it can use its entire potency by utilizing the in-progress research projects of the other researchers of the pTER project, defined by this dissertation. In addition to the presented scalar-pTER studies, we also propose one research study in the interval pTER project which goes one step further in Uncertainty-handling and takes the measurement errors of the scalar pTER sub-systems into account. The presented studies in scalar- and interval-pTER belong to three phases: (I) Uncertainty-handling NLP platform, (II) Uncertainty-handling Sentiment Analysis, and (III) Uncertainty-handling Collaborative Filtering. The conducted experiments in this dissertation prove the superiority of our Uncertainty-handling approaches in all of these phases, in comparison to the corresponding state-of-the-art.

**Keywords**: Sentiment Prediction, Uncertainty-handling, Sentiment Analysis, Collaborative Filtering, Possibility Theory, Fuzzy Theory, Probability Theory.

# 1 Introduction

This dissertation provides the required foundations for a system that utilizes written records (e.g. tweets, posts, etc.) of users of social networks to predict their sentiment/emotion about the items which they have not, even, written about, by means of Uncertainty-handling techniques. In other words, to the best of our knowledge, this dissertation, for the first time, proposes a Sentiment Prediction system, equipped with Uncertainty-handling techniques. Uncertainty-handling techniques can have a wide variety, including the Fuzzy mathematics, Fuzzy logic, Possibility theory, soft computing, and stochastic processes techniques.

However, Sentiment Prediction is comprised of two key steps: (1) Sentiment Analysis and (2) Recommender Systems -mainly Collaborative Filtering (CF). Additionally, for Uncertainty-handling in the Sentiment Analysis phase, there is also the need to go one step deeper and, moreover, address the Uncertainty-handling techniques in (3) Text Mining or Natural Language Processing platforms.

In this chapter, we firstly propose a literature review on different sub-fields of the abovementioned research fields. Then, considering that this dissertation is one of the main parts of an international research project (called pTER), we introduce the pTER project to present a better insight on the scope and application of this dissertation.

Then, chapter 2 is dedicated to the objectives of this dissertation, in which, the addressed problem and the fundamental doctrine of this dissertation are discussed. Then, chapter 3 proposes the presented algorithms in this dissertation. They are two algorithms in the field of Lexical Resources, one auxiliary text mining model for the evaluation purposes, one algorithm for text classification, one algorithm in the CF field, and one optimization algorithm, which is necessary for the proposed CF-related algorithm. Then, chapter 4 is devoted to experimenting the presented algorithms in chapter 3 and evaluating them, and finally, chapter 5 is devoted to the concluding remarks and future works.

## 1.1 Literature review

As mentioned above, on the one hand, Sentiment Prediction is a multidisciplinary approach, and on the other hand, we are focusing on a specific approach to Sentiment Prediction (uncertainty handling). Therefore, we deal with a wide scope of related studies. Some of them are related to Text Mining as we are going to address managing the uncertainty existing in basic tools and resources of Text Mining. Some other related studies belong to the field of Machine Learning and classification as we address the Machine-Learning-based approach to Sentiment Analysis. We also should deal with the CF-related studies, specifically the so-called cold start problem, which frequently occurs in Sentiment Prediction systems.

Thus, the structure of this literature review section is as follows: Subsection 1.1.1 focuses on the studies focusing on the Sentiment Prediction as a whole. Then, subsection 1.1.2 addresses WordNet-like Lexical Databases. After that, subsection 1.1.3 reviews an evolution line of the Text Mining models in the state-of-the-art. Thereafter, subsection 1.1.4 considers Sentiment Analysis and the role of fuzzy synsets on it. However, section 1.1.5, instead of reviewing the related studies to Machine-Learning-based Sentiment Analysis, analyzes Machine-Learning-based Authorship

Attribution methods. It is because, on the one hand, due to the request of one of the funding projects of this dissertation (IDENTITY H2020 European project), the author of this dissertation had to be engaged with Authorship Attribution for almost six months of his Ph.D., and on the other hand, the Uncertainty-handling classifier that will be presented in chapter 3 is also applicable on any other Text Mining problem, including Sentiment Analysis. Nevertheless, subsection 1.1.6 (as the last subsection of this literature review) addressed the cold-start problem in recommender systems, and specifically in CF.

### 1.1.1 Sentiment Prediction

The main goal of Sentiment Prediction systems is the prediction of the sentiments and not merely extracting/analyzing them, such as what in Sentiment Analysis. In Sentiment Prediction, the power of Social Networks is utilized for extracting the implicit ratings of users for other users or other entities.

It is because Social Networks have become a multipurpose media for sharing information and marketing. Numerous users including enterprises, political people, and legal entities utilize Social Networks such as Facebook and Twitter to express what they believe, what they feel, and what they urge others to believe and feel. The users of online Social Networks frequently react to such messages, as well as their reaction to the messages of each other. They express their emotional reactions on the subject. Detecting the sentiment of people on different events, entities, etc. can be essential for analyzing a model of thinking/liking for each of them. For answering to messages, any opinion (positive, negative, unbiased) can be communicated regarding past messages or the discourse subjects. Thus, Social Networks dialogs can be utilized for recognition of users' emotions on different subjects, analyzing the relation/similarity, and using this similarity as a basis for predicting their unexpressed opinions.

However, analyzing the relationship between the manners of thinking of different people has numerous difficulties related to the arrangement of this issue. One of the most critical issues is sparsity of the information which yields the uncertainty (the main research concern and the motivation of this dissertation). It is because people do not generally express their feeling towards all points or users, and it makes challenges for the mentioned steps of Sentiment Prediction.

Before starting to address the conducted studies in this research field, we introduce a concept, as it is frequently used in the majority of the related works. We call this concept as User/Item opinion matrix. It contains the sentiment/opinion of the users of social networks about different items or entities. In some studies, due to the Recommender-System-nature of the database / problem, and therefore the existence of the explicit ratings of people about items, these information, explicitly, exist. However, despite this fact, the main tools for creating / enriching this matrix is aspect-based / entity-based sentiment analysis. Then, this matrix will be utilized by prediction algorithms (mainly CF) for predicting its un-extracted cells.

As a hypothetical illustrating example of User/Item opinion matrix, suppose the three hypothetical Twitter users Joseph, Adam, and Peter. Suppose that Joseph has twitted about iPhone X and LG V30 smart phones, Adam has twitted about Samsung Galaxy S8 and iPhone X, and Peter has only twitted about LG V30. After applying an entity-based sentiment analysis on the Tweets of these three Twitter users, we will have (the ratings range in [-1,1]).

**Table 1.1. A hypothetical Sentiment Prediction progress on three imaginary Twitter users and discussed entities, by means of entity-based sentiment analysis (the left matrix) and then Collaborative Filtering on the extracted data for predictions (the bold numbers in the right matrix).**

| | iPhone X | Samsung Galaxy S8 | LG V30 | | | iPhone X | Samsung Galaxy S8 | LG V30 |
|---|---|---|---|---|---|---|---|---|
| **Joseph** | 0.9 | - | 0.6 | | **Joseph** | 0.9 | **0.73** | 0.6 |
| **Adam** | 0.7 | 1 | - | $\xrightarrow{yields}$ | **Adam** | 0.7 | 1 | **0.51** |
| **Peter** | - | - | 0.7 | | **Peter** | **0.84** | **0.97** | 0.7 |

As you can see in the Table 1.1, firstly, the opinion of these users about the discussed entities are extracted (depicted in the left matrix), and then, these extracted information are delivered to a CF algorithm for predicting the unexpressed opinions (bolded in the right matrix).

Evidently, this example is both imaginary and basic. Hundreds of studies have developed this idea and handled the difficulties / drawbacks, related to this progress. In the following, we propose a review of the entire “well-known” studies of this field.

(Dabeer, 2012) for the first time utilized the expression “Sentiment Prediction” in this context. Albeit, a few months after (Dabeer, 2012)[1], (Koukourikos, Stoitsis, & Karampiperis, 2012) also utilized it in their more well-known study.

However, disregarding utilization of this term, (C. W. Leung, Chan, & Chung, 2006), for the first time, provide a Sentiment Prediction system by integrating CF and Sentiment Analysis. They approach extracting the textually-expressed user preferences by Sentiment Analysis, mapped into the CF rating ranges, as the first proposed platform for Sentiment Prediction.

(B Galitsky & McKenna, 2008) in their patent propose a product recommender system based on the text query of customers. They analyzed the text of the asked question and analyze the writer’s sentiment about the item/feature, based on which the recommendation is done.

(Jakob & Weber, 2009) focus on movie reviews and upgrade the corresponding recommendations by means of analyzing the sentiments of text reviews. They extract different aspects of movies and consider them as opinion targets to be used as features in CF. Their experiments on a collected dataset with star ratings of thousands of movies prove their superiority over the compared state-of-the-art CF system which only utilizes ratings and genres.

(Faridani, 2011) proposes User/Item opinion matrices which provide the ratings based on different aspects of items such as service/price/value. He uses “Canonical Correlation Analysis or CCA” to derive a mathematical model for being used as a multivariate regression system. One of the

[1] The venue related to (Dabeer, 2012), was held on Feb 2012, whereas the venue related to (Koukourikos et al., 2012) was held on Sep 2012.

advantages of his model is its capacity for offline training and live usage. His experiments prove the superiority of CCA over the similar experiments provided by Principal Component Analysis on the same extracted opinion space.

(C. W. K. Leung, Chan, Chung, & Ngai, 2011) present a probabilistic inference framework for rating extraction (called PREF) for preference mining of users based on their reviews and mapping them to the CF numerical scales. They extract opinion words as well as features of products by means of NLP tools. Their method also estimates the "sentimental orientation" and "strength" of opinion words. Their proposed method can discriminate between semantically similar words by assigning different sentimental orientation. Their experiments prove the better efficiency of their proposed method over a number of the state-of-the-art counterparts, even by small training sets. Their proposed integration of PREF and CF, once again, demonstrates the added value of this integration.

(Kawamae, 2011) aims in the prediction of the future reviews of authors based on their previous ones. He presents a "Latent Evaluation Topic model" for inferring the preferences from their written texts. By this means, he reduces the dimensionality of the textual reviews to a low-dimensional set, comprising of the extracted latent variables. He uses this method, both, for the analysis and prediction phases.

(Levi, Mokryn, Diot, & Taft, 2012) propose a hotel recommendation system which its main data is text reviews of the people. They provide their recommendations based on context groups which are defined in their study based on the extracted reviews. For this purpose, they propose a new weighted Text Mining algorithm. Their recommendation approach utilizes, in the same system, (1) unsupervised clustering algorithm for building a hotel vocabulary for entities/aspects of hotels, (2) semantic analysis for knowing the sentiment of users for different hotel features, and (3) intent and nationality group profiling. For experiments, they utilize data from "TripAdvisor" and "Venere" websites and test 150 trip planning cases. The comparison of their system with the real-world suggestions of the mentioned websites proves their superiority by 20% of more satisfaction for the corresponding users.

(Y. Wang, Liu, & Yu, 2012) focus on movie reviews and provide an auxiliary User/Item opinion matrix extracted by opinion mining to improve the CF systems. Their proposed system is based on two phases: (1) opinion mining in which they summarize the opinion of each user on different aspects of different items, and (2) rating inference which infers the generic rating of each aspect of each item. Their experiments on the corresponding movie dataset illustrate the meaningful accuracy improvement of their method in comparison with the counterparts.

(Dong, O'Mahony, Schaal, McCarthy, & Smyth, 2013) present a recommendation ranking strategy based on (1) the opinionated descriptions of products among the user-generated textual reviews and (2) similarity of users, and propose a Sentiment Prediction system. Their experiments on an Amazon© extracted data illustrates the advantages of their method. The same authors (Dong, O'Mahony, Schaal, McCarthy, & Smyth, 2016) also propose one of the few studies that make the recommendation by means of content-based filtering (rather than CF). They take advantage of natural language processing tools for extracting the item features from the raw textual data and, moreover, utilize Sentiment Analysis tools to associate the sentiment-loads of the mentioned features. Then, they propose a hybrid ranking strategy for recommendation, which utilizes (1) sentiment and (2) similarity for the product recommendation. Their experiments on the Amazon product domain prove the efficiency of their proposed method.

(J. Kim et al., 2013) focus on Sentiment Prediction in the short-length conversation threads of Twitter. They extract the (holder, target) opinion pairs by which their proposed User/Item opinion matrix is filled in. Then, they utilize the provided User/Item opinion matrix, alongside the CF algorithms to predict the unknown opinion of users. It is worthy to note that their system is one of the “pure” Sentiment Prediction systems and does not utilize any explicit rating. They also point out the data sparsity problem as a drawback of Sentiment Analysis in microblogs that can be alleviated by their proposed system. Their experiments on two Twitter datasets prove the efficiency and validity of their proposed method.

The popular research of (Hongyan Liu, He, Wang, Song, & Du, 2013) in Chinese-customized recommendation systems presents a new algorithm for recommendation by means of online review analysis. Their algorithm combines a new method for Sentiment Analysis and a new one for recommendation. It analyzes the distance between the extracted opinion-ratings and the star-ratings for identification of the preferences of users. They also present a feature-based opinion extraction method (including the feature extraction itself) based on the reviews characteristics in Chinese language. Their experiments on a restaurant recommendation system prove the superiority of their algorithm over the corresponding state-of-the-art.

(W. Zhang, Ding, Chen, Li, & Zhang, 2013) focus on Chinese-customized Recommender Systems. They fuse the extracted opinions of users by their explicit ratings in the User/Item opinion matrix. They propose fusion of self-supervised opinion classification results into CF data. They explicitly emphasize on the benefit of their system for being utilized as a rating-independent Sentiment Prediction system. Their experiments on the considered Chinese dataset prove the meaningful impact of the opinion ratings in enhancing the recommendation accuracy, in different rating-density circumstances.

(Krishna, Misra, Joshi, & Obaidat, 2013) focus on Recommender Systems on the cloud. They present a system by means of a learning automata and opinion mining. The learning automata is utilized for optimizing the rating recommendation outputted by the system, taking advantage of opinion mining. Their “Learning Automata-based Sentiment Analysis” is applied on the recommendation of places in neighborhood of the users’ current location by means of analyzing the written records of feedback. Their experiments prove the superiority of their system in comparison with the state-of-the-art.

(García-Cumbreras, Montejo-Ráez, & Díaz-Galiano, 2013) categorize recommender system users based on their average opinion in their written comments. For this purpose, they generate a novel corpus of textual and rating sentiments of users, extracted from the Internet Movie Database (the so-called IMDb). Their experiments prove the added value of integrating the textual comments. Even, by solely taking the textual opinions into account (Sentiment Prediction) they provide rather accurate results with a Root Mean Square Error equal to 1.868.

(D. Yang, Zhang, Yu, & Wang, 2013) propose a popular research on location recommendation, by improving, both, the preference model of locations and the recommendation algorithm itself. Firstly, they present a location preference model that is fed by both of the check-in data and text-based tips, the latter of which analyzed by Sentiment Analysis algorithms. Then, they propose a "location-based social matrix factorization algorithm" which considers both of the social influence of users and their similarity influence while recommending locations. Their experiments on location-based Social Networks datasets prove the superiority of their algorithm over the state-of-the-art, in better characterization of user preferences, while keeping the consistency of the preferences.

(Ren & Wu, 2013) focus on Twitter and address how to take advantage of the extracted knowledge from tweets for prediction of users' sentiments, on different topics. They solve this problem by a Matrix Factorization framework incorporated from "Social context" and "Topical context". Their experiments on the analyzed Twitter dataset prove the efficiency of both of the proposed social context and topical context in performance of the Sentiment Prediction.

(Pappas & Popescu-Belis, 2013)(Pappas & Popescu-Belis, 2016) focus on multimedia recommendations over TED talks. They utilize the Sentiment Analysis techniques in parallel with one-class CF and propose a "sentiment aware nearest neighbor model or SANN" for this recommendation task on TED talks. Their experiments prove providing meaningful improvement in successful recommendation of unseen data in comparison with a number of baseline counterparts.

(Diao et al., 2014) propose one of the most well-known research studies of this field by merging CF and topic modeling systems to become able for capturing the distribution of the users' interest as well as the distribution of the movies contents. In this regard, they create a link between relevance and interest for differentiating between the negative and positive sentiments, for each aspect. The advantage of their work to the prior studies is being unsupervised and its independence from any prior knowledge on ratings/genres/etc. They evaluate their model on a crawled dataset from IMDb, whose experiments prove the superiority of their proposed models.

(Yongfeng Zhang et al., 2014) propose another very popular research of this field. They present the Explicit Factor Model and construct "explainable" recommendations, while keeping the accuracy. After extracting the aspects or features of products, they apply sentiment level and phrase level analysis on the reviews of users. Then, they generate "recommendations" and "dis-recommendations" for different features of products, based on the interest of the target user, as well as the hidden features, learnt by their system. The interesting point of this research is intuitional explanations (in feature level). In other words, they predict and explain while recommendation, why an item is recommended or not. They experiment their system on datasets in real-world and prove the superiority of their algorithm in both top-K recommendation and prediction problems, by measuring how influential their recommendations are.

(Yuan, Murukannaiah, Zhang, & Singh, 2014) focus on the link prediction problem in social media and combine users' sentiments and social relationships to analyze the sentiment homophily while link prediction. They introduce a group of opinion-based features which assist while prediction of the likelihood that two users are friends (i.e. mutually followed), based on the users' opinions about topics of their mutual-interest. Their experiments on their utilized Twitter dataset prove the superiority of their method to the state-of-the-art.

(Trevisiol, Chiarandini, & Baeza-Yates, 2014) focus on the menu recommendation problem, that is, the restaurant recommendations for the consumer meals, based on the chosen restaurant. First, they extract a set of food words from the textual reviews of users. Then, by means of Sentiment Analysis techniques they provide the required user-food opinion matrix. Then, they utilize an *a priori* algorithm alongside a number of recommendation algorithms for recommendation of the menus (what they call combination of dishes).

(Rosa, Rodríguez, & Bressan, 2015) focus on providing a "sentiment intensity metric" by combining a Lexicon-based sentiment metric and a factor of correction, functioning based on the profile of users. Their approach for correction factor discovery is using laboratory-conducted subjective tests. Then, their correction factor is utilized for sentiment intensity adjustment. For experiments, they utilize a mobile-based music recommender system and provide 91% of users' satisfaction.

(Dimah H. Alahmadi & Zeng, 2015) focus on a hybrid approach in utilization of textual ratings for microblog-customized recommendation. They rely on this hypothesis that the choices of users are highly influenced by their trusted friends as well as their opinions. Their recommendation systems consider three steps for recommendation: (a) computing the implicit trusts between friends by means of their communications. (b) creating a user-target opinion matrix by means of applying Sentiment Analysis techniques on the micro-reviews. Considering the two previously proposed steps, measuring the degree of impact of the trust level between two friends by means of Machine Learning regression algorithms such as support-vector-regression and linear-regression. Their experiments on the considered Twitter-extracted dataset prove the effectiveness of their method. In a next study of the same authors (Dimah Hussain Alahmadi & Zeng, 2016), they progress their idea by utilization of genetic algorithms for trust relation extraction.

(Wu & Ester, 2015) consider estimation of a personalized version of opinion polarities by means of aspect-based opinion mining techniques. They also provide a unified probabilistic model (the so-called FLAME) that uses a combination of the CF potency as well as aspect-based opinion mining; the prediction task in FLAME is done by "collective intelligence." Their experiments prove the superiority of FLAME to the state-of-the-art counterparts in both of the aspect identification and aspect rating prediction tasks.

(J. Sun, Wang, Cheng, & Fu, 2015) focus on item recommendation in social media websites (e.g. YouTube and Flicker). They present a new sentiment-aware framework for social media item recommendation. They take advantage of the extracted opinion feedback as well as the one-class CF for improvement of the recommendation performance. Their real-world experiments on social media prove the better efficiency of their method over the state-of-the-art.

(J.-D. Zhang, Chow, & Zheng, 2015) focus on "location-based Social Networks" and present an opinion-extracted Point-Of-Interest (POI) framework for recommendation (the so-called ORec). ORec detects the opinion of the textual reviews and then integrates it with the check-in data of customers. In this regard, they propose a supervised aspect-based approach for opinion mining. Moreover, they propose an algorithm for fusing polarities and social links. Their experiments on the utilized two large-scale datasets prove the improvement of the opinion mining as well as the recommendation tasks, over the state-of-the-art counterparts, thanks to the mentioned fusion.

(Muhammad, Lawlor, Rafter, & Smyth, 2015), in addition to taking the mined sentiments of users into account, while the recommendation ranking processes, they also provide reasoning insights to users for their choice by means of their analyzed/predicted sentiments. Their experiments on a TripAdvisor-extracted hotel dataset proves the effectiveness of their method.

(Xu Chen, Qin, Zhang, & Xu, 2016) propose a "tensor matrix factorization" algorithm, to be combined by CF for the purpose of Sentiment Prediction. Their experiments on the two considered datasets prove the performance improvement in capturing the favorite features/items of users, meaningfully.

(Lou, Zhao, Qian, Wang, & Hou, 2016) address location Recommender Systems and correspondingly sentimental attributes of location. They present an opinion-based method for mining "Point Of Interest or POI." By means of the main sentimental attributes they make the POI recommendations. Their experiments on Sina Weibo dataset prove the effectiveness of their method.

(H. Li, Cui, Shen, & Ma, 2016) focus on recommendation in online media sharing websites. They present a joint data utilization strategy. They borrow the textual information required for Sentiment Prediction from microblogs. By this approach, they partly enrich the information lack existing in the Sentiment Prediction, by means of microblogs such as Twitter.

(L. Chen & Wang, 2017)(L. Chen & Wang, 2013) focus on the problem of recommending high risk products such as expensive products that are less sold and therefore less rated. In the former study (L. Chen & Wang, 2013), they present a new clustering method based on "Latent Class Regression Model or LCRM" that, in the same time, considers ratings and feature-level opinion values for prediction of the homogeneity of users' preferences. They provide connections between the active users and the user clusters considering their preferences and inter-relevance. Their experiments on the two dealt datasets prove the performance superiority of their method to the state-of-the-art. In the latter study (L. Chen & Wang, 2017), they assist the users in providing an understanding to product space for a better formulation of product preferences of users. They also take the feature sentiments of users into account and combine them with the existing static specifications. Their experiments prove increment of the knowledge of users about the products, their certainty on the preferences, and their purchase intention.

(Boris Galitsky, 2017) adopts a pure Text Mining approach and, firstly, studies the "Text Relevance Assessment" based on "Syntactic Generalization or SG." This task is done by finding a common maximal sub-tree of two parse trees and calculating the relevance of them, as a pair, comprised of two chunks of the text. The superiority of his more accurate relevance measure than the keyword-based analysis is: His method can consider the sentences at phrase level, as well. He also maps the other available metadata in online Social Networks such as Facebook Likes (by means of reasoning techniques) to the extracted relevant categories. He applies SG to personalization and recommendation tasks. Although he does not directly create a Sentiment Prediction system, applying SG on recommendation inspires Sentiment Prediction as an extension of his research.

(Feltoni Gurini, Gasparetti, Micarelli, & Sansonetti, 2018)(Gurini, Gasparetti, Micarelli, & Sansonetti, 2013) focus on the user-recommendation problem in Twitter and analyze the hypothesis that "it is possible that users share similar interests while having different sentiments on them. In their former study (Gurini et al., 2013), they assume that including the sentiment information of users is necessary for enhancing the recommendation quality in Social Network users. They propose a new weighting function (the so-called sentiment volume objectivity or SVO) which integrates the interests and sentiments of users. Thus, they provide richer profiles for users to be employed in the mentioned Recommender Systems. Their comparisons on the utilized Twitter dataset illustrates the superiority of their method to the state-of-the-art. In their latter study, they (Feltoni Gurini et al., 2018) extend their study by means of a 3-dimensional space including sentiment, volume, and objectivity of the textual reviews of users, toward which, they utilize a three-dimensional matrix factorization technique.

(Sharma & Bedi, 2018) focus on the problem of Twitter Hashtag recommendation. In their method, firstly, by means of applying the Sentiment Analysis techniques on a Twitter-extracted dataset, the tweets are categorized and the community detection is accomplished, accordingly. They suggest an idea similar to TF-IDF, called "Hashtag Frequency – Inverse Community Frequency (HF-ICF)." Then, by means of CF, they compute the corresponding relevance scores, as the final step. Their experiments on the studied Twitter-extracted dataset prove the efficiency of their method.

(Tewari & Barman, 2018) address the problem of high-recall/low-precision in Recommender Systems, by Sentiment Prediction approach. They take advantage of opinion mining, alongside, collaborative- and content-based- filtering as well as matrix factorization. Thanks to taking the inclination of users to the items (based on the mined opinions and …) into account and therefore detecting the special taste of each user, the experiments of their proposed Recommender Systems prove a high recommendation accuracy.

The interested reader is referred to the survey paper of (L. Chen, Chen, & Wang, 2015) in which they provide a taxonomy of the research studies, conducted on review-based Recommender Systems. In the mentioned survey, they categorize the conducted studies into the two sets of "review-based user profile building" and "review-based product profile building." The former (review-based user profile building) is more related to the Sentiment Prediction systems and the latter (review-based product profile building) is more related to Content-based filtering systems. They, also, suggest many future studies based on their survey.

**Progress behind the state-of-the-art**

As you see in the reviewed papers, they mainly address either "applications" of Sentiment Prediction or the ones which address the "data sparsity," they focus on enriching the data by means of including the textual reviews of users into account, and no study focuses on how to improve the accuracy/efficiency by means of Uncertainty-handling approaches, so that having the existing necessary information for a Sentiment Prediction system (i.e. textual data in Social Networks) they can improve the efficiency/accuracy.

We claim that Uncertainty-handling methods can play important roles in three phases for Sentiment Prediction.

(1) Providing Uncertainty-handling Natural Language Processing (NLP) platforms can provide the potency of better understanding in every Text Mining application including Sentiment Analysis.

(2) Providing specific techniques for Uncertainty-handling Sentiment Analysis techniques, including Uncertainty-handling text classifiers can provide a stronger platform the uncertainty-bearing circumstances.

(3) Providing Uncertainty-handling Recommender Systems can improve the accuracy of such systems even in the presence of uncertainty; especially, the uncertainties that occur in Social Network extracted data (what we define as the cold-co-start problem in section 3.6).

The related literature to the abovementioned steps are discussed in the following subsections. Considering the Uncertainty-handling platform that we should create for Text Mining, we start by WordNet and the existing similar lexical databases as well as the fuzzy approaches which already exist for WordNet-like Lexical Databases (WLDs).

### 1.1.2 WordNet and Lexical Databases

(Miller, Beckwith, Fellbaum, Gross, & Miller, 1990) propose WordNet (Miller, 1995)(Fellbaum, 1998), a lexical database for the English language that groups English words into synonym sets, called synsets[2]. From there on, based on the WordNet structure, other lexical databases were also proposed for different languages (Bond & Paik, 2012)(Vossen, 1998)(Vossen, 2004) that collect synsets of their corresponding languages, as it is done in WordNet. We call these lexical databases under the umbrella-term WLD[3]. WLDs have a wide variety of applications in NLP (Erik Cambria & White, 2014)(Gangemi et al., 2013)(Gangemi, Presutti, & Reforgiato Recupero, 2014)(Wei, Zhou, Chang, Lu, & Bao, 2015)(Reforgiato Recupero, Presutti, Consoli, Gangemi, & Nuzzolese, 2015) (Consoli & Reforgiato Recupero, 2015), Knowledge Engineering (Yan, Wang, Cheng, Gao, & Zhou, 2018)(Presutti, Draicchio, & Gangemi, 2012)(Gangemi, 2013)(Crawford, Gingerich, & Eliasmith, 2015)(Ivasic-Kos, Pobar, & Ribaric, 2016), and Ontology Engineering (Simperl & Luczak-Rösch, 2014)(F. Lin & Sandkuhl, 2008)(Gangemi et al., 2012)(Zablith et al., 2015)(Madalli, Sulochana, & Singh, 2016)(Bimson, Hull, & Nieten, 2016).

However, in WLDs, all of the members of a synset are supposed to belong to a synset with the same degree and convey the meaning of that synset at the same level. In other words, WLDs assume synsets to be crisp (non-fuzzy) sets. But this simple assumption does not always properly model the complex nature of meaning in natural languages. For example, let’s consider the following synset of WordNet: *Synset(‘flower.n.02’): {flower, bloom, blossom};* it contains the

[2] It additionally providing short definitions and usage examples and records a number of relations among these synsets and their members.

[3] WordNet-like Lexical Databases

words that potentially (as one of their senses) stand for "reproductive organ of angiosperm plants especially one having showy or colorful parts" (the illustrative-definition of each synset is proposed in WordNet).

Before proceeding with the mentioned issue, it is worthy to introduce the concept of a "lemma" and the concept of a "wordsense," in WLDs: (1) Each word disregarding its various potential senses is called a "lemma". For example, "bloom" disregarding the sense for which it can stand is considered a lemma. It is also the case for all of the words of a dictionary. (2) A specific sense of a lemma that is logically a member of one specific synset, is called a wordsense. For example, the above-mentioned sense of the lemma "bloom" is called a wordsense[4].

Usually, the lemmas (e.g. flower, bloom …) related to the wordsenses of a synset (e.g. Synset('flower.n.02')), are not equally compatible with the meaning (definition) of the synset, and each of them can have a different degree of compatibility. Therefore, the concept of fuzzy synsets is proposed. Since 2005, some research studies are being conducted, studying on fuzzy synsets and the resulting WLDs.

(Velldal, 2005), without using the term "fuzzy synset" (even without using the term "synset"), proposes an algorithm for creating fuzzy semantic classes[5] (i.e. synsets) and states that "different words can represent more or less typical instances of a given concept. Some words may represent clear-cut instances of a given category, while others represent peripheral or border-line cases." In order to illustrate such categories, Velldal considers them as fuzzy sets and utilizes a fuzzy clustering algorithm for assigning membership values to the corresponding members, and proposes a Norwegian fuzzy WLD. (L Borin & Forsberg, 2010b) who coined the term "fuzzy synsets," view them from a pure linguistics point of view, and base them on "synonymy avoidance" (Hurford, 2003) which implies that two wordsenses of a human language are very unlikely to exactly stand for a same meaning/definition. Consequently, a dictionary that fundamentally assumes synonymy (linguistically speaking) cannot fairly project human lexical knowledge. In the mentioned study, (L Borin & Forsberg, 2010b)(L Borin & Forsberg, 2010a) utilize Synlex (People's synonym lexicon (Kann & Rosell, 2005) that contains synonymy[6] degree of word-pairs, provided by crowdsourcing) as well as SALDO[7] (Lars Borin, 2005)(Lars Borin & Forsberg, 2009) to present an algorithm to create fuzzy synsets for the Swedish language. In 2011, (Gonçalo Oliveira & Gomes, 2011) are the second research group which looks at fuzzy synsets from a linguistics point of view expressing that "from a linguistic point of view, word senses are not discrete and cannot be separated with clear boundaries (Kilgarriff, 1997) (Hirst, 2009)[8]… Sense division in dictionaries and lexical resources is most of the times artificial…" They propose an algorithm for

---

[4] Each lemma can have several wordsenses. In other words, each lemma can be a member of more than one synset.

[5] He applies his algorithm on Norwegian language.

[6] For more information about synonymity please refer to (Osgood, 1952)

[7] A full-scale Swedish lexical-semantic resource with non-classical, associative relations among word and multiword senses, identified by persistent formal identifiers.

[8] the original reference was older version of (Hirst, 2009)

generating fuzzy synsets and apply it to the Portuguese language, producing a Portuguese fuzzy WLD.

However, none of the mentioned directed studies towards a fuzzy understanding of synsets propose any approach to produce a fuzzy version of the crisp synsets in the existing WLDs (e.g. WordNet, EuroWordNet, Arabic WordNet, IndoWordNet …). In other words, in the mentioned few studies, the synsets either are not predefined and can be determined only after running the proposed algorithm (i.e. fuzzy synsets are the output of clustering (Velldal, 2005)(Gonçalo Oliveira & Gomes, 2011)), or there exists a lexical database (SALDO in (L Borin & Forsberg, 2010b); yet not WordNet-like), which is modified by the algorithm so that its synsets are not the fuzzy version of the previous synsets.

The aforementioned studies have not received much attention by the Text Mining community, whose research efforts utilize platforms defined on already existing WLDs. The community is reluctant to change its foundational platforms and migrate to, although useful, different and new platforms. In our opinion, this is the reason why fuzzy synsets are kept almost isolated in the field of Text Mining. However, as mentioned, no research[9] has solved this shortcoming.

Albeit from their fuzzy-ontology viewpoint, (León Aráuz, Pilar Gómez-Romero & Bobillo, 2012) mention this point in their study: “extending WordNet and EuroWordNet to include imprecise knowledge requires a considerable effort to define synset membership, similarity and equivalence degrees;” however, they do not propose any approach/solution in that study. In this dissertation, we propose an idea for overcoming this drawback.

In section 3.1, we present an algorithm to be able to assign membership functions for predefined synsets of any language, given a large corpus of documents of that language and a Word Sense Disambiguation (WSD)[10] as input. Then, we apply the algorithm to the English language, using the Open American National Corpus (OANC) and the well-known graph-based WSD system, named UKB, and construct the fuzzy version of WordNet, accessible online. To validate the obtained fuzzy synsets, we introduce the “Bag of fuzzy Synsets or BoFS” and “Bag of fuzzy wordsenses or BoFWS” auxiliary Text Mining models, which extend two existing Text Mining models in order to be able to operate on Fuzzy WordNets (FWN). We use these models in a Sentiment Analysis task to evaluate them against their crisp versions, which use standard WordNet. FWN is expected to outperform the crisp version of WordNet thanks to its extra information.

[9] There is a similar concept not to be confused with this scope, that is, “graded word sense assignment” (Erk & McCarthy, 2009)(Erk, McCarthy, & Gaylord, 2009)(Erk, 2010)(Erk, McCarthy, & Gaylord, 2013) that addresses fine-grained graded versions of wordsenses of lemmas whereas we are addressing fuzzy synsets (fine-grained graded versions of wordsenses of synsets).

[10] In cognitive and computational linguistics, Word Sense Disambiguation (WSD) is an open problem belonging to ontology and natural language processing. Considering a word in a sentence, WSD identifies which of its senses is used in that sentence (for multi-sense words) (Weaver, 1955).

### 1.1.3 Text Mining Models

In the previous subsection, we addressed the WordNet and the other crisp or fuzzy WLDs. However, for taking advantage of such lexical databases, including what we reviewed before or what we propose later, we have to use some Text Mining models. Therefore, in this subsection, we provide a review of a number of well-known Text Mining models.

However, as mentioned before and as you will see later, in section 3.3, we propose two auxiliary fuzzy models for Text Mining, based on the "fuzzy synsets" concept. Then, in section 4.1, we apply the proposed auxiliary Text Mining models to Sentiment Analysis to check if they improve the accuracy, in comparison with their crisp version. Therefore, here, we suffice to review one of the evolution lines in Text Mining models and the upgraded version of some of them, making them able to cope with fuzzy synsets data, proposed in the next chapter.

**Bag of Words (BoW).** The most straightforward, and well-known model for Text Mining is unigram (n-gram when n=1), that is known as BoW (Yin Zhang, Jin, & Zhou, 2010). The BoW model considers each word as a feature. For example, consider the following tweet.

*I think that we are supposing this plant incorrectly as a flower; only because this part of the plant is incorrectly supposed as bloom/blossom.*

Suppose that we, firstly, remove the punctuations and stop words (e.g. "I," "that," "we," "are," …) and lemmatize[11] the remained words (some common techniques, adopted before utilizing BoW). Then, BoW considers the words of the above tweet as document features and outputs the following model.

*{(think,1), (suppose,2), (plant,2), (incorrectly,2), (flower,1), (part,1), (bloom,1), (blossom,1)}*

**Bag of synsets (BoS).** BoS is an extension of BoW model, which instead of considering each word as a feature, considers the related synset of each word as a feature. (Whaley & Aslam, 1999) for the first time proposed this model (albeit naming it as "Bag of Concepts" and considering each synset as a concept). (Manjula, Aghila, & Geetha, 2003), for the first time used a similar term to BoS, that is, "bag of synset of words." This idea was followed in the next years by well-known research studies in this field such as the studies of (Semeraro, Lops, & Degemmis, 2005), (Ye & Baldwin, 2006), (Basile, Degemmis, Gentile, Lops, & Semeraro, 2007), (Lops, Degemmis, & Semeraro, 2007), (Semeraro, Degemmis, Lops, & Basile, 2007) (de Gemmis, Lops, Semeraro, & Basile, 2008), and one of the most known studies of this field, SentiWordNet 3.0 (Baccianella, Esuli, & Sebastiani, 2010) which uses a similar idea (to BoS) in Step 2 of SentiWordNet construction algorithm. The BoS idea is also applied to Text Mining-related research studies in other languages as well (Lops et al., 2010)(Pouramini & Minaei-Bidgoli, 2016). However, in the recent years, the research attention to this model has decreased under the effect of the Bag of Concepts model, which is explained in the following.

[11] Mapping the words to the fittest lemma to them.

**Table 1.2. List of some synsets with hypothetical synsets ID.**

| Synset ID | Word senses |
|---|---|
| **S1** | think, opine, suppose, imagine, reckon, guess |
| **S2** | plant, flora, plant life |
| **S3** | flower, bloom, blossom |
| **S4** | falsely, incorrectly |
| **S5** | merely, simply, just, only, but |
| **S6** | part, portion |

For further illustration of BoS model, we present an example. Consider the synsets, represented in Table 1.2 The BoS model of the following tweet

*I think that we are supposing this plant incorrectly as a flower; only because this part of the plant is incorrectly supposed as bloom/blossom.*

would be the following set (based on Table 1.2 info)

*{(S1, 3), (S2, 2), (S3, 3), (S4, 2), (S5, 1), (S6, 1)}.*

**Bag of Concepts.** The Bag of Concepts model for Text Mining can be considered as an extension of BoW or BoS, which is proposed, in parallel (in the same year) with BOS. In this model, instead of gathering words into different sets (BoW) or gathering synonyms into different sets (BoS), more general sets are used to gather lexical objects together. These more general sets conceptually differ, as they have been introduced in different research studies, multiple times.

For the first time (Kiryakov & Simov, 1999) proposed this model, as a model packing (bagging) the hypernyms and hyponyms of a word, while considering each synset. From then on, this idea has been followed, applied, or redefined by hundreds of studies (dozens of them in the recent two years) among which we enumerate the most well-known research studies. (Sahlgren & Cöster, 2004) address Bag of Concepts extending the "concept" to include synonyms of words and latent dimensions of them (addressing dimension reduction). (Navigli, 2006) proposes a model that after including the other wordsenses of the (corresponding) synset of a word bags them with their direct hypernyms as well as labeled domains (specified in the WordNet and Oxford Dictionary of English). (T. Li, Mei, Kweon, & Hua, 2011) proposes a method which groups conceptually relational words, and calls the output as Bag of Concepts. (E. Cambria & Hussain, 2012) propose a similar model called Small Bag of Concepts (SBoC), to include the cognitive and affective information associated with the input text. In 2014, Cambria (the presenter of SBoC) (Erik Cambria & White, 2014) whose review study focuses on evolution of NLP research according to the three main paradigms of BoW, Bag of Concepts, and a next generation that they introduce as Bag of Narratives (we call BoN) model, explain "how and why NLP research has been gradually shifting from lexical semantics to compositional semantics" and recommend the next generation of NLP technology, as BoN, which is predicted to be engaged with narratives. (Poria, Cambria, Gelbukh, Bisio, & Hussain, 2015) utilize Bag of Concepts in Sentiment Analysis, besides some other techniques, to improve the Sentiment Analysis accuracy. (Kalloubi, Nfaoui, & El Beqqali, 2016) propose an extension of this model as the "graph of concepts." Also, (E. Cambria, 2016) proposes an algorithm based on Bag of Concepts, as an input to either Deep Learning system or Sentic Patterns for Sentiment Analysis. (H. K. Kim, Kim, & Cho, 2017) propose another version of Bag of Concepts by bagging the clustering distributed representation of words generated from word2vec.

Considering that Bag of Concepts, since its appearance, has gradually become an umbrella term for different approaches to bagging concepts, (for brevity) we avoid detailed illustration of them one by one.

After addressing the necessary handling platforms for NLP, it is the turn to address Uncertainty-handling in the second phase of Sentiment Prediction, that is, Sentiment Analysis.

#### 1.1.4 Sentiment Analysis and Fuzzy Synsets

After getting introduced to the fuzzy and crisp lexical databases, and seeing how such WLDs are utilized in state-of-the-art as Text Mining models, we should review how they are utilized in Sentiment Analysis.

Sentiment Analysis has received broad attention in the recent decade. However, extracting sentiment information from unstructured text data is a multi-disciplinary problem, because sentiments can be expressed in numerous forms and combinations where it might be difficult to find any sort of regular behavior.

From one point of view, the majority of approaches to Sentiment Analysis are divided into two categories: "Machine Learning approach," and "Lexicon-based approach" [21]. The former utilizes Machine Learning algorithms mainly to solve Sentiment Analysis as a regular text classification problem using syntactic and/or linguistic features, whereas the latter basically utilizes an opinion lexicon (i.e. a list of opinion words and phrases), and a set of rules for determining the opinions orientations in a sentence and also considers opinion shifters and but-clauses [20]. The former provides maximum accuracy whereas the latter provides better generality [26]. However, the Lexicon-based approach is more often used recently [21]. Lexicon-based approach (utilizing opinion lexicons [21] as well as generating them [20] for Sentiment Analysis purposes) is further divided into dictionary-based and corpus-based categories. In the former, the domain of the opinion-words is as wide as the domain of a complete dictionary, whereas in the latter the domain is limited to those included in the analyzed corpus (corpora). The corpus-based approach, alone, is not as effective (for identifying all opinion words) as the dictionary-based approach because it is hard to prepare a huge corpus to cover all the English words. Conversely, the corpus-based approach has the major advantage of finding domain- and context-specific opinion words and their orientations using a domain corpus [20].

In brief, based on the [20] [21] categorizations, Lexicon-based Sentiment Analysis approaches are categorized into dictionary-based and Corpus-based the latter of which has the sub-approaches of Statistical, Semantic, and NLP-based. Synset-based Lexical databases such as WordNet [14] that organize words of a language in synonym groups -called synsets- are being utilized by dictionary-based approach as well as semantic sub-approach of the corpus-based approach in Sentiment

Analysis, several of which take advantage of the synset-based opinion lexicons such as SentiWordNet [13][2]. (SentiWordNet is a lexical resource in which each WordNet synset is associated to Objective, Positive, and Negative values in the continuous interval [0,1] for describing how objective, positive and negative the terms contained in that synset are). However, in the prevalent Synset-based Lexical databases such as WordNet, all the members of a synset are supposed to belong to a synset with the same degree and convey the meaning of that synset at the same level. In other words, such Synset-based Lexical databases assume synsets to be crisp and non-fuzzy sets. However, this simple assumption does not always properly model the complex nature of "meaning" in natural languages. This fact might be considered as a drawback for such Synset-based Lexical databases.

The drawback of crisp synsets also permeates synset-based Sentiment Analysis methods including Synset-based-Lexical-database-utilizing Lexicon-based Sentiment Analysis methods[12], because they use the same crisp synsets. For instance, SentiWordNet 3.0 assigns a sentiment pair (positive, negative) to each of the WordNet synsets and assumes all of its wordsenses to have the same sentiment load. Such Lexicon-based Sentiment Analysis methods can be upgraded by fuzzy versions of their utilized crisp synsets, discriminating between wordsenses of one fuzzy synset, how much each of its wordsenses contains the sentiment load of that fuzzy synset, and thus, assigning a low (high) semantic load to low (high) membership-graded wordsenses of that synset. For example, the Synset('run_into.v.01') is annotated as (+0, -0.25) in SentiWordNet 3.0. Suppose the fuzzy version of this synset to be {(run_into, 1.0), (encounter, 0.4)}. Then, considering that the wordsense 'encounter' is not fully compatible with this synset (40% compatible), it is not precise to assign (+0, -0.25) (the sentiment load of that synset) to this wordsense in Sentiment Analysis process. Its sentiment load does not inherit all the negativity of its synset; yet, it might inherit sentiment of other synsets to which it is compatible (e.g. 'run_into' is also wordsense of Synset('run_into.v.02'), Synset('hit.v.02'), and Synset('meet.v.01')) regarding which upgraded Sentiment Analysis methods shall use "graded wordsense assignment" [12][11] and/or fuzzy WSD [27][10] and specify the grade by which 'run_into' belongs to the other 3 synsets and then aggregate the semantic load of all those synsets based on the membership (intra-synset) and grade (inter-synset) of 'run_into' to each of those synsets. Then, the aggregated value would be more informative than simply using (+0.0, -0.25) for it, inheriting from its synset. For the mentioned upgrade in Synset-based-Lexical-database-utilizing Lexicon-based Sentiment Analysis methods, we require the algorithm that is going to be introduced in section 3.1) for converting the synsets of the existing Synset-based Lexical databases to a fuzzy version.

#### 1.1.5 Machine-learning-based Text Classification and Authorship Attribution

As mentioned, one of the successful approaches to Sentiment Analysis is the Machine Learning approach. In this subsection, we discuss this approach and review its important sub-approaches literature in the state-of-the-art. Nevertheless, because one of the funding providers of this dissertation (IDENTITY H2020 European project) asked us to apply our research studies to the

[12] There are also other synset-based Sentiment Analysis methods to which we do not address in this short introduction.

Authorship Attribution field, we first provide a literature review on Authorship Attribution and then provide an introduction to the Machine Learning approach, utilized there. However, please note that as the nature of the text classification is quite similar in both of the fields; thus, we can simply apply the classifiers of one filed to another just by changing the appropriate Text Mining model for that field.

Therefore, in this subsection, we provide a slight introduction to what Authorship Attribution is and then focus on the Machine Learning approach to this field of science. We would like to emphasize that since the utilized techniques in the Machine Learning approach to Sentiment Analysis and Authorship Attribution are almost the same, these approaches can be utilized in both of the reviewed scientific fields.

Now, let us go through an introduction to what Authorship Attribution is.

The treatment of learner machines with similar structures differs based on the dataset by which they are trained. This difference is quite trivial and of no relevance, because those machines are disposable and identity-less, and if they are expected to treat differently, they can be trained again on another training set. Contrariwise, while human's brain architecture and its neural map are very similar across human beings, their different treatment matters very much in the field of Biometrics. These differences may be so delicate that they are not directly sensible, but they can still be detected by Biometrics machines.

One of the most evident examples of such habits is walking, by which, the identity of a human is detected (the so-called Gait Recognition (Jiwen Lu, Wang, & Moulin, 2014)). Habit-like treatments –after consolidation– become stable and almost unchangeable, and insistence of brain neurons to reveal their real treatment becomes hard-to-control (Evans & Stanovich, 2013)(Wood & Rünger, 2016) because those neural organs are almost autonomous and uncontrollable.

### *Stylome*

Other examples of the mentioned habit-like treatments are speaking and writing. Expressing emotions and information from mind to words is a skill, and correspondingly, the choice of words, the order of them, or the grammar by which they are connected differ from subject to subject. Stylome is defined as a fingerprint of writing (or similarly, the fingerprint of speech). Stylome can be used for identifying people. The more texts a writer writes, the harder his fingerprints can be wiped out, and the harder his writing traits can be avoided (Zhao & Zobel, 2007). This makes the writer addicted to his own stylome. Assuming that a writer has a wide vocabulary and he writes in different contexts, potentially different word-choices can be selected. However, experiments prove that in practice this does not happen, and writers stick to their idiosyncratic word-choice patterns (El Manar El Bouanani & Kassou, 2014). This fact has made "stylome" the key concept discussed in Stylometry for attributing authors (Ayogu & Olutayo, 2016), as one of the main problems of Forensic Document Examination (J. A. Lewis, 2014).

### *Stylometry and Authorship Attribution*

Stylometry (as the name implies) is the science studying the relation between stylomes and subject identity. This can attribute authors' identity by analyzing their written contents (*Authorship*

*Attribution* (Juola, 2008)), verifying if the document is written by a specific author (*authorship verification* (Luyckx & Daelemans, 2008)), or profiling the author of a document based on demographics such as gender, age, native language, personality, and etc. (*authorship profiling* (Palomino-Garibay et al., 2015)).

The most popular subtask of Stylometry is Authorship Attribution (Efstathios Stamatatos, 2009). Computational Authorship Attribution (Chaski, 2013) can either follow a traditional literary approach or a Machine Learning approach. In the current century, Machine Learning approaches have been developed by means of the artificial intelligence, information retrieval, and NLP techniques (Efstathios Stamatatos, 2009). Machine Learning methods require two key phases: Feature Extraction and Classification (Neal et al., 2017). We address these two phases in the following.

### ***Feature Extraction Phase***

In the feature extraction phase, text features are utilized as stylomes, and correspondingly, the documents are represented by data structures of the chosen features. For example, if words (unigrams) are selected as stylomes, vectors or bags of words are used for document representation. Then, for the classification phase, a sufficient number of those data structures which represent text documents of different authors is required to train a classifier. Finally, the authorship can be determined by applying the classifier to new data (Gavrilova & Yampolskiy, 2011).

The mentioned stylometric features can be categorized into the following main classes: lexical features (e.g., word n-gram frequency (Raghavan, Kovashka, & Mooney, 2010)), character features (e.g., character n-gram frequency (E Stamatatos, 2013)), syntactic features (e.g., part-of-speech tag frequency (Zeldes & Schroeder, 2015)), bag of fuzzy (Hossayni, Akbarzadeh-T, Reforgiato Recupero, Gangemi, & de la Rosa i Esteva, 2016) and interval fuzzy (Hossayni, Rajati, et al., 2016) synsets, features, semantic features (e.g., semantic dependencies (Grieve, 2007)), and application-specific features (e.g., specific word frequency (Das, Tasmim, & Ismail, 2016)). The survey papers (Grieve, 2007) and (Efstathios Stamatatos, 2009) provide detailed descriptions on each of the mentioned categories.

Among the mentioned categories, the most effective measures are lexical and character features (Efstathios Stamatatos, 2009)(Sidorov et al., 2014). Character n-grams outperform the other feature types when confronted with sets of up to tens of candidate authors. For the larger author sets, lexical features show more robustness (Luyckx, 2011). Recently, (Sidorov et al., 2014) introduced a new lexical feature named "syntactic n-gram" (sn-gram) that models text documents by means of parsing them. It adopts a different manner for specifying what elements are considered as neighbors, in comparison to standard n-grams (Figure 1.1).

In sn-grams, the neighbors are taken by following syntactic relations in syntactic parse trees whereas n-grams are formed as they appear in texts. sn-grams are shown to be the most effective model for the case in which the written documents are large (e.g. books vs. tweets). In such cases,

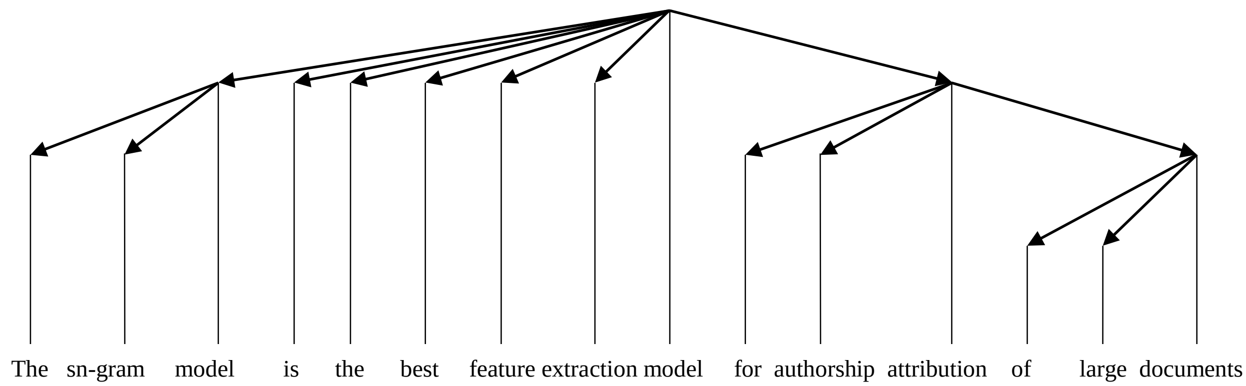



**Figure 1.1. Traversing an example sentence by the sn-gram model. Each arrow is one bigram.**

sn-grams are proven to outperform the word n-grams, Part Of Speech (POS) tags, and characters (Sidorov et al., 2014).

### *Classification*

In general, four main approaches are adopted for the classification phase of machine-learning-based Authorship Attribution (Neal et al., 2017): (1) pattern recognition methods[13], (2) probabilistic methods, (3) distance-based methods, and (4) rule-based methods[14]. A fifth category may also be considered which includes the hybrid approaches.

Examples of "pattern recognition" category include support vector machines (SVM) (Diederich et al., 2000)(Sidorov et al., 2014)), neural networks (Savchenko, 2013), deep learning (Rhodes, 2015), backpropagation (X. Yang, Xu, Li, Guo, & Zhang, 2017), prediction by partial matching (Rocha et al., 2017), discriminant analysis (E Stamatatos, Fakotakis, & Kokkinakis, 2001) and etc. Instances of the "probabilistic" category include probabilistic context-free grammars (Raghavan et al., 2010), conditional random field models (Elming, Jakob and Johannsen, Anders and Klerke, Sigrid and Lapponi, Emanuele and Alonso, Hector Martinez and Sogaard, 2013), and Bayesian and Naïve Bayes classifiers. The "distance based" category includes k-nearest-neighbors (Halvani, Steinebach, & Zimmermann, 2013), Dissimilarity measure (Segarra, Eisen, & Ribeiro, 2015), common n-grams proportion (Brocardo, Traore, Saad, & Woungang, 2013), nearest shrunken centroid (Schaalje, Fields, Roper, & Snow, 2011) and etc. Finally, rule-based classifier examples include Dominance-Based Rough Set Approach (Stańczyk, 2015) and cascading rough set-based classifiers (Aslantürk, Sezer, Sever, & Raghavan, 2010).

Each of these classifiers has specific drawbacks and advantages and are recommended for special applications. Comparative studies such as (Jockers & Witten, 2010), Eder (Eder, 2015), (El Manar El Bouanani & Kassou, 2014), and (Neal et al., 2017) provide more detailed insights for the mentioned specifications.

[13] Neal et al. (Neal et al., 2017) utilize the term "Machine Learning" for naming this category. However, considering that Machine Learning is also as an umbrella term for all of the categories, we avoid using this term, here.

[14] Neal et al. (Neal et al., 2017) do not enumerate "rule based" systems but we believe rule-based systems are a (rather sparse but) independent category.

In this phase of this dissertation (mainly section 3.4), we focus on the classification phase and specifically on the uncertainty problem which is not well-addressed yet in Authorship Attribution. The next subsection is devoted to this problem.

#### *Uncertainty-handling in the classification phase*

The "uncertainty problem," or the process of handling the uncertainty existing in different problems or algorithms has upgraded thousands of techniques in different fields of engineering, especially in artificial intelligence (Rutkowski, 2013) and Forensics (Garnaev, Baykal-Gursoy, & Poor, 2014)(Garnaev & Trappe, 2016) including Machine Learning and classification problems (Bounhas, Ghasemi Hamed, Prade, Serrurier, & Mellouli, 2014). However, this is not well addressed in Authorship Attribution. There are a few studies on handling uncertainty in the feature phase, such as (Homem & Carvalho, 2011), (Stańczyk, 2013), or (Rovenchak, 2011). There are also a few Uncertainty-handling rule-based classifiers proposed in Authorship Attribution (Aslantürk et al., 2010)(Stańczyk, 2015). However, despite the fact that the pattern recognition approach is the most dominant approach in the classification phase, unfortunately to our best knowledge no study has been conducted on this scope. Moreover, the mentioned studies do not provide a comprehensive analysis of different cases of the uncertainty problem.

In section 3.4, we address three types of possible uncertainties in Authorship Attribution and visit and propose different Weibull-based Bayesian classifiers for handling them based on the probability theory and its counterpart, the possibility theory (Zadeh, 1978), for the addressed uncertain circumstances.

### 1.1.6 Collaborative Filtering, Cold-start problem, and Significance Weights

After introducing the Text Mining steps of a Sentiment Prediction, it is the turn to address the Recommender Systems and especially CF. As mentioned at the beginning, the Sentiment Prediction systems mainly rely on the CF methods. Such methods although powerful, yet, suffer from the drawback of vulnerability under sparsity and sparsity is quite expected in the Social Network extracted information, even after including the textual reviews.

Therefore, in this subsection, we firstly provide a slight review the Recommender Systems as the umbrella field of CF methods. Then, we provide a general overview of the CF methods and how they work. Thereafter, we focus on the cold-start problem that is the nearest discussed problem in CF to the mentioned drawback. We go, then, one step further and introduce a more delicate view to this problem by defining the cold-co-start problem (that normally occurs in the Social Network extracted data). After introducing the cold-co-start problem, because the approaches of the solving algorithms mainly rely on the SWs, we survey the already existing approaches for solving this problem. In the following, we start to review the mentioned topics in the state-of-the-art:

In everyday life, it is often difficult to choose an option without enough personal experience of the alternatives. We usually rely on recommendations from other people (Mourão, Rocha, Araújo, Meira, & Konstan, 2017) via miscellaneous means (hearing, recommendation letters, published reviews, etc.) (Rosaci & Sarné, 2013). Recommender Systems assist and augment this natural social process (Resnick & Varian, 1997)(Papadimitriou, Symeonidis, & Manolopoulos, 2012). They are software tools and techniques that provide suggestions for items to a user (Ricci, Rokach, & Shapira, 2004). The automatic suggestions are aimed at supporting the users in their various decision-making processes (Ricci et al., 2004). The interested reader is referred to the corresponding most popular surveys in (Burke, 2002)(Montaner, López, Rosa, & De La Rosa, 2003)(Adomavicius & Tuzhilin, 2005)(Gunawardana & Shani, 2009)(Bobadilla, Ortega, Hernando, & Gutiérrez, 2013)(Jie Lu, Wu, Mao, Wang, & Zhang, 2015).

One of the widely used recommending techniques is Collaborative Filtering (CF). "CF is the process of filtering or evaluating items through the opinions of other people. CF technology brings together the opinions of large interconnected communities on the web, supporting filtering of substantial quantities of data" (Schafer, Frankowski, Herlocker, & Sen, 2007). It consists of two phases: Similarity-Finding and Rating-Prediction. The former measures the similarity of users/entities and the latter, using the similarity information, predicts a rating for the user whom the prediction is performed for. Applications of CF, typically, involve very large datasets (Takács, Pilászy, Németh, & Tikk, 2009).

The CF algorithms are divided into Memory-based and Model-based algorithms (Breese, Heckerman, & Kadie, 1998) and their hybrid methods (X. Su & Khoshgoftaar, 2009). Memory-based CF preserves a database of the entire users' determined preferences for all of the items and (for each prediction) carries out some computations across the whole database (Pennock, Lawrence, & Giles, 2000). But Model-based algorithms, firstly, compile the preferences of users into a descriptive model and then recommendations are generated by making a plea to the model (Pennock et al., 2000). Each of the Memory-based CF, Model-based, and hybrid algorithms have some advantages and disadvantages (X. Su & Khoshgoftaar, 2009) and in different cases, the ideal choice may be different, based on the requirements of that case. One of the drawbacks of Memory-based CF is low-quality rating predictions in cold-start condition[15] (i.e. the condition in which a user or an item does not have any or has only a small number of recorded ratings (Schein, Popescul, Ungar, & Pennock, 2002)(D. Maltz & Ehrlich, 1995)(Ahn, 2008). In this subsection, we focus on a special case of the cold-start problem. Before addressing the problem, we provide some additional explanation about Memory-based CF.

Memory-based CF utilizes the entire user-entity database to generate a prediction. Among the common strategies for opinion prediction, in Memory-based CF, many of them apply some variation of the neighborhood-based prediction algorithms (Jon Herlocker, Konstan, & Riedl, 2002). In these algorithms, the prediction would be the aggregation of the $N$-nearest ($|U|$) neighbors of the user $u$. In (Schafer et al., 2007) and (Adomavicius & Tuzhilin, 2005), the authors point out the following 3 strategies for this aggregation

[15] We found the first usage of this concept in scientific literature in 1994 (D. A. Maltz, 1994).

Strategy 1: $r_{u,i} = \frac{1}{N} \sum_{u' \in U} r_{u',i}$ ,

Strategy 2: $r_{u,i} = k \sum_{u' \in U} simil(u, u') \times r_{u',i}$ , Equation 2.1

Strategy 3: $r_{u,i} = \bar{r}_u + k \sum_{u' \in U} simil(u, u') \times \left(r_{u',i} - \bar{r}_{u'}\right)$,

In the Equation 2.1, $r_{u,j}$ represents the predicted opinion of the user $u$ about the item $i$, $U$ denotes the set of top $N$ users who are most similar to the user $u$; $k$ is a normalizing factor defined as $k = \frac{1}{\sum_{u' \in U} |simil(u,u')|}$ and $\bar{r}_{u'}$ is the average rating of the user $u'$ on all of the rated entities. In the state-of-the-art, the 3rd strategy is frequently used and provides acceptable results (Pirasteh, Hwang, & Jung, 2014)(H. Kwon, Kwon, & Hong, 2011)(Schelter, Boden, & Markl, 2012)(Choi & Suh, 2013).

There are a number of techniques for calculating the similarity (or dissimilarity) of $u$ and $u'$. There are some proven traditional similarity measures in Memory-based CF including Cosine similarity measure, Pearson and Spearman rank correlation coefficient (H.-J. Kwon & Hong, 2009)(Jl Herlocker & Konstan, 1999)(H.-F. Sun et al., 2012). The cosine-based approach (Adomavicius & Tuzhilin, 2005) defines the similarity between the users $u$ and $u'$ as

$$simil(u, u') = \frac{\sum_{i \in I_{u,u'}} r_{u,i} r_{u',i}}{\sqrt{\sum_{i \in I_{u,u'}} r_{u,i}^2} \sqrt{\sum_{i \in I_{u,u'}} r_{u',i}^2}}$$ Equation 2.2

where $I_{u,u'}$ is the set of entities rated by both users $u$ and $u'$. Also, the Pearson correlation approach (Jl Herlocker & Konstan, 1999) defines the similarity of the users $u$ and $u'$ as

$$simil(u, u') = \frac{\sum_{i \in I_{u,u'}} (r_{u,i} - \bar{r}_u)(r_{u',i} - \bar{r}_{u'})}{\sqrt{\sum_{i \in I_{u,u'}} \left(r_{u,i} - \bar{r}_u\right)^2 \sum_{i \in I_{u,u'}} \left(r_{u',i} - \bar{r}_{u'}\right)^2}}$$ Equation 2.3

### *The cold-start problem*

The traditional measures do not use the domain-dependent meanings of the data (ratings), particularly, in the case which the available data is not adequate, generally leading to the cold-start problem which refers to the situation in which “a user, an item, or the entire system is new” (Park, Pennock, Madani, Good, & DeCoste, 2006). In order to alleviate the cold-start problem in CF, different studies have been conducted with various approaches.

(Ahn, 2008) presents a novel heuristic similarity measure, based on the elaborated meanings of co-ratings. Its focus is on the improvement of the recommendation performance under cold-start conditions. In (H.-J. Kwon & Hong, 2009), the authors propose Moment Similarity of Random Variables, by considering the two user profiles ($u$ and $u'$) as two discrete random variables ($U$ and $U'$) and define the absolute value of their linear difference as a third random variable ($D_{U,U'}$)

and base Moment Similarity of Random Variables on the moment of $D_{U,U'}$. (Jamali & Ester, 2009) propose a random walk model combining the trust-based and the CF approach for the recommendation. (J.-H. Su, Wang, Hsiao, & Tseng, 2010) propose a novel recommender, FRSentiment Analysis (Fusion of Rough-Set and Average-category-rating), that integrates multiple contents and collaborative information for predicting user's preferences based on the fusion of Rough-Set and Average-category-rating. (C. C. Chen, Wan, Chung, & Sun, 2013) propose a cold-start recommendation method for the new user; it integrates a user model with trust and distrust networks to identify trustworthy users. The suggestions of these users are then aggregated to provide useful recommendations for cold-start new users. (C. Lin, Xie, Guan, Li, & Li, 2014) propose PRemiSE (a new Personalized news Recommendation framework via implicit Social Experts), which treats the sentiments of potential influencers on virtual Social Networks (extracted from implicit feedbacks) as auxiliary resources for the recommendation. Moreover, (Haifeng Liu, Hu, Mian, Tian, & Zhu, 2014) present a new user similarity model which considers, both, the local context information of user ratings and the global preference of user behavior, to reduce this problem. The interested user can refer the survey papers of (SUN, HE, & ZHANG, 2012), (Bobadilla et al., 2013), and (Son, 2016).

One pre-mentioned idea for this problem is applying a weight which assigns less value to the users with few common ratings and much value to those with many common ratings, the so-called "SW" (SW). Various SWs are introduced in the state-of-the-art. Most of the proposed SWs, are functions of $|u_i \cap u_j|$ which assign low weights for law values and high weights for high values. Normally, these SWs have a parameter by which the function is horizontally stretched (by default, SWs range is [0,1] and therefore there is no room for vertical stretching). However, the mentioned SWs have a fix y-intercept (e.g. 0 or 0.5) whereas, except some naïve assumptions that the researchers adopt, there is no reason for fixing the y-intercepts. It means that the mentioned SWs are generalizable to have a parameter for tuning y-intercept. This generalizability implies upgradability of SWs for more alleviation to the cold-start problem. However, proposing Generalized SWs (GSW), yields in the requirement to optimizing the respective parameters. Optimized GSWs result in more alleviation to the problem arisen in the situation which the number of co-rated items by two users are not significant enough for assigning a similarity value (the co-called cold-co-start problem). For this purpose, the most important requirement is selecting a suitable optimization algorithm.

### *A class of Meta-heuristic Optimization Algorithms*

Considering that the target (to be optimized) cost function, that is the average error of rating-estimation, stochastically depends on the rating data of real-world users, analytic and classic optimization algorithms are not adoptable for their optimization, and therefore, Meta-heuristic Optimization Algorithms (MOA) should be utilized. However, as we will elaborate in section 3.7, the selected MOA should be "global" with "low-population/iteration" and "linear time-complexity." But, there is no MOA in state-of-the-art, satisfying the three abovementioned requirements, at the same time. Thus, we have to propose such MOA.

Therefore, we propose a straightforward Surrogate Modelling (SM) fuzzy optimization algorithm (a sub-class of population-based MOAs[16]) that from the one hand, by utilizing Soft Computing techniques bypasses the local optimums (like other population-based MOAs) and from the other hand, requires rather few populations, and also, has linear time complexity in training phase, which makes it distinguishable from its counterparts. As a solid MOA, we expect the yielding optimization results (optimized GSW parameters) as accurate as its other MOA counterparts, while having way fewer iterations.

By the proposed algorithm, GSWs will become, not only customizable for any given dataset/application but also updated by every new record added into the recommender system. We approach justifying the effectiveness of the generalization idea as well as the optimization algorithm, by testing the proposed GSWs while utilizing 4 similarity measures, Pearson, Cosine, Moment Similarity of Random Variables (H.-J. Kwon & Hong, 2009), and a mapped version of Adjusted Euclidean Distance[17] (H. Sun, Peng, Chen, Liu, & Sun, 2011) [18]; testing them on different train / test proportions; comparing their error with the least errors, obtained by an exhaustive optimizer; and moreover, investigate the most-effective GSW for cold-start alleviation. Please, meanwhile, note that the proposed MOA, per se, is expected to be utilized as a fast and real-time-friendly optimization algorithm, along with its other rivals in MOAs state-of-the-art, when the addressed optimization problem has the three restrictions, enumerated above.

However, in the following, we will address the existing SWs in the state-of-the-art.

## *Existing SWs and their upgrade*

Here, we first present the existing SWs and then propose SWs which do not have the disadvantages of the existing ones.

Note that SWs evaluate the degree (i.e. weight) of the significance of the measured similarity between users $u_i$ and $u_j$. From a perspective, there are two main categories / clusters of SWs; univariate and multivariate: (1) SWs which are functions of only one variable ($|u_i \cap u_j|$) (almost half of SWs). SWs which are functions of two or more variables (e.g. $|u_i \cap u_j|$ and $|u_i \cap u_j|$).

In the following, we analyze these two categories, separately.

## *Univariate SWs*

The first category includes linear SW functions of $|u_i \cap u_j|$ which are ascending on a bounded interval $[0, C]$[19] or nonlinearly-ascending SW functions on the semi-infinite interval $[0, \infty)$. (Jl

[16] Some scholars –especially those who suggest the well-known taxonomies for optimizers- do not explicitly enumerate SMs under the umbrella term of population-based MOAs. However, because the first step of SM requires a group of cost-evaluations (regular or irregular meshes), which can be considered as a population of evaluations, based on which the SM is estimated, we consider them as population-based MOAs.

[17] It unifies all Euclidean distances between vectors in different dimensional vector spaces.

[18] To have a review on different proposed similarity measures, the interested reader is referred to (Bagchi, 2015)

[19] $C$ is a positive integer after which the function becomes equal to the constant 1 (maximum weight).

Herlocker & Konstan, 1999) present the first SW for tuning the significance of the similarity measures. If the number of two users' co-rated items are less than 50, the similarity value is multiplied by $\frac{|\text{Co-rated Set}|}{50}$. The authors named this fraction as "SW". In 2002, in the journal version of the same study (Jon Herlocker et al., 2002), the idea became generalized by substituting 50 by a 'significance threshold'. In 2004, in another paper of Herlocker (McLaughlin & Herlocker, 2004) [20], this SW is written in a more formulated style, but mistakenly the function was written as $\frac{\max(|I_u \cap I_v|, \gamma)}{\gamma}$ instead of $\frac{\min(|I_u \cap I_v|, \gamma)}{\gamma}$ (that is a typo). This is a typo as they write: "In order to achieve the best possible implementation, we have used the modification described in (Jon Herlocker et al., 2002), which weights similarities by the number of item ratings in common between u and v when <u>*less*</u> than some threshold parameter γ: $sim'(u, v) = \frac{\max(|I_u \cap I_v|, \gamma)}{\gamma}\ sim(u, v)$." Indeed in (McLaughlin & Herlocker, 2004), not only no innovation about SWs is provided, but also the paper has no focus on SWs and simply utilizes a previous work of Herlocker, as one of the main authors. However, in 2007, this mistake made the authors of (Ma, King, & Lyu, 2007) to a double mistake so that they, again, presented the $\frac{\min(|I_u \cap I_v|, \gamma)}{\gamma}$ SW in their study, supposed as a modification to Herlocker's SW.

Nevertheless, (Jamali & Ester, 2009) take advantage of sigmoid function as an SW that, unlike the Herlocker's SWs, has a y-intersect greater than 0 and instead of becoming 1 (after a specific $|u_i \cap u_j|$ value), it tends to 1 (when $|u_i \cap u_j|$ tends to +∞) as a horizontal asymptote. (Koren, 2010) proposes another SW, based on the simple technique of adding a constant number to the denominator ($\frac{|u_i \cap u_j|}{|u_i \cap u_j| + y}$) to keep the SW always less than 1 and therefore having the horizontal asymptote idea while keeping the y-intersect as 0 unlike the sigmoid SW (in which y-intersect is 0.5). (Ali Ghazanfar, Prugel-Bennett, Ghazanfar, & Prugel-Bennett, 2010) propose a SW that depends on the sign of the measured similarity which its significance is being weighted. If the measured similarity is positive, then the SW is $\frac{|u_i \cap u_j|}{y}$, which means scaling toward zero if $|u_i \cap u_j| < y$ and scaling toward infinity when $|u_i \cap u_j| > y$. But, if the measured similarity is negative, it is always scaled toward zero by $\frac{|u_i \cap u_j|}{y + \max(|u_i \cap u_j|, y)}$ using the horizontal asymptote idea. The considerable point of this SW is its dependence to similarity measure, as it has a different behavior in the positive domain than the negative domain of similarity measures. This logic is chosen for always paying less significance to negatively similar users (with opposite ratings) as represented in the Table 1.3.

**Table 1.3. Analyzing the "Ali" SW treatment. Comparing sim < 0 vs. sim ≥ 0.**

| SW treatment | $|\boldsymbol{u_i} \cap \boldsymbol{u_j}|$ from 0 to $\boldsymbol{y}$ | $|\boldsymbol{u_i} \cap \boldsymbol{u_j}|$ from y to +∞ |
|---|---|---|
| $\boldsymbol{sim < 0}$ | linearly increasing from 0 to "½" | concavely increasing tending to "1" |

[20] Co-authored with McLaughlin.

| $sim \geq 0$ | linearly increasing from 0 to "1" | linearly increasing tending to "+∞" |
|---|---|---|

This discriminating strategy is adopted to keep the significance of positive cases (the user pairs with compatible ratings), meaningfully, always more than negative cases (the user pairs with opposite ratings). Please note that this SW, despite its different domain ($[0, +\infty)$) from the domain of the well-known SWs in state-of-the-art ($[0,1]$), does not break the logic behind its "weighting nature" regarding the existence of the normalizing factor $k$ in Equation 2.1.

(M. Wang & Ma, 2016) propose an SW that is very similar to the one utilized by (Jl Herlocker & Konstan, 1999) with the difference that its domain begins from 2, and therefore it excludes the always-utilized point, 1, from the domain. However, considering that it is a modified version of Herlocker's SW, we do not consider it as a separated SW and deal with it as a special case of the Herlocker's GSW, which will be introduced later.

### *Multivariate SWs*

The second category of the proposed SWs in state-of-the-art are functions that depend on additional factors than the always-used variable $|u_i \cap u_j|$. In (Candillier, Meyer, & Fessant, 2008), without using the term 'significance', the Jaccard index ($\frac{|u_i \cap u_j|}{|u_i \cup u_j|}$) is utilized as a SW, which in addition to $|u_i \cap u_j|$, depends on the union of items which are rated by $u_i$ and $u_j$. (Zheng, Ma, R.Lyu, & King, 2009) propose an SW which depends on the number of items rated by each of $u_i$ and $u_j$, $\frac{2|u_i \cap u_j|}{|u_i|+|u_j|}$ which is also followed in his own studies later (Zibin Zheng, Hao Ma, Lyu, & King, 2011)(Xi Chen, Zheng, Yu, & Lyu, 2014). (Q. Sun, Wang, Wang, Ma, & Hsu, 2016) propose the $\frac{|u_i \cap u_j|}{\sqrt{|u_i| \times |u_j|}}$ SW that, similar to Zheng's SW, is a function of $|u_i|$ and $|u_j|$ in addition to $|u_i \cap u_j|$.

In section 3.6, we address the philosophy behind the SWs, based on which we propose generalized versions of them.

After finishing to address the literature review, and before illustrating the contribution of this dissertation (in chapter 2), it is necessary to explain an international project that is founded by the author of this dissertation and this dissertation provides the fundamental studies of that research project. Section 1.2 addresses this project.

## 1.2 pTER project

In this section, we firstly introduce an international research project group, called pTER, which is defined and started up by the author of this dissertation. The fundamental research part of pTER is conducted in this Ph.D. dissertation. However, because it is an international project composed of seven professors and seven researchers, it is necessary to firstly introduce and illustrate the contribution of pTER project itself before addressing the contribution of this dissertation (in chapter 2).

pTER project addresses a system that utilizes text documents of Social Networks authors to predict their sentiment/emotion about the items which they have not, even, written about, by means of Uncertainty-handling techniques. We call this system as possibilistic/probabilistic Text-based Emotion Rater (pTER).

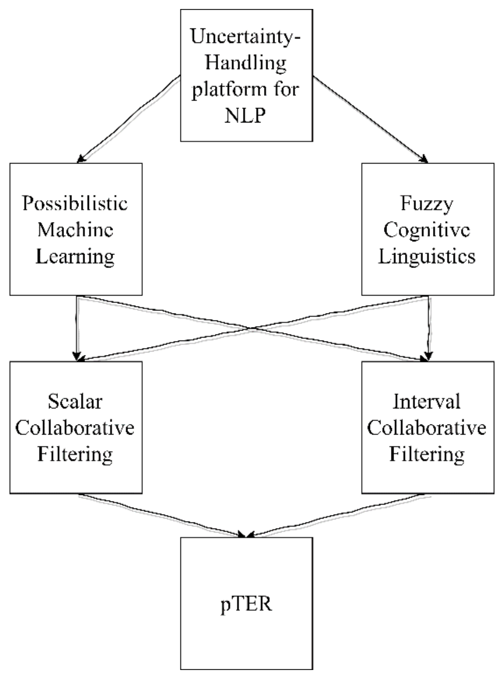



**Figure 1.2. Concept Map of the pTER project.**

As seen in the pTER concept map (Figure 1.2), from a perspective, all of the pTER sub-phases originate from a more informative generation of Text Mining resources; an Uncertainty-handling platform for Text Mining, in which the existing potential uncertainties in lexical databases are addressed. It can be fuzzification of the mentioned databases or their conceptualization, providing hierarchical relations between them to be utilized, while handling uncertainties.

Then, two different Text Mining approaches to Sentiment Analysis are adopted for Sentiment Analysis: (1) Possibilistic Machine Learning (PosML), and (2) Fuzzy Cognitive Linguistics (Fuzz-Cog-Lin), each of which extracting sentiments of users on different items/aspects, from their text documents. Thus, each of the two sentiment extractors produces a User/Item opinion matrix in which each cell represents the sentiment of the corresponding user about the corresponding item.

After constructing such User/Item opinion matrix, we have to handle a problem that we call "cold-co-start problem." This problem addresses the situation in which users have few commonly commented items. In "Scalar Collaborative Filtering" sub-phase of pTER (the bottom left node in the concept map), we provide a generalization to a series of Uncertainty-handling techniques, which mainly affect this vulnerability. Having a CF system, which is able to deal with cold-co-start problem, enables us to correctly fill in the empty cells of the inputted User/Item opinion matrix. Filling in the empty cells of the mentioned matrix is exactly the Sentiment Prediction of users for the items about which they have not written.

What described so far, explained the logic of Scalar pTER that is the very bottom-left node in the above concept map. However, pTER has also an interval version. It is because both of the (1) Uncertainty-sustainable Text Mining platform and (2) Uncertainty-handling analysis of the expressed sentiment of users from their written documents have a measurement error in their output. If we do not take this error into account, the errors will permeate to the next step that is Sentiment Prediction, and this permeation causes in prediction precision to be decreased.

Thus, by taking advantage of the interval-supporting versions of the Uncertainty-sustainable Text Mining platforms, each of the two Sentiment Analysis approaches provide an uncertain version as well, in which, the User/Item opinion matrix is a matrix of interval-valued sentiments, or else, at least, it is created by means of a new generation of Uncertainty-sustainable Text Mining platforms which is expected to outperforms the other methods, in the presence of uncertainty. For example, the sentiment of user 'i' about the item 'j' in such a User/Item opinion matrix can be represented as $S_{i,j} = [-0.05, +0.15]$. However, after the creation of such opinion matrices, the standard CF systems in which the input is a scalar-valued User/Item matrix, cannot predict/fill-in the empty cells. That is the reason considering which, we have also approached a new generation of CF techniques in which the inputted User/Item opinion matrix is Interval-valued.

By having this new generation of CF systems, we can again perform the Emotion Rating/prediction process, but this time with better accuracy, thanks to taking the measurement errors into account. The uncertain CF, as well as the interval-output version of pTER, have been represented in the bottom-right two nodes of the concept map.

In brief, pTER project provides two versions of Sentiment Prediction systems, both of which (as expected) converting text-documents of people to their unexpressed sentiment/emotion, having similar treatment but different potency, resulting in different accuracies where the interval version (the one with the better accuracy) is founded on the standard version.

# 2 Objectives

This dissertation provides the required foundations for the pTER project. In other words, by the six defined research studies in this dissertation an Uncertainty-handling Text-based Emotion Rating system can be established.

However, as mentioned, the main fruit of the pTER project, is decided to be disseminated only after all of the other research projects are finished so that, the entire defined Uncertainty-handling potency of that research study is freed.

The scope of a dissertation can be elaborately specified by specifying its position in Problem Space and Approach Space. In this chapter, we would bring a brief discussion about the position of this dissertation (Scalar pTER) in these two spaces.

## 2.1 Problem Space

In this section, we discuss the dissertation position in the Problem Space. In other words, we specify which problem is going to be solved among all of the attracting problems for the research communities. As mentioned at the beginning, this dissertation (Scalar-pTER) and in general pTER addresses an almost new problem whose lifetime is almost one decade:

**Input**: The written documents/notes of people (e.g. their posts in Social Networks)

**Output**: The predicted sentiment of each subject about what he has not written about.

As an illustrating example, suppose that the Budgetplaces© company is going to estimate / predict the sentiment / opinion of its customers about different products, by means of the information of the customers in the Twitter online social network, while not all of its customers have written Tweets about all of the products. Then, Budgetplaces© would require an effective "Sentiment Prediction" algorithm to solve this problem.

However, the novelty of this dissertation is on its new approach to this (existing) problem. The proposed new approach is going to resolve some of the drawbacks of the existing Sentiment Prediction algorithms. In the following, we discuss the position of this dissertation in the Approach Space.

## 2.2 Approach Space

In this section, we discuss the dissertation position in the Approach Space. In other words, we specify our philosophically-novel approach/viewpoint (the proposed doctrine) by which the problem is better-solved, in comparison with the existing approaches.

The main claim of this dissertation is the following philosophic doctrine:

**"Uncertainty-handling makes text-based sentiment prediction more certain"**

It is represented by the following sub-doctrines:

1- "Uncertainty-handling Text Mining makes Sentiment Analysis more certain"
2- "Uncertainty-handling Similarity Measures makes Sentiment Prediction more certain"

In the following, we provide an abstract of what the above sub-doctrines are, and how we have approached their proof.

### 2.2.1 Sub-Doctrine 1

This subsection is about this doctrine:

"Uncertainty-handling Text Mining makes Sentiment Analysis more certain"

In the classical view to Text Mining, the very small brick of Text Mining (i.e. wordsense) is potentially considered equivalent to some other wordsenses all of which equally addressing the same meaning. In a more modern already existing uncertainty-handling view, wordsenses are considered to address meaning by a compatibility degree in [0,1]. However, this modern and more informative look suffers lack of the efficiency proof as well as lack of a fuzzification algorithm, two reasons for which this novel approach has been kept far from Text Mining community trust, and therefore far from the application. In the following, we describe these two shortcomings.

(1) As mentioned in section 1.1, there is no algorithm to fuzzify the already under-usage WordNets. This is despite the fact that the already existing WordNets are, in practice, the platform of thousands of Text Mining applications. The reluctance of the Text Mining community to the existing fuzzy WLDs is because of two reasons. First, the synsets of the existing fuzzy WLDs are different from the already existing standard (non-fuzzy) WLDs. Second, the community is reluctant to abandon the well-known WLDs on which several research works have been conducted. In this dissertation, we provide an algorithm for including uncertainty in a standard Text Mining platform (WordNet) of any natural language. As a hypothetical illustrating example of one of the outputted fuzzy synsets of the proposed algorithm in this dissertation (for the standard WordNet), please consider the following synset

*{(WS(flower.n.02.flower), 1), (WS(flower.n.02.bloom), 0.7), (WS(flower.n.02.blossom), 0.6)}*

The above hypothetical fuzzy synset represents the fuzzified version of an already existing synset in the standard WordNet. However, it also specifies the membership degree of each wordsense (represented by 'WS') in this synset, which is the degree of compatibility of each wordsense with the mentioned synset. As it can be seen, the difference of the abovementioned fuzzy synset, with the already existing fuzzy synsets (proposed by the state-of-the-art fuzzy WordNet producing algorithms) is preserving the structure of the inputted / existing WordNets. It is because the synsets and the wordsenses, resulted by the mentioned algorithm, are not modified while fuzzification and they are simply annotated by the membership degrees.

(2) The second shortcoming that is the main subject of this section is: There is no proof to show that fuzzification of the existing lexical databases practically improves Text Mining applications. Thus, this dissertation provides a proof for the sub-doctrine: "Uncertainty-handling Text Mining makes Sentiment Analysis more certain." In other words, we prove that this more informative look to Text Mining is practically improving at least one of its applications that is Sentiment Analysis. In particular, we provide bridges from Uncertainty-handling WordNets to Sentiment Analysis in Possibilistic Machine-Learning-based Sentiment Analysis (Uncertainty-handling Cognitive-Linguistics-based Sentiment Analysis is defined and left for the in-progress research of the mentioned students) and demonstrate improvement of Sentiment Analysis in each of them.

(3) After addressing these two shortcomings, it is necessary, to address the relation of fuzzification and "Text Mining" (vs. Natural Language Processing) concept. Text Mining usually addresses a Data Mining look to text and natural language. The fuzzified version of the WLDs can both be utilized in pure linguistic models which may model a piece of text by a parse tree or even a complex dependency graph or in Data Mining applications by means of modelling text the standard mathematical data structures such as vectors or sets or … The same discussion can be presented for Sentiment Analysis. If the modelled text, by fuzzified WLDS, is a linguistic model (e.g. fuzzified version of parse trees or dependency graphs), it can be utilized in linguistic-approached sentiment analysis systems such as (Reforgiato Recupero et al., 2015). However, if it is a Text Mining model, it still can be utilized by Machine Learning based Sentiment Analysis. In this dissertation, we focus on the Text Mining approach by proposing auxiliary fuzzified Text Mining models to illustrate the efficiency of fuzzified WLDs over the standard versions, and therefore, we choose the Text Mining term for this doctrine. However, it is necessary to note that the next steps of this research (which have already been planned in the pTER project) focus on the Linguistics approaches, as the main goal.

This proof, on the one hand, provides the Text Mining community with the first proof/motivation for application of this modern look to Text Mining, and on the other hand, considering that Sentiment Analysis is half of the Text-based Sentiment Prediction process, this dissertation has so far proven half of the doctrine: "Uncertainty-handling makes Text-based Sentiment Prediction more certain."

The following subsection provides the second part of the mentioned doctrine.

### 2.2.2 Sub-Doctrine 2

This subsection is about the following sub-doctrine:

> "Uncertainty-handling Similarity Measures makes Sentiment Prediction more certain"

Significance Weights (SW) (already existing in state-of-the-art) are mathematical tools for providing a degree of certainty/uncertainty for Similarity Measures, based on the number of common items in two under-comparison vector-based profiles.

CF, as one of the widely used Sentiment Prediction tools by Recommender Systems, is one of the main applicants of SWs. Considering that, on the one hand, Similarity Measurement is a key step in memory-based CF, and on the other hand, in the mentioned Sentiment Prediction algorithms profiles of users/items are recorded/represented by vectors, SWs are quite effective tools for handling uncertainty in Similarity Measurement phase of such prediction algorithms to specify how-certainly a standard Similarity Measure measures the similarity of two users/items, based on the number of common items of the profiles.

Although SWs are already being used in state-of-the-art of CF algorithms, they still cannot provide Similarity Measures with uncertainty-handling in some other dimensions. Many of the already existing SWs suffer from the drawback of rigidness. Representing a vector-based profile by $v$, all of the already existing SWs are specified functions of $|v_i \cap v_j|$, $|v_i \cup v_j|$, $|v_i|$, $|v_j|$, or a combination of them. For example, the following plot represents the behavior of Sigmoid SW proposed in 2009 (as a function of $|u_i \cap u_j|$) for being used in the mentioned Sentiment Prediction systems.

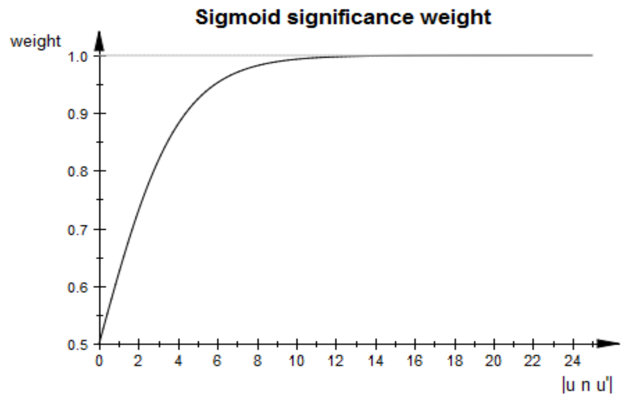



**Figure 2.1. Sigmoid SW, used for including uncertainty in Similarity Measurement of two vector-based profiles.**

As depicted in Figure 2.1, the above SW assigns 50% of certainty to a standard Similarity Measure while comparison of two vector-based profiles when the number of common items is 0; assigns 62% for 1 common neighbors, 73% for 2, 82% for 3, and... As it can be seen, although this SW is utilized for handling uncertainty in Similarity Measurement, yet it, itself, has a rigid and inflexible behavior in assigning the measurement significance/certainty.

Although the number of items / "common items" / "total items" of vector-based profiles is *de facto* the only factor for the uncertainty which SWs handle, a fix number of such items may provide more certainty in one context, rather than other ones. This context-based certainty difference can be handled by means of a flexible SW; flexibility of stretchiness in x-axis direction as well as flexibility in y-intercept (e.g. Figure 2.2).

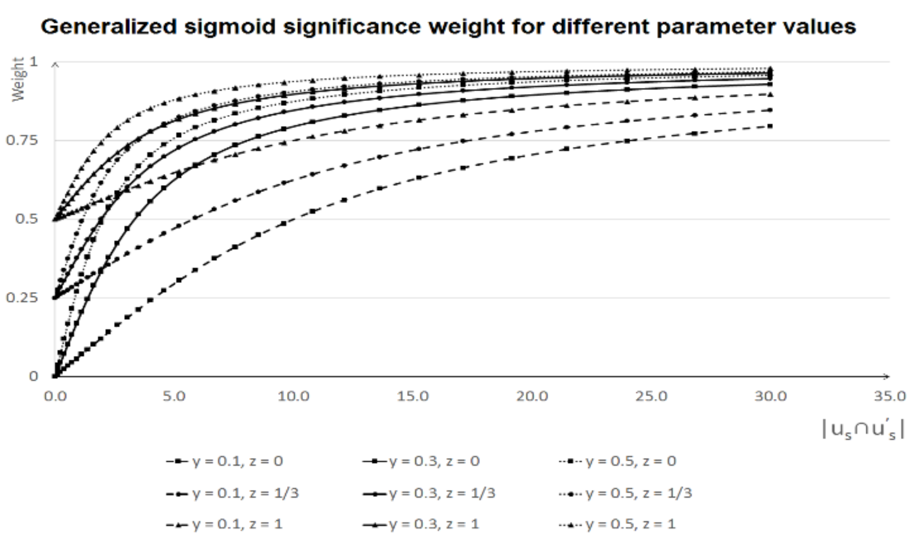



**Figure 2.2. More Uncertainty-handling Sigmoid SW.**

As an example for expressing the different uncertainty in different rating contexts, assuming that the vector-based profiles are representing users in a community, if the users have enough information for being analyzed and can be clustered in $R^n$ into two very far clusters, then even with very few number of similarly rated/commented common items we can reach a high value of certainty about similarity and the SW plots have to have high initial acceleration; on the contrary, if the data volume is not enough in proportion to data diversity, we would not know how correlated the users are and how their behavior is[21]. This frequently happens in the cold-co-start situation that is the common problem in Text-based Sentiment Prediction systems, as we have defined in the Scalar CF sub-group.

In other words, more flexible SWs, and therefore Uncertainty-handling Similarity Measures, provide a better (more certain) prediction process, that is the second sub-doctrine, in the proposed doctrine.

In the following chapters of this dissertation, we address the concepts, algorithms, and proof, promised while explanation of this doctrine.

[21] Generally, Uncertainty-handling is useful mostly in situations in which the data is relatively insufficient.

# 3 Methods

In this chapter, we address the theoretical novelties of this dissertation. We would like to remind the series of the related scientific scopes/fields which are required for Uncertainty-handling Sentiment Prediction systems: Text Mining, Machine Learning, and Collaborative Filtering. These three can be considered as rings of a united chain. While reviewing the related literature (section 1.1), we mentioned shortcomings in each of these three rings for achieving Uncertainty-handling in Sentiment Prediction. In this section, we propose methods/algorithms for resolving those shortcomings.

As the first ring of the chain, we propose two novel algorithms on fuzzy WordNet-like Lexica Databases, which are presented in sections 3.1 and 3.2, as well as, one auxiliary Text Mining model which is proposed for the evaluation purposes that is proposed in section 3.3. Then, as the second ring of the chain, in section 3.4, we propose two novel text classifier algorithms for handling the uncertainty that normally occurs while dealing with the Social Networks data/information. For a pre-evaluation of the proposed classifiers in section 3.4, section 3.5 is devoted to providing a complexity analysis on the mentioned classifiers, along with a comparison with the state-of-the-art counterparts. In the end and as the third ring of the chain, in section 3.6 we provide our novel idea for more alleviation of the cold-co-start problem. Then, regarding the requirement that we encounter while dealing with the real-time applications of GSWs, due to the lack of theory for a linear-complexity Meta-heuristic optimizer, in section 3.7, we would provide a fuzzy optimizer with linear updating-complexity.

## 3.1 An algorithm for fuzzification of WordNet-like lexical databases

In this section, we propose an algorithm for constructing fuzzy synsets in any language. As its input, the algorithm requires: (1) A large corpus ($C$) of documents of that language and (2) a WSD algorithm $W$ (each WSD algorithm is paired with a WLD and each WLD contains a set $S^{(W)}$ of synsets of that language; $u_{i,k}^{(W)}$ stands for the wordsense $k$ from the synset $i$ of the WLD engaged with $W$, and $S_k^{(W)}$ stands for the $k^{th}$ synset of $W$.).

This algorithm is comprised of the following 4 steps:

**Step 1) Frequency:** For each wordsense $u_{i,k}^{(W)}$ of each synset $S_k^{(W)}$ calculate $f^{(C,W)}\left(u_{i,k}^{(W)}\right)$, that is the frequency of $u_{i,k}^{(W)}$ in $C$.

**Step 2) Probability:** For each wordsense $u_{i,k}^{(W)}$ of each synset $S_k^{(W)}$ calculate

$pr^{(C,W)}(u_{i,k}^{(W)}) = f^{(C,W)}(u_{i,k}^{(W)}) / \sum_{u_{m,k}\in S_k} f^{(C,W)}(u_{m,k}^{(W)})$.

**Step 3) Possibility:** For each wordsense $u_{i,k}^{(W)}$ of each synset $S_k^{(W)}$ calculate

$$\pi_{1983}^{(C,W)}(u_{i,k}^{(W)}) = \Sigma_{u_{m,k}^{(W)}\in S_k^{(W)}} \min\left(pr^{(C,W)}(u_{i,k}^{(W)}), pr^{(C,W)}(u_{m,k}^{(W)})\right)$$

$$\pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right) = \Sigma_{u_{m,k}^{(W)}|pr^{(C,W)}\left(u_{m,k}^{(W)}\right)\le pr^{(C,W)}\left(u_{i,k}^{(W)}\right)} pr^{(C,W)}\left(u_{m,k}^{(W)}\right)$$

**Step 4) Membership:** For each wordsense $u_{i,k}^{(W)}$ of each synset $S_k^{(W)}$ calculate the membership degree of $u_{i,k}^{(W)}$ in the fuzzy set $S_k^{(W)}$

$$\mu_{S_k,1983}^{(C,W)}\left(u_{i,k}^{(W)}\right) = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$$
$$\mu_{S_k,1993}^{(C,W)}\left(u_{i,k}^{(W)}\right) = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right).$$

### 3.1.1 Proof of the algorithm

Here, we propose a theoretical proof for the algorithm validity. The proposed proof has three parts: First, we prove that the formula related to the step 2 of the algorithm is the same as the desired probability; the second part proves that the formula related to the step 3 of the algorithm is the same as the desired possibility; and at the end, the third part proves that the formula related to the step 4 of the algorithm is the same as the desired membership.

**Part 1. $\boldsymbol{pr^{(C,W)}}$() = probability**

**Definition 1.** Given a WSD algorithm W and a corpus of ordered documents $C$, the sequence $L_{k,C,W} = \left(l_{k,C,W}^{(a)}\right)_{a=1}^{n}$, is defined so that $l_{k,C,W}^{(a)}$ represents the $a^{th}$ occurrence of any of the wordsenses (recognized by $W$) of the synset $S_k$ in $C$.

**Definition 2.** For a WSD $W$, $U_{i,k,W}: S_k^{(W)} \to \{0,1\}$ is defined as a Bernoulli random variable that for a given $u \in S_k^{(W)}$, it outputs 1 if $u = u_{i,k}^{(W)}$ and outputs 0, otherwise.

**Definition 3.** The Bernoulli process $C_{i,k,W}$ is defined as the sequence of random variables $\left\{U_{i,k,C,W}^{(a)}\right\}_{a=1}^{|L_{k,C,W}|}$, which its $a^{th}$ element represents $U_{i,k,W}\left(l_{k,C,W}^{(a)}\right)$, for an arbitrary corpus $C$ and WSD $W$.

**Lemma 1.** Consider an arbitrary Bernoulli process $C_{i,k,W}$, assuming that $\left\{U_{i,k,C,W}^{(a)}\right\}_{a=1}^{|L_{k,C,W}|}$ are independent and identically distributed (i.i.d) Bernoulli random variables with success probability of $pr_{i,k}$. Then, for the random variable $\overline{U_{\iota,k,C,W}} = \frac{1}{|L_{k,C,W}|}\sum_{i=1}^{|L_{k,C,W}|} U_{i,k,C,W}^{(a)}$, we have $Pr\left(\lim_{|L_{k,C,W}|\to\infty} \overline{U_{\iota,k,C,W}} = P\left(U_{i,k,W} = u_{i,k}\middle|S_k\right)\right) = 1.$

**Proof.** A direct result of the Khintchine's Strong Law of Large Numbers (Sen & Singer, 1993) results in $Pr\left(\lim_{|L_{k,C,W}|\to\infty} \overline{U_{\iota,k,C,W}} = pr_{i,k}\right)$. Moreover, we know that $\forall a \in \{1,2,\dots,|L_{k,C,W}|\}: pr_{i,k} = pr_{i,k}^{(a)} = P\left(U_{i,k,C,W}^{(a)} = u_{i,k}\middle|S_k\right)$. However, we know that the i.i.d. $U_{i,k,C,W}^{(a)}$ Bernoulli random variables are the i.i.d elements of the Bernoulli process $C_{i,k,W}$. This implies that $\forall a \in \{1,2,\dots,|L_{k,C,W}|\}: U_{i,k,C,W}^{(a)} = U_{i,k,W}\left(l_{k,C,W}^{(a)}\right)$. In other words, $U_{i,k,C,W}^{(a)}$ are tantamount to i.i.d trials of the random variable $U_{i,k,W}$, all of which having the distribution $U_{i,k,W}$. Thus, we can write $pr_{i,k} = pr_{i,k}^{(a)} = P\left(U_{i,k,C,W}^{(a)} = u_{i,k} \middle| S_k\right) = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)$. ■

**Definition 4.** Given a WSD algorithm W and a corpus of ordered documents $C$, the sequence $L_{C,W} = \left(l_{C,W}^{(a)}\right)_{a=1}^{n}$, is defined so that $l_{C,W}^{(a)}$ represents the $a^{th}$ occurrence of any of the wordsenses (recognized by $W$) in $C$.

**Definition 5.** For a WSD $W$, $U_{k,W}: WLD(W) \to \{0,1\}$ is defined as a Bernoulli random variable that for a given $u \in WLD(W)$, it outputs 1 if $u \in S_k$ and outputs 0, otherwise, where $WLD(W)$ stands for the WLD, engaged with the WSD $W$.

**Definition 6.** The Bernoulli process $C_{k,W}$ is defined as the sequence of random variables $\left\{U_{k,C,W}^{(a)}\right\}_{a=1}^{|L_{C,W}|}$, which its $a^{th}$ element represents $U_{k,W}\left(l_{C,W}^{(a)}\right)$, for an arbitrary corpus $C$ and WSD $W$.

**Lemma 2.** In an arbitrary Bernoulli process $C_{k,W}$, assuming that $\left\{U_{k,C,W}^{(a)}\right\}_{a=1}^{|L_{C,W}|}$ are i.i.d Bernoulli random variables with success probability of $pr_k = P\left(U_{k,W} \in S_k\right)$, then, for the random variable $\overline{U_{k,C,W}} = \frac{1}{|L_{C,W}|}\sum_{i=1}^{|L_{C,W}|} U_{k,C,W}^{(a)}$, we have $Pr\left(\lim_{|L_{C,W}| \to \infty} \overline{U_{k,C,W}} = pr_k\right) = 1$.

**Proof.** The same as the proof of Lemma 1. ■

**Lemma 3.** Consider an arbitrary infinitely-large corpus $C$, a precise WSD $W$, and a probable $S_k$. If the usage of each wordsense / synset, in $C$, is independent of the usage of other wordsenses / synsets, we almost surely, have $\left|L_{k,C,W}\right| \to +\infty$.

**Proof.** Because $C$ is infinitely large ($\left|L_{C,W}\right| \to +\infty$), Lemma 2 implies that $Pr\left(\sum_{i=1}^{|L_{C,W}|} U_{k,C,W}^{(a)} = pr_k \cdot \left|L_{C,W}\right|\right) = 1$. But, we know that $S_k$ is probable (i.e. $pr_k > 0$), and therefore, $\sigma_k = pr_k \cdot \left|L_{C,W}\right| \to +\infty$. Moreover, we know that $\left|L_{k,C,W}\right| = \sum_{i=1}^{|L_{C,W}|} U_{k,C,W}^{(a)}$. Thus, we have $Pr\left(\left|L_{k,C,W}\right| = \lim_{\sigma_k \to +\infty} \sigma_k\right) = 1$. Thus, almost surely, $\left|L_{k,C,W}\right| \to +\infty$. ■

**Theorem 1.** Consider an arbitrary infinitely-large corpus $C$, a precise WSD $W$, and a probable $S_k$. If the usage of each wordsense / synset, in $C$, is independent of the usage of other wordsenses / synsets, we almost surely, have $pr_{i,k} = pr^{(C,W)}\left(u_{i,k}^{(W)} | S_k\right) = f^{(C,W)}(u_{i,k}^{(W)}) / \sum_{u_{m,k} \in S_k} f^{(C,W)}(u_{m,k}^{(W)})$.

**Proof.** Lemma 3 implies that $\left|L_{k,C,W}\right| \to +\infty$. Now, Lemma 1 implies that for any $u_{i,k} \in S_k$, we have $Pr\left(\frac{1}{|L_{k,C,W}|}\sum_{i=1}^{|L_{k,C,W}|} U_{i,k,C,W}^{(a)} = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)\right) = 1$. However, we know that

$\sum_{i=1}^{|L_{k,C,W}|} U_{i,k,C,W}^{(a)} = f^{(C,W)}\left(u_{i,k}^{(W)}\right)$ and also know that $|L_{k,C,W}| = \sum_{u_{m,k}\in S_k} f^{(C,W)}(u_{m,k}^{(W)})$. Therefore, we have $Pr\left(\frac{1}{\sum_{u_{m,k}\in S_k} f^{(C,W)}(u_{m,k}^{(W)})}\sum_{i=1}^{|L_{k,C,W}|} f^{(C,W)}(u_{i,k}^{(W)}) = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)\right) = 1$, and equally, $\Pr\left(pr^{(C,W)}\left(u_{i,k}^{(W)}\right) = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)\right) = 1$. ■

## Part 2. $\boldsymbol{\pi\left(u_{i,k}\right)}$ = possibility

**Definition 7** (D. Dubois & Prade, 1983)**.** The degree of necessity of event A ⊆ X is the extra amount of probability of elementary events in $A$ over the amount of probability assigned to the most frequent elementary event outside $A$. In other words, $N(A)$ is defined as the necessity measure of $A$, so that, $N(A) = \sum_{x_i\in A} \max\left(pr_i - \max_{x_k\notin A} pr_k\right)$. It is also called the Shafer's consonant belief function (Shafer, 1976).

**Proposition 1.** $N(A)$ satisfies the following 3 axioms of necessity function: $N(\emptyset) = 0$, $N(X) = 1$, and $\forall A, B \subseteq X, N(A\cap B) = \min\left(N(A), N(B)\right)$.

**Proof.** proven in (D. Dubois & Prade, 1983). ■

**Definition 8** (D. Dubois & Prade, 1983)**. "**Viewing $N(A)$ as the grade of the impossibility of the opposite event $\bar{A}$ we can define the grade of the possibility of $A$ by $\forall A \subseteq X, \Pi(A) = 1 - N(\bar{A})$."

**Proposition 2.** The set function Π is a possibility measure in the sense of Zadeh (Zadeh, 1978).

**Proof.** proven in (D. Dubois & Prade, 1983). ■

**Lemma 4.** Consider $\pi(x), pr(x)$ as possibility and probability mass functions, engaged with the Possibility and Probability distributions Π and $P$. Adopting the Shafer's consonant belief function as the necessity measure, we will have $\pi(x_i) = \sum_{j=1}^{n} \min\left(pr(x_i), pr\left(x_j\right)\right), \forall x_i \in X$.

**Proof.** proven in (D. Dubois & Prade, 1983). ■

**Theorem 2.** Consider an arbitrary infinitely-large corpus $C$, a precise WSD $W$, and a probable $S_k$. If the usage of each wordsense / synset, in $C$, is independent of the usage of other wordsenses / synsets, and if the Shafer's consonant belief function is adopted as the necessity measure, then, for any $u_{i,k} \in S_k$, we almost surely, will have

$$\pi_{i,k} = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right) = \sum_{u_{m,k}^{(W)}\in S_k^{(W)}} \min\left(pr^{(C,W)}\left(u_{i,k}^{(W)}\right), pr^{(C,W)}\left(u_{m,k}^{(W)}\right)\right).$$

**Proof.** Theorem 1 implies that, almost surely, $pr^{(C,W)}\left(u_{i,k}^{(W)}\right) = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)$. Using this fact, besides Lemma 4, we almost surely will have

$\pi\left(u_{i,k}|S_k\right) = \sum_{j=1}^{n} \min\left(pr^{(C,W)}\left(u_{i,k}^{(W)}\right), pr^{(C,W)}\left(u_{j,k}^{(W)}\right)\right) =$
$\sum_{u_{m,k}^{(W)} \in S_k^{(W)}} \min\left(pr^{(C,W)}\left(u_{i,k}^{(W)}\right), pr^{(C,W)}\left(u_{m,k}^{(W)}\right)\right)$, or equally, $\pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. Therefore, we almost surely have $\pi\left(u_{i,k}|S_k\right) = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. ■

**Definition 9** (D. J. Dubois & Prade, 1980)**.** Consider the probability distribution $P$ and possibility distribution $\Pi$ defined on $X$. Then, $P$ and $\Pi$ have DP-consistency[22] if $\forall A \subseteq X, P(A) \leq \Pi(A)$.

**Proposition 3.** DP-consistency is a standard consistency measure in the sense of Delgado-Moral.

**Proof.** Proven in (Delgado & Moral, 1987).

**Definition 10.** Consider the probability distribution $P$ and possibility distribution $\Pi$, defined on $X$. Then, $P$ and $\Pi$ have the preference-preservation relation if $\forall x, x' \in X: \pi(x) > \pi(x') \Leftrightarrow pr(x) > pr(x')$, where $\pi(x)$ and $pr(x)$ are the possibility and probability mass functions, engaged with $\Pi$ and $P$, both defined on $X \rightarrow [0,1]$.

**Proposition 4.** The condition $\forall x, x' \in X: \pi(x) > \pi(x') \Leftrightarrow pr(x) > pr(x')$ is equal with $\pi(x) < \pi(x') \Leftrightarrow pr(x) < pr(x')$ or $\pi(x) \leq \pi(x') \Leftrightarrow pr(x) \leq pr(x')$ or $\pi(x) \geq \pi(x') \Leftrightarrow pr(x) \geq pr(x')$.

**Proof.** Considering that $x$ and $x'$ do not have any discriminative specificity, the condition can be read as $\pi(x') < \pi(x) \Leftrightarrow pr(x') < pr(x)$. Moreover, contraposition of the mentioned equal conditions, yields in conditions with "≤" and "≥."

**Definition 11** (D. Dubois, Prade, & Sandri, 1993)**.** Given $X$ as a finite set of elements and $P$,$\Pi$ as probability and possibility distributions on $X$, and $p, \pi$ the corresponding mass functions, the transformed possibility $\pi$ is maximally specific when $\sum_{x \in X} \pi(x)$ has the minimum value, respecting preference-preservation and DP-consistency of $P$,$\Pi$.

**Lemma 5.** Given a probability distribution $P$ and probability mass function $pr(x)$ in the finite Universe of discourse $X$, the possibility distribution $\Pi$, in the same time, satisfies the 3 restrictions: DP-consistency, preference preservation, and maximally specificity, if and only if $\forall x \in X, \pi(x) = \sum_{\{x' : pr(x') \leq pr(x)\}} pr(x')$.

**Proof.** Without losing the generality, suppose that $X = \{x_1, x_2, \dots, x_n\}$ while (upon Preposition 4) we have $pr(x_1) \leq pr(x_2) \leq \cdots \leq pr(x_n)$. Utilizing Preposition 4, preference preservation implies that $\pi(x_1) \leq \pi(x_2) \leq \cdots \leq \pi(x_n)$. Cosnider $A_i = \{x_1, x_2, \dots x_i\}$. DP-consistency implies that $\forall A_i, \Pi(A_i) \geq P(A_i)$. Thus, $\forall A_i, \max\{\pi(x_1), \pi(x_2), \dots \pi(x_i)\} \geq \sum_{k=1}^{i} pr(x_k)$. Therefore, we have $\forall A_i, \pi(x_i) \geq \sum_{k=1}^{i} pr(x_k)$. Now, because $\pi(x_i) = \sum_{k=1}^{i} pr(x_k)$, from the one hand satisfies the preference preservation and DP-consistency restrictions, and from the other hand, includes the minimum allowed values of the $\pi(x_i) \geq \sum_{k=1}^{i} pr(x_k)$ constraint, $\pi(x_i) = \sum_{k=1}^{i} pr(x_k)$ would be

[22] DP stands for Dubois-Prade. There are two other consistency measures, proposed by Zadeh (Zadeh, 1978) and Sugeno (Sugeno, 1972). The interested reader is referred to Delgado and Moral (Delgado & Moral, 1987) which analyzes these three, in detail.

the unique minimal case satisfying the 3 mentioned constraints. Please note that the expressions $\pi(x_i) = \sum_{k=1}^{i} pr(x_k)$ and $pr(x_1) \leq pr(x_2) \leq \cdots \leq pr(x_n)$ equals with $\pi(x_i) = \sum_{\{x_k: pr(x_k) \leq pr(x_i)\}} pr(x_k)$. ■

Please note that the formula $\pi(x_i) = \sum_{\{x_k: pr(x_k) \leq pr(x_i)\}} pr(x_k)$ although introduced in 1982 (D. Dubois & Prade, 1982), it is usually known and referenced by (D. Dubois et al., 1993), a better known research work from 1993 where the same authors propose both its discrete and continuous versions.

**Theorem 3.** Consider an arbitrary infinitely-large corpus $C$, a precise WSD $W$, and a probable $S_k$. If the usage of each wordsense / synset, in $C$, is independent of the usage of other wordsenses / synsets, and if the 3 constraints of DP-consistency, preference-preservation, and maximally specificity have to be satisfied, then, for any $u_{i,k} \in S_k$, we almost surely, will have $\pi_{i,k} = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right) = \sum_{u_{m,k}^{(W)} | pr^{(C,W)}\left(u_{m,k}^{(W)}\right) \leq pr^{(C,W)}\left(u_{i,k}^{(W)}\right)} pr^{(C,W)}\left(u_{m,k}^{(W)}\right)$.

**Proof.** Theorem 1 implies that, almost surely, $pr^{(C,W)}\left(u_{i,k}^{(W)}\right) = P\left(U_{i,k,W} = u_{i,k} \middle| S_k\right)$. Using this fact, besides the Lemma 5, we almost surely will have $\pi\left(u_{i,k}|S_k\right) = \sum_{u_{m,k}^{(W)} | pr^{(C,W)}\left(u_{m,k}^{(W)}\right) \leq pr^{(C,W)}\left(u_{i,k}^{(W)}\right)} pr^{(C,W)}\left(u_{m,k}^{(W)}\right)$, or equally, $\pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. Therefore, we almost surely have $\pi\left(u_{i,k}|S_k\right) = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. ■

Please note that Dubois and Prade, in 1993 (D. Dubois et al., 1993), illustrate that the possibility mass function v.93 provides a maximally informative transformation from probability to possibility distribution. Both transformations have advantages and drawbacks; the possibility mass function v.83 produces more homogeneous values, denser around 1 and always greater than or equal to v.93 values[23]. However, being the v.93 the maximally informative transformation, we expect the Fuzzified WLDs v.93 to be more efficient (than v.83) in Text Mining applications.

## Part 3. $\boldsymbol{\mu_{S_k}\left(u_{i,k}\right) = \pi\left(u_{i,k}\right)}$

**Definition 12** (Zadeh, 1978)**.** Let $F$ be a fuzzy subset of a universe of discourse $U$, which is characterized by its membership function $\mu_F$, with the grade of membership, $\mu_F(u)$, interpreted as the compatibility of $u$ with the concept labeled $F$. Also, Let $X$ be a variable taking values in $U$. Then, $F$ is postulated to act as a fuzzy restriction, $R(X)$, associated with $X$ and the proposition "$X$ is $F$," translates into $\mathrm{R(X)} = \mathrm{F}$.

**Definition 13** (Zadeh, 1978)**.** An arbitrary fuzzy restriction $R(X)$ associates a possibility distribution, $\Pi_X$, with $X$ which is postulated to be equal to $R(X)$ (i.e., $\Pi_X = R(X)$).

**Definition 14** (Zadeh, 1978)**.** Consider a fuzzy set $F$, a variable $X$ taking values in the universe of discourse $U$ and the $R(X)$ associated with $F$ and $X$. The possibility distribution function associated

[23] The least informative version is a version that assigns 1 to possibility of the entire classes.

with $X$ is denoted by $\pi_X$ and is defined to be numerically equal to the membership function of F (i.e. $\pi_X \triangleq \mu_F$).

**Lemma 6.** Consider a fuzzy set $F$, a variable $X$ taking values in the universe of discourse $U$, and the $\Pi(X)$, associated with $F$ and $X$. Then, $\pi_X(u)$ the possibility that $X = u$, given that "X is F," is postulated to be equal to $\mu_F(u)$.

**Proof.** Upon Definition 12, we know that "$X$ is $F$," translates into $R(X) = F$ and upon Definition 13, we know that $\Pi_X = R(X)$. Thus, "$X$ is $F$," is an intrinsic assumption in $\Pi_X$. Moreover, upon the Definition 14, it is postulated that $\pi_X \triangleq \mu_F$. Thus, $\pi_X(u) = \mu_F(u)$, given that "X is F." In other words $\pi_X(u)$ equals the possibility that $X = u$, given that "X is F." ■

**Lemma 7.** Consider a fuzzy synonym-set (synset) $S_k$, a variable $U_{i,k,W}$ taking values in the universe of discourse $WLD(W)$, and the $\Pi(U_{i,k,W})$, associated with $S_k$ and $U_{i,k,W}$. Then, $\pi_{U_{i,k,W}}(u_{i,k})$ the possibility that $U_{i,k,W} = u_{i,k}$, given that "$U_{k,C,W}$ is in $S_k$," is postulated to be equal to $\mu_{S_k}(u_{i,k})$ (i.e. $\pi(U_{i,k,W} = u_{i,k} | S_k) \triangleq \mu_{S_k}$).

**Proof.** A direct result of them Lemma 6. ■

**Theorem 4.** Consider an arbitrary infinitely-large corpus $C$, a precise WSD $W$, and a probable $S_k$. If in $C$, the usage of each wordsense / synset is independent of the usage of other wordsenses / synsets.

(a) If the Shafer's consonant belief function is adopted as the necessity measure, then, for any $u_{i,k} \in S_k$, we almost surely, will have $\mu_{S_k} = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$.

(b) If the 3 constraints of DP-consistency, preference-preservation, and maximally specificity have to be satisfied, then, for any $u_{i,k} \in S_k$, we almost surely, will have $\mu_{S_k} = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right)$.

**Proof.** (a) By Theorem 2, given the assumptions of part (a), we would have $\pi(u_{i,k} | S_k) = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. Also, the Lemma 7 implies that $\mu_{S_k} \triangleq \pi(U_{i,k,W} = u_{i,k} | S_k)$. Thus, we have $\mu_{S_k} = \pi_{1983}^{(C,W)}\left(u_{i,k}^{(W)}\right)$.

(b) By Theorem 3, given the assumptions of part (b), we have $\pi(u_{i,k} | S_k) = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. Also, the Lemma 7 implies that $\mu_{S_k} \triangleq \pi(U_{i,k,W} = u_{i,k} | S_k)$. Thus, we have $\mu_{S_k} = \pi_{1993}^{(C,W)}\left(u_{i,k}^{(W)}\right)$. ■

### 3.1.2 Pseudocode of the algorithm.

In the following, you see the pseudocode of the algorithm. In the following pseudocode, the input of the algorithm is a corpus of documents of a specific natural language (e.g. English) that is analyzed by a WSD. The analyzed corpus is called the WSF (Words Sense Frequency) matrix, as a 2-dimensional matrix. The 1st and the 2nd dimensions of WSF matrix represent synsets and wordsenses, respectively. Then, each cell of WSF represents the frequency of the corresponding wordsense in the whole inputted corpus.

Then, the output of the algorithm is the fuzzified version of the WordNet, engaged with the utilized WSD system. It is worthy to remind that the following algorithm is language-independent and can be applied on any natural language (given a large corpus and a WSD system of that natural language).

*//PMV, Possibility1983, and Possibility1993 stand for Probability Mass Value, Possibility (v.83) mass value, and Possibility (v.93) mass value, respectively. Dimensions are the same as what in WSF.*

```
For i = 1 to total number of synsets
  synSize = numberOfWordsenses(synset[i]);
  totalFrequencyOfSynset = synSize;
  For j = 1 to synSize
    totalFrequencyOfSynset += WSF[i][j];
  For j = 1 to synSize
    PMV[i][j] = (WSF[i][j]+1) /
                totalFrequencyOfSynset;
  For j = 1 to synSize
    possibility1983OfJ = 0;
    possibility1993OfJ = 0;
    pIJ = PMV[i][j];
    For m = 1 to synSize
      pIM = PMV[i][m];
      possibility1983OfJ += min(pIJ,pIM);
      possibility1993OfJ += piecewise(pIM <= pIJ , pIM , 0);
    FuzzyWordNet1983[i][j] = possibility1983OfJ;
    FuzzyWordNet1993[i][j] = possibility1993OfJ;
```

Please note that the above pseudocode utilizes the auxiliary technique of smoothing for bypassing the realistic limitations, occurring when the frequency of some wordsenses in the corpus in zero. This is the reason why `totalFrequencyOfSynset` is initialized by `synSize`, as it is assumed that each wordsense of a synset is visited once before analyzing the corpus.

For a better understanding of the mechanism of the algorithm, please consider the following hypothetical example. Suppose that, in a text corpus, the frequency of the *Synset(flower.n.02)* is as follows

*Synset(flower.n.02): {(WS(flower.n.02.flower), 54), (WS(flower.n.02.bloom), 24), (WS(flower.n.02.blossom), 19)}*

The mentioned algorithm, as the first step, computes the probability of each wordsense (after smoothing the frequencies by +1)

*Prob(flower.n.02): {flower, 55/100=0.55), (bloom, 25/100=0.25), (blossom, 20/100=0.2)}*

Then, it converts the probabilities to possibilities by the 1983 and 1993 formulas

*$Poss_{1983}$(flower.n.02): {flower, 0.55+0.25+0.2=1), (bloom, 0.25+0.25+0.2=0.7), (blossom, 0.2+0.2+0.2=0.6)}*

*$Poss_{1983}$(flower.n.02): {flower, 0.55+0.25+0.2=1), (bloom, 0+0.25+0.2=0.45), (blossom, 0+0+0.2=0.2)}*

The above possibilities have been proven to be the same as the membership degrees. Therefore, we have

*$FuzzySynset_{1983}$(flower.n.02): {flower, 1), (bloom, 0.7), (blossom, 0.6)}*

*$FuzzySynset_{1983}$(flower.n.02): {flower, 1), (bloom, 0.45), (blossom, 0.2)}*

**Conditions for the validity of the algorithm.** Although we proposed a proof for the validity of the results of the abovementioned algorithm, considering the real-world experiments-limitations, the above pseudocode produces the accurate membership values of the predefined synsets of the lexical database associated with the utilized WSD algorithm if and only if two conditions are satisfied.

- **Condition 1:** Corpus is large enough to provide accurate probability values, as a basis for membership functions. This is because the corpus has to be large enough to satisfy the law of large numbers (utilized in the first step of the proof).
- **Condition 2:** WSD algorithm works precisely so that the recognized wordsenses will be trustable. This is because: the $f\left(u_{i,k}\right)$ function is fed by the output of WSD algorithm and if it does not work properly, the results in all the next steps will be corrupted.

### 3.1.3 Applying the algorithm to the standard WordNet

To apply our algorithm to the English language, as the algorithm input we use the English corpus "Open American National Corpus" (OANC (Fillmore, Ide, Jurafsky, & Macleod, 1998), comprising almost 16.6 million words (Fillmore et al., 1998)(de Melo, Baker, Ide, Passonneau, & Fellbaum, 2012)) and the well-known graph-based Word Sense Disambiguation algorithm "UKB." We publish the entire list of English fuzzy synsets for both versions (v.83 and v.93) online. It can be found at http://dmls.iust.ac.ir/CogLing/FWN.zip. It is necessary to note that the published synsets are the same as the standard Princeton WordNet and the contribution of this dissertation in fuzzy-membership annotation of them is 100%.

It is necessary to note that, for a part of the produced / published fuzzy synsets, UKB detected no occurrence, in the entire 17M-word OANC corpus. Therefore, as mentioned in the pseudocode, after smoothing the frequencies by +1, the frequency of all of the wordsenses of such frequency-less synsets are considered as 1, and therefore, because all of the occurrences are the same, the membership degrees are also the same. An example of such synsets is the following fuzzy synset:

*acaroid_resin.n.01{'acaroid_resin': 1.0, 'accaroid_resin': 1.0, 'accroides': 1.0, 'accroides_resin': 1.0, 'accroides_gum': 1.0, 'gum_accroides': 1.0}*

However, about competence of UKB for our algorithm, satisfying the abovementioned second condition (WSD precision), it is worthy to note that the UKB has been evaluated in several outstanding research tasks including usage of WordNet for WSD (Agirre & Soroa, 2009)(Agirre E., Lopez de Lacalle O., 2014), WSD on medical domain (Martinez, Otegi, Soroa, & Agirre, 2014), improvements of Information Retrieval using WordNet (A Otegi, Xavier, & Eneko, 2011)(Arantxa Otegi, Arregi, Ansa, & Agirre, 2014), Word Embedding[24] on WordNet (Goikoetxea, 2015), etc. It is also worthy to remind that the proposed algorithm (for producing fuzzy synsets) is language-free and the interested researcher can apply it to his favorite language.

For validation of the produced results in this section, we would require applying them to one of the Text Mining applications. However, for the purpose of applying the produced fuzzified synsets, we would require a Text Mining model that can be fed by the produced fuzzy synsets. In subsection 1.1.3, we reviewed a research line of the existing Text Mining models. Correspondingly, in section 3.3, we propose two Text Mining models, which can be fed by the extra fuzzy information, provided in this section.

However, the main target of this section was presenting a Type-1 fuzzification algorithm for WLDs. In the next section, we go one step further and discuss the still-remained uncertainties and provide an approach for creating the interval version of the fuzzified WLDs.

## 3.2 Toward Interval-fuzzification of WordNet-like lexical databases

In this section, we briefly discuss the lack of information in the standard fuzzy synsets as well as how they can be covered by an interval version of fuzzy synsets, and then propose our algorithm for constructing such interval fuzzy synsets.

As mentioned, the existing studies on fuzzy synsets, consider them as standard fuzzy sets that assign a scalar membership degree to wordsenses of a synset (the membership function of a wordsense '$x$' of synset '$S$' is defined as $\mu_S(x)$: Synset → [0,1]). However, for precisely assigning a $\mu_S(x)$ to $x$, the following uncertainties should be considered:

**[Method-uncertainty]** The uncertainty associated with the methods by which $\mu_S(x)$ values are computed. This relates to the field of Interval Type 2 Fuzzy Sets (Mendel & Wu, 2010) that is going to be addressed in the future work of this study.

**[Context-uncertainty]** The various expectations of the possibility of occurrence of $x$ as a member of $S$ according to the context in which the wordsense is being used, or the nationality, ethnicity ... of the writer/speaker as the effects of different contexts on him.

**[Subject-uncertainty]**

[Intra-uncertainty] The uncertainty of a subject while judgment (Mendel, 1999) about the compatibility of a wordsense with its synset definition.

[24] Produced with random walk

[Inter-uncertainty] Different judgments of different people (Mendel, 1999) on the compatibility of the same wordsense and its synset definition, considering their different manner of thinking.

For taking each of the mentioned uncertainties into account, the range of $\mu_S(x)$: Synset → [0,1] should be upgraded to [0,1]×[0,1] to represent the membership by a fair interval; for assigning such interval we should deal with the following "tradeoff": From the one hand, it should include the memberships, related to different contexts, judgments …, and from the other hand, it should exclude the rare happening contexts, judgments ... In this section, we use the following two tradeoff strategies: (1) While dealing with context-uncertainty, we use the tradeoff strategy of including the "more-than-average" occurring membership values and excluding the less-than-average occurring ones: Considering an arbitrary wordsense ($ws_i$) of a Synset with $N$ different membership values, each of which representing its membership in a different context, the average of the $N$ membership values is considered as the lower membership degree and the maximum membership value as the upper membership. It is for including membership values, related to the "important contexts in which the wordsense is usually used" and excluding the membership values related to "non-important contexts in which the wordsense is only used from time to time" as they does not deserve widening the membership interval of a wordsense. (2) While dealing with subject-uncertainty (intra- and inter-uncertainty), we use the trade-off strategy of "average of intra over inter:" Uncertain judgment of each person (intra-uncertainty) on the wordsense compatibility with the synset-definition can be represented by a [lower, upper] membership degree. Avoiding the marginal personal views (marginal intra-uncertainty), we average the lower membership degrees over judgments of different subjects (inter-uncertainty) for reaching a fair (excluding the low marginal) lower membership, and follow the same, for a fair (excluding the high marginal) upper membership degree.

Nevertheless, for computing the membership value in each category, in this section, we approach computing the "Possibility" of the wordsense occurrence as a member of that synsets. (Zadeh, 1978) for the first time proposed the "Possibility theory" as a counterpart for "Probability theory" that deals with a fundamentally different type of uncertainty. In (Zadeh, 1978), he postulated that the possibility of occurrence of a member of a set is equal to its membership degree to that set ("possibility-membership equivalence"). Correspondingly, computing the [lower, upper] $\mu_{\mathrm{S}}(x)$, for $x \in S$ can be converted to computing a lower Possibility Mass Function (PMF) as well as an upper PMF for $S$.

As we mentioned in context-uncertainty (the uncertainty #2), $\mu_{\mathrm{S}}(x)$ varies by the context in which $x$ is used. Thus, we approach different context-customized lower and upper PMFs. If for the appearance of each synset in each context, we extract a PMF, then, by aggregating such context-customized PMFs, we can approach the required fair lower/upper PMFs of that synset. For this purpose, the next question will be how to extract the PMF of a synset in a given context. A standard method for computing PMFs is transforming probability mass function to possibility mass function. (D. Dubois et al., 1993) propose the most informative probability to possibility transformer

$$\pi(c_i) = \sum_{c_j | p(c_j) \le p(c_i)} p(c_j) \qquad \text{Equation 3. 1}$$

where $\pi(c_i)$ stands for the possibility mass value of the member $c_i$ and $p(c_i)$ stands for its probability mass value, where the summation is over all of the possible members. Now, we can find context-based membership degrees of a wordsense by computing its probability mass value in various contexts. If, in a corpus of documents of a context, a synset has a high frequency of wordsenses, then probability mass values can be estimated by their relevant frequency (considering the $Synset$={$ws_1, ws_2$,... , $ws_M$}, if the $ws_i$ frequencies in the corpus are represented by $\{f_1, f_2, \dots, f_M\}$, then if F=$\sum_{i=1}^{M} f_i$ is large enough, we can estimate the probability mass function of "$Synset$" as {$\frac{f_1}{F}$,$\frac{f_2}{F}$,...,$\frac{f_M}{F}$}). Finding the wordsenses frequency ($f_i$ values) of different synsets is subject of WSD (In cognitive and computational linguistics, WSD is an algorithm that, getting a multi-sense word in a sentence as input, identifies which of its senses is used in that sentence (Weaver, 1955)[25] (Mihalcea, 2011)).

We can represent the steps of our algorithm as follows:

Applying WSD on $N$ context-based corpora produces $N \times S$ context-based frequency list $\{f_1, f_2, \dots, f_M\}$ (assuming having a precise enough WSD).

$\left\{\frac{f_1}{F}, \frac{f_2}{F}, \dots, \frac{f_M}{F}\right\}$ yields $N \times S$ context-based Probability Mass Functions.

Applying Equation 3.1, yields $N \times S$ Context-based Possibility Mass Functions.

Based on the "possibility-membership equivalence postulation" of Zadeh, the output of step 4 is considered as $N \times S$ context-based membership functions.

Following the "more than average occurring" tradeoff/aggregation strategy, the $N$ context-based membership functions of each synset will be aggregated into 1, finally producing $S$ interval-valued ([Lower, Upper]) Membership Functions.

($N$ stands for the number of categories and $S$ stands for the total number of synsets.)

For a better understanding of the mechanism of the algorithm, please consider the following hypothetical example. Suppose that, in a text corpus with the 3 categories "social," "politics," and "economics," the frequency of the *Synset(flower.n.02)* is as follows

*Synset(flower.n.02)*

*Social: {(WS(shake.v.01.shake), 55), (WS(shake.v.01.agitate), 25)}*

*Politics: {(WS(shake.v.01.shake), 45), (WS(shake.v.01.agitate), 35)}*

*Economics: {(WS(shake.v.01.shake), 35), (WS(shake.v.01.agitate), 35)}*

The mentioned algorithm, as the first step, computes the probability of each wordsense in each category

*Probability$_{Social}$(shake.v.01): {shake, 55/80=0.69), (agitate, 25/80=0.31)}*

[25] To the best of our knowledge, this reference is the first publication that addresses automatic wordsense disambiguation.

*Probability$_{Politics}$(shake.v.01): {shake, 45/80=0.56), (agitate, 25/80=0.44)}*

*Probability$_{Economics}$(shake.v.01): {shake, 30/65=0.46), (agitate, 35/70=0.54)}*

Then, it converts the probabilities to possibilities by the Equation 3.1

*Possibility$_{Social}$(shake.v.01): {shake, 0.69+0.31=1), (agitate, 0.31+0.31=0.62)}*

*Possibility$_{Politics}$(shake.v.01): {shake, 0.56+0.44=1), (agitate, 0.44+0.44=0.88)}*

*Possibility$_{Economics}$(shake.v.01): {shake, 0.46+0.46=0.92), (agitate, 0.46+0.54=1)}*

The above possibilities have been proven to be the same as the membership degrees. However, as mentioned, for computing a final interval for membership degrees, we should have the average membership degree of each wordsense. As it can be seen, the average membership degree of *WS(shake.v.01.shake)* is (1+1+0.92)/3 = 0.97 and the average membership degree of *WS(shake.v.01.shake)* is (0.62+0.88+1)/3 = 0.83. Thus, following the more-than-average strategy, we have

*IntervalFuzzySynset(shake.v.01):*

$$\{(shake, \left[\min_{x\geq 0.97}\{1,1,0.92\}, \max\{1,1,0.92\}\right]), (agitate, \left[\min_{x\geq 0.83}\{0.62,0.88,1\}, \max\{0.62,0.88,1\}\right])\}$$

Therefore, we have

*IntervalFuzzySynset(shake.v.01): {(shake,* [1,1]*), (agitate,* [0.88,1]*)}*

Nevertheless, after fully presenting the Type-1- and Interval- fuzzification ideas for the existing WLDs, it is the turn to utilize them. However, considering that the proposed idea for the (interval) context-uncertainty is a preliminary algorithm and its results are validated by comparison with the measurements provided by native English speakers, we suffice to propose the Text Mining models that deal with the fuzzy synsets. We would like to remind that the proposed WLD-fuzzifier is globally utilizable in any language and recommended to the Text Mining society.

## 3.3 An Auxiliary Text Mining Model for Evaluating WordNet Fuzzifiers

In this section, we intend to propose two Text Mining models. However, please note that we do not claim the proposed Text Mining models to outperform all the Text Mining models in state-of-the-art, but we intend to show that they work better than their crisp version using standard non-fuzzy synsets. Thus, the proposed Text Mining models are only intended to prove the superiority of Fuzzified WordNet over its crisp counterpart[26].

[26] Albeit, in the experiments section, we also demonstrate that the provided accuracies are near, even, to the accuracy of very high time-complexity counterparts.

**Bag of Fuzzy Synsets (BoFS).** One of the two main proposed Text Mining models in this section is BoFS, as the fuzzified version of the BoS model. Considering that, in the introduction, fuzzy synsets are fully introduced, the BoFS model can be easily understood. However, for clarification, we pursue the example presented in subsection 1.1.3. Consider Table 3.1 as the fuzzified version of Table 1.1, in which the membership degree of each wordsense is specified.For modeling the (same) example presented while introducing BoS in subsection 1.1.3, instead of adding the occurrences of each synset, we add the membership of each wordsense. Therefore, we model it as

**Table 3.1. List of some fuzzy synsets with hypothetical synsets ID.**

| Synset ID | Word senses |
|---|---|
| **S1** | think (1), opine (0.7), suppose (0.8), imagine (0.4), reckon (0.2), guess (0.5) |
| **S2** | plant (1), flora (0.8), plant life (0.7) |
| **S3** | flower (1), bloom (0.7), blossom (0.6) |
| **S4** | falsely (1), incorrectly (0.9) |
| **S5** | merely (1), simply (0.5), just (1), only (1), but (0.8) |
| **S6** | part (1), portion (0.7) |

*{(S1, 2.6), (S2, 2), (S3, 2.3), (S4, 1.8), (S5, 1), (S6, 1)}.*

**Bag of Fuzzy Wordsenses (BoFWS).** Following a similar idea to what proposed in BoFS, an analogous Text Mining model can be defined by using wordsenses, instead of synsets, called BoFWS. It can be considered as the fuzzified version of Bag of Wordsenses (BoWS), firstly proposed in (Smeaton, 1995). Although BoWS never enjoyed great success, we believe that this is due to the fact that it assigns similar weights to all the wordsenses. We believe that the fuzzification of this model would solve this problem and returns BoWS to competence. We use the same illustrative example explained above and represent it by BoFWS.

*{(WS(think.v.02.think), 1), (WS(think.v.02.suppose), 1.6), (WS(plant.n.02.plant), 2), (WS(flower.n.02.flower), 1), (WS(flower.n.02.bloom), 0.7), (WS(flower.n.02.blossom), 0.6), (WS(falsely.r.02.incorrectly), 1.8) , (WS(merely.r.01.only), 1) , (WS(part.n.02.part), 1)}.*

As you can see, the wordsenses are considered as features of the text document (such as what in BoWS) but the frequency of each wordsense is multiplied in its membership degree. For example, the frequency of the wordsense *WS(falsely.r.02.incorrectly) is 2; but it is multiplied in 0.9 to adopt 1.8 as its value.*

In addition to BoFS and BoFWS, and along the reviewed evolution line of the Text Mining models, another fuzzy Text Mining model can be devised. By the same algorithm, proposed in section 3.1, and simply by substituting $S_k$ (k$^{th}$ synset) by $C_k$ (the k$^{th}$ concept which encompasses a number of synsets), we can fuzzify the concepts defined over a WLD to produce the Bag of Fuzzy Concepts (BoFC) model, as a fuzzification of Bag of Concepts. BoFC can be produced in the same way as we fuzzified BoS to reach BoFS. We let addressing these two issues (fuzzified concepts and BoFC) to be addressed in an individual study.

In section 4.1, we evaluate the Fuzzy Synsets by means of the proposed auxiliary Text Mining models and standard classifiers in the state-of-the-art.

Nevertheless, as mentioned, another necessary component of the pTER system is proposing the classifiers which enhance the classification accuracy in the presence of uncertainties which normally occur in the Social Network extracted information, due to their sparsity. We recall that due to the request of the H2020 IDENTITY project (as one of the funding providers of this dissertation), the text classifiers provided in this dissertation are focused on the Authorship Attribution field of science. However, as pointed out before, they can be simply utilized in the Sentiment Analysis field of science, just by modifying the utilized Text Mining model.

## 3.4 Uncertainty-handling in Text Classification

As mentioned in subsection 1.1.5, for Authorship Attribution purposes, we choose the well-known sn-gram model as the text representation model, as it is proven to outperform the other state-of-the-art models in Authorship Attribution of sizable documents.

After choosing and fixing the Text Mining model, for introducing the desired classifier, as promised, it is the turn to develop the classifiers (with pattern recognition approach) that can handle the uncertainty existing in Authorship Attribution problem.

Before proposing the classifiers, we have to enumerate the possible uncertainty-dealing cases in the attribution or equally the classification process. For enumerating such uncertainties, we first take a look at the logic of classification. Classification can be considered as a black box which inputs and outputs the information, represented in Table 3.2.

**Table 3.2. A symbolic view of the input and output of a classification problem.**

| | Input | Output |
|---|---|---|
| Train | $\left(\left(f_{i,j}, j = 1 \dots m\right), c_i, i = 1 \dots n\right)$<br>$c_i \in \{c_1, c_2, \dots c_p\}$ | # |
| Test | $\left(f_j, j = 1 \dots m\right)$ | $c_i \in \{c_1, c_2, \dots c_p\}$ |

In Table 3.2, $f_{i,j}$ stands for the value of the feature $j$ of the training document $i$, $c_i$ stands for the class (author) label of that document, $m$ stands for the total number of features, $n$ for the number of training items, and $p$ for the number of possible classes (existing authors).

Regarding the mentioned black-box view to classification, four categories of uncertainties are assumable. Three of these views are related to data (input/output) and the fourth one is related to the intrinsic uncertainty existing in the classifier black box itself. We name the first three uncertainties as: "only I," "only II," and "I AND II," where I and II are defined as (I) the number of the available features is low (i.e. $m$ is low[27]) and (II) the number of the training data belonging to each class (author) is low (i.e. $n/p$ is low).

---

[27] Please note that the low value of $m$ (related to type I) can be arisen either from smallness of document content size (e.g. short documents such as tweet) or requirement to a faster attribution that can be done by few number of features. While addressing the type I uncertainties (I and I & II), we conduct our experiments on the latter.

In the rest of this section, we address each of them and introduce appropriate classifiers for the underlying types of uncertainty.

### 3.4.1 Small $m$ and $n/p$

In many studies, such as the famous study of (Rennie, Shih, Teevan, & Karger, 2003) or (Ting, Ip, & Tsang, 2011), it is proven that Naïve Bayes classifiers can function as well as the other well-known classifiers such as SVM or Decision Tree if the classifier specifications (e.g. feature models, smoothing, etc.) are set appropriately. On the contrary, there also exist other studies, such as the well-known study of (Kibriya, Frank, Pfahringer, & Holmes, 2004) which claims that although Naïve Bayes is very effective regarding its simplicity and speed, it is not necessarily the best when the only important factor is accuracy. These two contradicting claims can be resolved by claiming that Naïve Bayes can function as well as the other well-known classifiers such as SVM in the presence of uncertainty and have weaker accuracy otherwise.

In this subsection, we proceed one step further and claim that given high-quality enough specifications, Naïve Bayes classifiers can even outperform the others if an appropriate Text Mining model and an appropriate Probability Distribution Function are chosen at the same time. Considering that the Text Mining model is fixed as “sn-gram” that is proven to be the best for the Authorship Attribution of the large documents, in the following, we explain two appropriate Probability Distribution Functions.

The first one (i.e. Poisson Naïve Bayes or PNB) addresses a fast and accurate classifier and does <u>not</u> intend to prove the abovementioned claim[28]. But the second classifier (i.e. Weibull Naïve Bayes or WNB) is going to prove the mentioned claim, because from the one hand the Weibull distribution is very fit for Text Mining, and from the other hand, this distribution is very flexible, thanks to its two parameters. The better performance of WNB than PNB (which can prove the mentioned claim) is at the price of increasing its complexity from linear to polynomial.

#### *Poisson Naïve Bayes Classifier*

As mentioned before, Naïve Bayes would become a classifier only after determining the corresponding probability distribution. Different Probability Distributions have been adopted for Naïve Bayes-based text classifiers in the state-of-the-art. However, from the one hand, here, we intend to choose a Probability Distribution for which a linear-complexity estimator exists, and from the other hand, the chosen Probability Distribution must make the resulting Naïve Bayes classifier competitive with the other Authorship Attribution classifiers utilized in the state-of-the-art.

Among such Probability Distributions, Poisson Probability Distribution is an informative one utilized in Text Mining classifiers. The most well-known study in which Poisson is presented to be utilized in Naïve Bayes text classifiers is the study of (S. B. Kim, Han, Rim, & Myaeng, 2006)[29] who propose it in combination with feature normalization based on text document length.

[28] “Given high-quality enough specifications, Naïve Bayes classifiers can even outperform the others.”

[29] and its conference version (S.-B. Kim, Seo, & Rim, 2003) in 2003

However, despite the existence of such fast and accurate classifier in Text Classification area, PNB has not had much success in Author Attribution, due to its low output quality. For instance, the results reported in the well-known study of (Airoldi, Anderson, Fienberg, & Skinner, 2006) in Authorship Attribution show the superiority of Multinomial Bayes (MB) (the so-called Multinomial Naïve Bayes) over the Poisson-based version[30].

Although Poisson Probability Distribution appears inefficient in those studies, the mentioned inefficiency does not represent the weakness of Poisson for Authorship Attribution. Indeed, it shows its weakness for modeling *"the probability that an author utilizes a word"* when the text documents are modeled by the utilized Text Mining models for Authorship Attribution. This fact becomes apparent when we see that the same classifier provides accurate results in Text Classification. We believe that this fact is because the utilized text models in those studies are not trustable enough fingerprints for Author Stylome.

The proposed hypothesis will be proved if PNB provides accurate Authorship Attribution results when sn-gram (as a trustable Text Mining model for the Stylome fingerprint) is adopted. We let the evaluating experiments of PNB to be addressed in section 4.3.

### *Weibull Naïve Bayes Classifier*

The Weibull distribution is very appropriate for modeling the frequency of text document features for two main reasons: (1) its appropriate domain ($[0,\infty)$) and unimodal nature and (2) the presence of two tuning parameters that make it very flexible to fit the training data (Figure 3.1). In addition, it is empirically shown that "the frequency distribution of bigrams is approximated by the Weibull distribution" (S. Kim, Yoon, & Song, 2001). However, Weibull distribution has never been utilized as the probability distribution of a Naïve Bayes classifier for text classification. Although the Weibull distribution has been welcomed in text classification (mainly since 2014 when the well-known study of (Scheirer, Jain, & Boult, 2014) on Weibull-calibrated SVM was proposed), this distribution is not utilized for naïve Bayes text classification.

[30] The Arabic Stylometry study of Altheneyan and Menai (Altheneyan & Menai, 2014) also proposes the superiority of the other version of Bernoulli-based Naïve Bayes over the Poisson-based version.

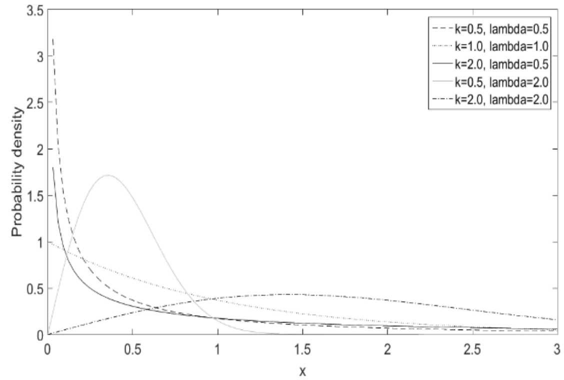



**Figure 3.1 The Weibull probability distribution function for different values of $\boldsymbol{\lambda}, \boldsymbol{k}$.**

Thus, considering the already competitive performance of Naïve Bayes with the other well-known counterparts, upgrading Naïve Bayes by the Weibull distribution is expected to make Naïve Bayes even superior to its counterparts, in the presence of the addressed high uncertainty situation.

For presenting WNB, one of the most important components is proposing or utilizing a high-quality parameter estimation technique for Weibull probability distribution. Such an estimator, given a series of observations, estimates $\lambda, k$ in the following Weibull Probability Distribution Function

$$f_{\lambda,k}(x) = \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{\left(-\frac{x}{\lambda}\right)^{k}} \qquad \text{Equation 3.2}$$

In Equation 3.2 $x \in [0, \infty)$ and $\lambda, k > 0$ are called scale and shape parameters, respectively. Method of Moments (MoM) estimator (Nwobi & Ugomma, 2014) is one of the best parameter estimators for the Weibull distribution that can estimate $\lambda, k$ well even in the presence of uncertainty. Therefore, we adopt this estimator for WNB.

So far, we have set the classifier (Naïve Bayes), the probability distribution (Weibull), and an accurate estimator (MoM) required for the probability distribution. It means that WNB classifier is fully specified, and as mentioned, is expected to outperform its counterparts in the presence of the addressed uncertainty. The experiments for proving the efficiency of this classifier as well as the other discussed classifiers in this section will be discussed in section 4.4.

### 3.4.2 Large $m$ and small $n/p$: Weibull and Multinomial Bayes classifiers

When the $n/p$ is small but the $m$ value is high, Naïve Bayes classifiers, albeit still function acceptably, suffer from too many naïve simplifying assumptions of conditional independence of features. The potential error caused by this naïve assumption increases with the high values given to $m$.

Intrinsically, simplifying assumptions improve the performance in the presence of uncertainty if they are not too many so that the small errors caused by each of them (arisen from the simplifying assumption) are aggregated and this negative effect becomes more than the positive effect of their simplifying role. In other words, such assumptions can be unhelpful or even harmful when they are too many.

The naïve conditional independence assumption -for simplification purposes- assumes independence of each feature pair from the others, which means $\binom{m}{2}$ assumptions. Although (H. Zhang, 2004) proves that the conditional independence simplifying assumptions cancel each other under some circumstances and thus would function as well as their non-naïve version, such circumstances does not always occur. That is the reason why, despite its efficiency, Naïve Bayes is shown in several studies not to be the most accurate classifier.

However, this problem belongs only to the "naïve" assumptions, but, Bayesian techniques are intelligent enough to model the problems even in the presence of uncertainty (Meyniel et al., 2015). Therefore, in such circumstances, the "non-naïve" version of the same classifiers can be considered as appropriate options because they avoid the mentioned drawback.

However, "non-naïve" Weibull Bayes classifier is a rather complex probabilistic classifier that requires an independent study. But, still, an alternative (non-Weibull) non-naïve option can function better than WNB which deals with large $m$ (regarding the abovementioned problem).

Fortunately, there already exists a very well-known non-naïve Bayes classifier: Among the so-called Naïve Bayes classifiers, Multinomial Naïve Bayes is not really naïve. It is because it does not include the naïve assumption of conditional independence. The Multinomial Probability Distribution Function, which is explicitly utilized in Multinomial Naïve Bayes, is $\frac{z!}{x_1!x_2!\ldots x_m!}p_1^{x_1}p_2^{x_2}\ldots p_k^{x_m}$ where $x_i$ stands for the frequency of the feature $i$, $p_i$ for the probability distribution parameter related to that feature, and $z = \sum_{j=1}^{m} x_i$. At the first glance, this distribution function may be looked like the multiplication of the independent Probability Distribution Functions $\sqrt[m]{z!}\frac{p_i^{x_i}}{x_i!}$. But, $\sqrt[m]{z!}\frac{p_i^{x_i}}{x_i!}$ functions are not independent due to the presence of the factor $z = \sum_{j=1}^{m} x_i$. Indeed, although in Multinomial distribution the trials are independent, their outcomes are dependent to each other by the factor $z$. Therefore, this classifier should be called Multinomial Bayes (MB) rather than Multinomial Naïve Bayes. Thus, although the recommended classifier for the uncertainty that is addressed in this subsection is Weibull Bayes, considering that Weibull Bayes requires a separate mathematical study to be developed and introduced, for now, Multinomial Bayes is alternatively recommended.

It is very important to note that although Multinomial Bayes is our alternative recommendation, "revisiting" Multinomial Bayes should be considered as a recommendation of this study; because no research in the state-of-the-art recommends Multinomial Bayes as the best classifier, whereas, we claim it as the best option in occurrence of the addressed uncertainty (large $m$ and small $n/p$).

### 3.4.3 Small $m$, large $n/p$: Weibull Naïve Possibilistic Classifier

When $n/p$ is large and correspondingly the amount of the training data is high, Naïve Bayes classifiers are not recommended anymore. However, recently, (Bounhas et al., 2014) propose a "possibilistic" generation of Naïve Bayes classifiers called Naïve Possibilistic Classifier (NPC) which is demonstrated to outperform the state-of-the-art probabilistic classifiers, if the training phase of the classifier includes uncertainties. But, considering that the addressed uncertainty in this subsection (small $m$ and large $n/p$) assumes having large enough training data, the amount of train information required for estimating the $\lambda, k$ parameters is enough and no uncertainty is seen at the first glance.

*The intrinsic uncertainty of WNB.* Despite the abovementioned fact, in the training phase of WNB, there exists a hidden type of uncertainty. We can approach estimation of the Weibull parameters by different estimators. MoM (Nwobi & Ugomma, 2014), Approximated MoM (AMoM) (Heo, Salas, & Kim, 2001), Maximum Likelihood Estimator (MLE) (Skinner, Keats, & Zimmer, 2001), Least Square (LS) (which has Bernard and Herd-Johnson versions (L. Zhang, Xie, & Tang, 2008)) are the most well-known estimators of Weibull distribution (Nwobi & Ugomma, 2014). However, although MoM is the best estimator for the case in which the number of training data is low, the other estimators such as MLE and LS also function well when the number of training data is high. An interesting fact here is: By estimating different numerical inputs by the mentioned classifiers, we notice that although they are all effective estimators, the outputted $\lambda, k$ parameters of them are not the same.

In other words, when the training data is sufficient, despite the effectiveness of the $\lambda, k$ values, they do not converge to a golden $\lambda, k$. It means that the estimators are outputting different locally optimal parameters rather than a globally optimal $\lambda, k$. We call this type of uncertainty "the intrinsic uncertainty" of the classifier. Because this uncertainty also occurs in the training phase and is related to the parameter estimation, the idea of (Bounhas et al., 2014) can be applied for upgrading the WNB classifier for this type of uncertainty.

*Log-linear time complexity.* Now that we have the opportunity of utilizing more than one estimator (all of which functioning well), we adopt a smart strategy for this choice. Among the mentioned estimators, MoM and MLE (while estimating the $\lambda, k$) require solving a nonlinear equation, which imposes a time complexity burden to the training phase. Thus, by avoiding these two estimators among the five mentioned estimators, we utilize the AMoM, LS (Bernard), and LS (Herd-Johnson) estimators which are all low-complexity estimators.

AMoM (Equations 11 and 14 in (Heo et al., 2001)) has a linear time complexity and LS Herd-Johnson and LS Bernard (Equations 3.3 and 3.4 in (L. Zhang et al., 2008)) only require sorting the feature frequency array and, therefore, have log-linear time complexity. Thus, if we adopt the AMoM, Herd-Johnson LS, and Bernard LS for developing NPC, we expect to have an accurate and fast (log-linear) classifier.

However, NPC (Bounhas et al., 2014) is designed based on the Gaussian probability distribution, whereas it is not appropriate for modeling the frequency of textual features, at all. Therefore, in this subsection, we propose a version of NPC that is based on Weibull distribution and name it as "Weibull Naïve Possibilistic Classifier" (WNPC). In the following, the training and testing steps of WNPC are explained in details. Figure 3.2 represents a schematic view of the training steps.

*Training Step 1:* For an arbitrary class $c_i$ and feature $f^{(j)}$, let $F_i^{(j)} = \left(f_{i,1}^{(j)}, f_{i,2}^{(j)}, \ldots, f_{i,n_i}^{(j)}\right)$ represent the frequency sequence of the feature $j$ in each of the $n_i$ documents labeled as $c_i$. Each $F_i^{(j)}$ is

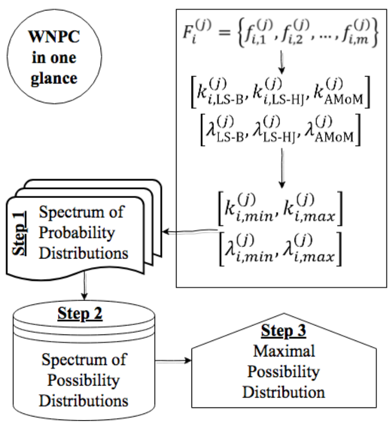
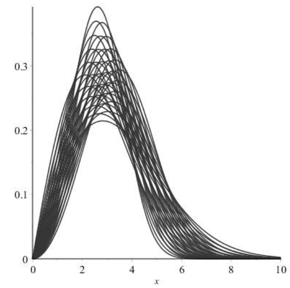
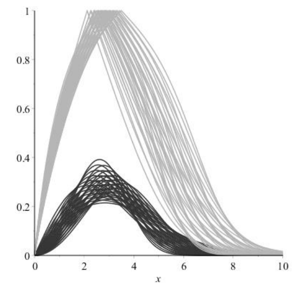


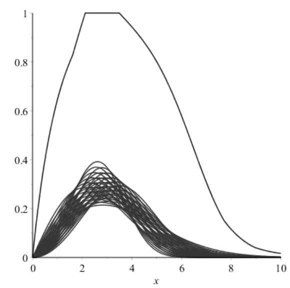

**Figure 3.2. A schematic view of the required steps for training WNPC based on the feature frequency set ($F_i^{(k)}$) of the feature $f^{(k)}$ in the documents of the class $c_i$.**



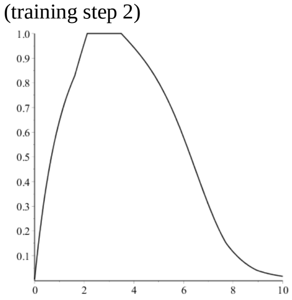
(a) A spectrum of Weibull Probability Distribution Functions, made by scrolling $k, \lambda$ in their confidence interval (training step 1).



(b) Converting Probability Distribution Functions to Possibility Distribution Functions by the mentioned transformation. (training step 2)



(c) Supremum of Possibility Distribution Function plots is chosen as the possibilistic projection of the Probability Distribution Functions spectrum (training step 3).



(d) Each [s2-gram, subject pair] is modelled by the achieved Possibility Distribution Function (Output of the steps).

**Figure 3.3. Steps of estimating the "possibility" distribution functions, required for training the Weibull Naïve Possibilistic Classifier.**

supposed to have Weibull distribution; therefore, we should estimate $\lambda$ and $k$ in Equation 3.2 based on the $F_i^{(j)}$ sample. However, due to the NPC logic (Bounhas et al., 2014), instead of estimating a real number for $\lambda$ and $k$, we have to estimate a confidence interval for them. We set them as $[\lambda_{min}, \lambda_{max}]$ and $[k_{min}, k_{max}]$ (we call them $[\lambda_1, \lambda_2]$ and $[k_1, k_2]$) where the $\lambda_1$, $\lambda_2$, $k_1$, and $k_2$ stand for the smallest and the largest values of $\lambda$ and $k$ in AMoM, LS Herd-Johnson, and LS Bernard estimators, respectively. Considering that each $(\lambda, k)$ represents one Probability Distribution Function, the combination of the values in the estimated $[\lambda_1, \lambda_2]$ and $[k_1, k_2]$ intervals provides a spectrum of (uncountably infinite) Probability Distribution Functions (Figure 3.3 (a)).

*Training Step 2:* Each estimated $(\lambda, k)$ represents a Probability Distribution Function. Based on the logic of NPC (Bounhas et al., 2014), we should convert Probability Distribution Functions to Possibility Distribution Functions. (Weng et al., 2012) provides a general formula by which the transformation of Weibull Probability Distribution Function to its Possibility Distribution Function can be extracted (Figure 3.3 (b)). The special case of that formula for our problem is represented in Equation 3.3 ($\pi_{\lambda,k}(x)$ stands for the corresponding Possibility Distribution Function).

$$\pi_{\lambda,k}(x) = \begin{cases} f_{\lambda,k}(x) & k = 1 \\ 1 - e^{\frac{-k}{\lambda}x} + e^{-\frac{k}{\lambda}f_{\lambda,k}^{(-1)}(x,-1)} & x \le \widehat{f_{\lambda,k}} \\ 1 + e^{\frac{-k}{\lambda}x} - e^{-\frac{k}{\lambda}f_{\lambda,k}^{(-1)}(x,0)} & x > \widehat{f_{\lambda,k}} \end{cases},$$

$$f_{\lambda,k}^{(-1)}(x,q) = \lambda \cdot e^{\frac{b_{\lambda,k}(x) - \left(W_q\left(l_{\lambda,k}(x,q)\right)\cdot(k-1)\right)}{k(k-1)}}$$

$$l_{\lambda,k}(x,q) = -\frac{k \cdot e^{\left(\frac{b_{\lambda,k}(x)}{k} - 1\right)}}{k-1},$$

$$b_{\lambda,k}(x) = k \cdot \left(k \cdot \ln\left(\frac{x}{\lambda}\right) + \ln\left(\frac{\lambda}{x}\right) - \left(\frac{x}{\lambda}\right)^k\right). \qquad \text{Equation 3.3}$$

In Equation 3.3, $W_0(x), W_1(x)$ stand for the two versions of the Lambert-W function, $e^x$ for the exponential function, and $\widehat{f_{\lambda,k}}$ for the mode of the $f_{\lambda,k}(x)$ Probability Distribution Function.

<u>*Training Step 3:*</u> After conversion of Probability Distribution Functions to Possibility Distribution Functions, there will be a spectrum of Possibility Distribution Functions, and regarding the logic of NPC (Bounhas et al., 2014), they should be converted to a unique Possibility Distribution Function, by simply applying a maximization over them (Figure 3.3 (c)).

$$\pi_{I,\lambda,k}(x) = \begin{cases} \max\left(\pi_{\lambda_1,k_1}(x), \pi_{\lambda_1,k_2}(x)\right) & x \le \widehat{f_{\lambda_2,k_1}} \\ 1 & \widehat{f_{\lambda_2,k_1}} < x \le \widehat{f_{\lambda_2,k_2}} \\ \max\left(\pi_{\lambda_2,k_1}(x), \pi_{\lambda_2,\frac{k_1+k_2}{2}}(x), \pi_{\lambda_2,k_2}(x)\right) & x > \max\left(\widehat{f_{\lambda_2,k_1}}, \widehat{f_{\lambda_2,k_2}}\right) \end{cases}$$

Equation 3.4 [31]

Following the proposed training steps, for each $\left(F_i^{(j)}, c_i\right)$ pair, a Possibility Distribution Function can be assigned, which models the possibility of the number of times that that $c_i$ utilizes the corresponding "feature" in its corresponding documents (Figure 3.3 (d)).

<u>*Kernel function.*</u> In the cases that the $F_i^{(j)}$ set has only one member (let us call it $f_i^{(j)}$) or its entire members have the same value, $f_i^{(j)}$, the parameter estimation techniques such as MLE, MoM, LS, etc. do not provide valid values. Therefore, in these cases, we use a "kernel" Probability Distribution Function and Possibility Distribution Function by setting $\lambda, k$ so that the max value of the $f_{\lambda,k}(x)$ occurs at $f_i^{(j)}$ (i.e. $\widehat{f_{\lambda,k}} = f_i^{(j)}$) and the value of the probability (and correspondingly

[31] The exact form in the third condition is $\max\left(\max_{k\in[k_1,k_2]} f_{\lambda_2,k}(x)\right)$. But, for decreasing the complexity, we substitute it by the mentioned expression. Analytically solving $\max_{k\in[k_1,k_2]} f_{\lambda_2,k}(x)$ can result in an upgraded version of it.

possibility) distribution function at $f_i^{(j)}$ becomes one (i.e. $f_{\lambda,k}\left(f_i^{(j)}\right) = 1$). By solving the mentioned system of two equations with two unknowns, we would have

$$\lambda = f_{i,j} \cdot \left(\frac{k}{k-1}\right)^{\frac{1}{k}}, \;\; (k-1) \cdot e^{\frac{1-k}{k}} - f_{i,j} = 0$$

<u>*Testing Step:*</u> After finishing the training phase, for each $F_i^{(j)}$, we would have one Possibility Distribution Function that represents $\pi\left(f^{(j)} \middle| c_i\right)$ or the possibility of $F = f^{(j)}$, given that $C = c_i$. Substituting the estimated Possibility Distribution Functions in the Equation 3.5[32], we have

$$\pi_{WNPC}\left(c_i \middle| f^{(1)}, f^{(2)}, \dots f^{(m)}\right) \propto \pi(c_i) * \Pi_{j=1}^{m} \pi_I\left(f^{(j)} \middle| c_i\right) \qquad \text{Equation 3.5}$$

Please note that the operator $*$ (and $\Pi$ as its extension) in Equation 3.5 is defined in the Possibility Theory and may be chosen as $min$ or *product*. Similar to (Bounhas et al., 2014), here we adopt it to be the product operator.

For finding the prior possibility of each class (i.e. $\pi(c_i)$) we first simply assign its prior probability to be its relevant frequency ($p(c_i) = fr_i / \Sigma_{i=1}^{n_i} fr_i$) where $fr_i$ stands for the frequency of documents labeled as $i$. Then, following the probability to possibility transformation of (D. Dubois et al., 1993), we have

$$\pi(c_i) = \Sigma_{s_a | p(c_a) \le p(c_i)} p(c_a), \qquad \text{Equation 3.6}$$

Now, having a document ($D_k$) for identification, we compute the Equation 3.5 for different classes, $c_i$

$$\pi(c_i | D_k) = \pi\left(c_j \middle| f_{1,D_k}, f_{2,D_k}, \dots f_{N,D_k}\right) \qquad \text{Equation 3.7}$$

Equation 3.7 is the value that WNPC assigns to the "possibility" of $D_k$ belonging to $c_i$. For the final classification, following the maximum a posteriori strategy, the WNPC output is determined.

In the next section, we study the computational complexity of the proposed algorithms along with the complexity of the corresponding state-of-the-art counterparts.

## 3.5 Complexity Analysis of Text Classifiers

In general, computing approach to Stylometrics is divided into literary authorship identification and machine-learning-based text classification (Chaski, 2013). The most common practice of Authorship Attribution is in supervised learning, in which, the textual documents of authors are modelled by their stylistic features; then, a classifier is trained by the known textual documents of candidate authors; and at the end, the trained classifier is used to determine the stylistically-closest author to the questioned document (Fridman et al., 2015).

[32] for more information refer to (Bounhas et al., 2014)

**Table 3.3. Time complexity and average value of the reported accuracies for the classifiers utilized in** (Sidorov et al., 2014)**. 'n' stands for the number of whole the training samples, 'm' for the features number of each item, and 'p' for the number of classes.**

| Method | Worst-case Time Complexity | Average value of the reported accuracies |
|---|---|---|
| **SVM-SMO** | $O(p \cdot n^3 \cdot m)$ | 100% |
| **C4.5 Classifier (WEKA J48)** | $O(n \cdot m^2)$ | 75% |
| **Multinomial Bayes** | $O(n \cdot m)$ | 63% |

### 3.5.1 The counterpart classifiers

To the best of our knowledge (Sidorov et al., 2014) is the most well-known state-of-the-art research in Authorship Attribution of sizable documents. In their evaluating experiments, they adopt the Machine Learning approach with the three classifiers proposed in Table 3.3. Correspondingly, and because the addressed experiments in sections 3.3 and 3.4 are also conducted on sizable documents (indeed, the same dataset that Sidorov et al. introduce and utilize), we let these three methods as counterparts and analyze their complexities, in the following.

The 3 utilized classifiers are: (1) SVM with Sequential Minimal Optimization (SMO) algorithm for training phase, (2) C4.5 classifier (John Ross Quinlan, 2014) (called J48 (Patil, 2013) in Weka data mining tool (Hall et al., 2009)), and (3) Multinomial Bayes with probability distribution (Hall et al., 2009). Although utilizing SVM has provided the best accuracy in (Sidorov et al., 2014) (errorless attribution), each of the utilized classifiers has a different time complexity and is suitable for special applications. Considering that the complexities are not reported in (Sidorov et al., 2014), we first briefly analyze them, here.

As in Table 3.2, we let '$n$' to stand for the number of whole the training samples, '$m$' for the features number of each item, and '$p$' for the number of classes; and overview the classifiers complexity one by one.

The first and the most successful classifier utilized in (Sidorov et al., 2014) is SVM utilizing SMO algorithm. Considering that, on the one hand, the worst-case time complexity of SMO (Platt, 1998) is $O\ (n^3 \cdot m)$ (for binary classification), and on the other hand, assuming one vs. all strategy for multiclass SVM, the classification will repeat $p-1$ times, each time removing training data of one class, for time complexity of multiclass SVM-SMO we will have the Equation 3.8.

$$O\left(\sum_{i=2}^{p}\left(i \cdot \frac{n}{p}\right)^3 \cdot m\right) = O\left(\left(\frac{n}{p}\right)^3 \cdot m \sum_{i=2}^{p} i^3\right) = \qquad \text{Equation 3.8}$$

$$O\left(\left(\frac{n}{p}\right)^3 \cdot m \cdot \left(\frac{p^4}{4} + \frac{p^3}{2} + \frac{p^2}{4} - 1\right)\right) = O\left(p^4 \cdot \left(\frac{n}{p}\right)^3 \cdot m\right) = O(n^3 \cdot p \cdot m),$$

Moreover, based on what (J. Su & Zhang, 2006) report, the time complexity of C4.5 Classifier (WEKA J48) is $O(n \cdot m^2)$, that can be written as $O(p \cdot n \cdot m^2)$. Moreover, the time complexity of Naïve Bayes Classifier with Multinomial distribution, considering that from the one hand, point estimation of the Multinomial distribution has $O(\frac{n}{p})$ solution (Murphy, 2006), and from the other hand it performs the training procedure, separately, for each class (totally $p$) and each feature (totally $m$), the corresponding time complexity will be $O(n \cdot m)$.

Now, let us have a brief look at the reported Authorship Attribution results of (Sidorov et al., 2014) in relation to their complexity. As you can see in Table 3.3, (although the choice of the Authorship Attribution system is a tradeoff between time complexity and accuracy) the three proposed algorithms have either relatively high complexity or relatively low accuracy. More in detail, a real Authorship Attribution application in which hundreds or thousands of candidate authors exist, the existence of $p \cdot n^3$ in the time complexity of SVM-SMO will be considered as its drawback. Or considering the C4.5 classifier, it is $m$ times slower than the Multinomial Bayes and considering that the standard number of $m$ is at least a couple of hundreds (e.g. 400 is the minimum reported in (Sidorov et al., 2014)), it means that C4.5 is at least hundred times slower than Multinomial Bayes. Considering the Multinomial Bayes, although it has the best possible (linear) time complexity, yet it provides relatively non-accurate results. As it can be seen, the above evaluation shows that the proposed methods although have had remarkable results and correspondingly high attention in the state-of-the-art of Author Attribution, they are not very appropriate for real-time applications. Thus, the PNB fast and accurate method (with linear time complexity) or WNPC (with log-linear time complexity), proposed in section 3.4 would be very welcome for real time or large data volume applications.

However, because PNB, WNB, and WNPC are the proposed Authorship Attribution in this dissertation, we let their results to be performed in section 4.3. But, we provide a complexity analysis of them in the following.

### 3.5.2 Weibull Naïve Possibilistic Text Classifier

Here, we again analyze the complexities by means of the black-box view proposed in Table 3.2. We assume that a classifier receives $c$ matrices (belonging to $c$ different classes) of $m \times \frac{n}{p}$ dimension, where their rows represent features, their columns represent the training documents of the related class (i.e. the related author), and their cells represent the feature frequency in the corresponding document.

Training a Naïve Bayes classifier is equal to estimating the parameters of the corresponding probability (or possibility) distribution for all of the $p \times m$ matrix rows. Thus, the time complexity of the training phase is $O(p \times m \times \text{Parameter\_Estimation})$.

As mentioned in the previous section, WNPC requires only sorting the $n$ values and, therefore, has the low computational complexity of $n/p \cdot \log(n/p)$ for parameter estimation. Thus, we would have $O\left(n \cdot m \cdot \log\left(\frac{n}{p}\right)\right)$ as the time complexity of WNPC.

The same argumentation can illustrate the complexity of the PNB as $O(n \cdot m)$.

### 3.5.3 Weibull Naïve Bayes Text Classifier

WNB requires solving a nonlinear equation (Eq. 2.20 in (Nwobi & Ugomma, 2014)).

$$g(x) = \frac{\Gamma^2\left(1+\frac{1}{x}\right)}{\Gamma\left(1+\frac{2}{x}\right)} - \frac{\mu^2}{\mu^2+\sigma^2}, x > 0 \qquad \text{Equation 3.9}$$

Considering that, in Equation 3.9, $g(x)$ is infinitely differentiable on $(0,\infty)$, it can be represented in Taylor series form. Therefore, we can approximate it by a truncated Taylor series $\sum_{i=0}^{t} a_i(x-x_0)^i$. However, truncated Taylor series are not precise approximations.

Based on the recently proposed study (Parand, Hossayni, & Rad, 2016), we know that if we let $V_t(y) = \left[B_{0,t}(y), B_{1,t}(y), \dots, B_{(t,t)}(y)\right]^T$ to stand for a "Bernstein polynomial" basis vector and adopt the inner product $\langle h_1(y), h_2(y)\rangle = \int_0^1 h_1(y)h_2(y)dy$ for constructing the inner-product function space $Span\big(V_t(y)\big)$, then the best approximation of $g(x)$ in $Span\big(V_t(y)\big)$ would be

$$g(x) = g\left(\frac{y}{1-y}\right) = a^T V_t(y) = \sum_{i=0}^{t} a_i B_{i,t}(y)\,,$$

$$\langle a^T V_t(y) - g\left(\frac{y}{1-y}\right), V_t^T(y)\rangle = \mathbf{0}_{t+1}$$

$$\Rightarrow a^T Q_t = \langle g\left(\frac{y}{1-y}\right), V_t^T(y)\rangle$$

$$\Rightarrow a^T = \langle g\left(\frac{y}{1-y}\right), V_t^T(y)\rangle\, Q_t^{-1},$$

$$Q_t[i,j] = \frac{\binom{t}{i-1}\binom{t}{j-1}(2(t+1)-(i+j))!(i+j-2)!}{(2t+1)!} \qquad \text{Equation 3.10}$$

where $x \in (0,\infty), y = \frac{x}{x+1}, B_{i,t}(y) = \binom{t}{i} t^i(1-t)^t - i$ (Parand et al., 2016).

For the sentence $\langle g\left(\frac{y}{1-y}\right), V_t^T(y)\rangle$, in Equation 3.10, due to the linearity of inner products, we can write

$$\langle g\left(\frac{y}{1-y}\right), V_t^T(y)\rangle = \langle \frac{\Gamma^2\left(1+\frac{1-y}{y}\right)}{\Gamma\left(1+\frac{2-2y}{y}\right)} - \frac{\mu^2}{\mu^2+\sigma^2}, V_t^T(y)\rangle$$

$$= \langle \frac{\Gamma^2\left(\frac{1}{y}\right)}{\Gamma\left(\frac{2-y}{y}\right)}, V_t^T(y)\rangle - \frac{\mu^2}{\mu^2+\sigma^2} V_t',$$

$$V_t' = \langle 1, V_t^T(y)\rangle = \left[\frac{m!}{(m+1)!}, m\frac{(m-1)!}{(m+1)!}, \dots, \binom{m}{i}\frac{i!(m-i)!}{(m+1)!}, \dots, \frac{m!}{(m+1)!}\right] \qquad \text{Equation 3.11}$$

In the Equation 3.11, the term $\langle \frac{\Gamma^2\left(\frac{1}{y}\right)}{\Gamma\left(\frac{2-y}{y}\right)}, V_t^T(y)\rangle$ is left to be computed by numerical integration. Please note that this sentence is data-independent (does not include any of $X_n$ members, $\mu(X_n)$, or $\sigma(X_n)$) and is computed only once, "before the experiments." Therefore, the complexity of this integration is excluded from the computational complexity of the method. Thus, the best approximation of $g\left(\frac{y}{1-y}\right)$ in $Span\big(V_t(y)\big)$, regarding the inner product $\langle h_1(y), h_2(y)\rangle$ can be approximated by

$$g(x) = g\left(\frac{y}{1-y}\right) \approx \sum_{i=0}^{t} a_i B_{(i,t)}(y) = a_t^T V_t(y) = a^T V_t\left(\frac{x}{x+1}\right),$$
$$a = \left(\langle \frac{\Gamma^2\left(\frac{1}{y}\right)}{\Gamma\left(\frac{2-y}{y}\right)}, V_t^T(y)\rangle - \frac{\mu^2}{\mu^2+\sigma^2} V_t'\right) Q_t^{-1},$$
$$V_t' = \left[\frac{m!}{(m+1)!}, m\frac{(m-1)!}{(m+1)!}, \dots, \binom{m}{i}\frac{i!\,(m-i)!}{(m+1)!}, \dots, \frac{m!}{(m+1)!}\right],$$
$$V_t(y) = \left[B_{(0,t)}(y), B_{(1,t)}(y), \dots, B_{(t,t)}(y)\right]^T. \qquad \text{Equation 3.12}$$

The computational complexity of constructing the vector "$a$" in Equation 3.12 is $O(t^3)$. It is because of the independence of $\langle \frac{\Gamma^2\left(1+\frac{1}{x}\right)}{\Gamma\left(1+\frac{2}{x}\right)}, V_t^T(x)\rangle$ to the training data and the presence of the matrix multiplication and inversion operations.

Thus, the best-approximating function, yielded by $a$ (i.e. $\sum_{i=1}^{t} a_i B_{i,t}\left(\frac{x}{x+1}\right)$) is a polynomial of degree $t$. Therefore, root finding in the approximated function has the low computational complexity of $O(t^3 \log_2(A))$ where $A$ is the maximum coefficient in the mentioned polynomial (Kobel, Rouillier, & Sagraloff, 2016).

Finally, the computational complexity of WNB can be computed as the multiplication of the complexity related to Naïve Bayes logic (i.e. $O(p.m)$), the complexity of computing $\frac{\mu^2}{\mu^2+\sigma^2}$ (i.e. $O\left(\frac{n}{p}\right)$), and the complexity of the root-finding algorithm (Kobel et al., 2016) (i.e. $O(t^3 \log_2(A))$. In other words, the WNB computational complexity is $O(t^3 \cdot \log_2(A) \cdot n \cdot m)$.

### 3.5.4 Comparison

In section 4.4, we describe the reason why, in the addressed problem/dataset, the PNB, Multinomial Bayes, and SVM deserve being the state-of-the-art competitors for our experiments. Therefore, we can here gather together the computational complexity of the proposed methods as well as the compared methods in Table 3.4.

**Table 3.4. The complexity of the proposed methods as well as their competitors (the gray terms are either bounded or data independent). 'n' stands for the number of whole the training samples, 'm' for the features number of each item, and 'p' for the number of classes. 't' and 'A' are variables related to the utilized approximation technique in WNB which are set once forever and are problem-size-independent.**

| Level | Method | Worst-case Time Complexity |
|---|---|---|
| **1.** | MB | $O(n \cdot m \times 1)$ |
| | PNB | $O(n \cdot m \times 1)$ |
| **2.** | WNPC | $O\left(n \cdot m \times \log_2\left(\frac{n}{p}\right)\right)$ |
| | WNB | $O(n \cdot m \times t^3 \cdot \log_2(\mathrm{A}))$ |
| **3.** | SVM-SMO | $O(n \cdot m \times p \cdot n^2)$ |

Based on Table 3.4, all of the classifiers are divisible by $O(n \cdot m)$ and the best computational (or time) complexity belongs to Multinomial Bayes and PNB. However, it is necessary to remind that, on the one hand, as mentioned in the previous section, revisiting Multinomial Bayes is recommended only for low $n/p$ and high $m$ circumstances, and on the other hand, PNB is proposed to provide a balance between accuracy and complexity and, thus, is never intended to provide the best accuracy (especially in comparison with WNB and WNPC which have more appropriate and flexible probability distributions).

The next low complexities belong to WNPC and WNB. In addition to $O(n \cdot m)$, they have the terms $O(\log_2(n/p))$ and $O(t^3 \log_2(A))$, respectively, which are quite small factors. For example, even assuming an ideal big-data problem in which $10^6$ training documents (e.g. tweets) are available for each author (i.e. $n/p = 10^6$), yet the value of the extra term of WNB (i.e. $\log_2(n/p) < 20$) is quite small. Alongside, in WNPC, the additional term $O(t^3 \log_2(A))$ is completely data independent; $t$ is the degree of the polynomial that is used for estimating $g(x)$. $t$ is set once forever and even low $t$ values (e.g. $t = 10$) can provide very appropriate approximations for $g(x)$. Figure 3.4 shows the error of the approximation provided by $g_1\left(\frac{y}{1-y}\right) \approx \sum_{i=0}^{10} a_i B_{(i,10)}(y)$ and $g_1(x) \approx \sum_{i=0}^{10} a_i B_{(i,10)}\left(\frac{x}{x+1}\right)$, (the left and right plots, respectively). As it can be seen, the plots represent errors less than $2 \times 10^{-3}$ for function approximation.

However, although WNB and WNPC are in a same level of complexity thanks to their bounded or data-independent additional terms (to $O(n \cdot m)$), comparing at a smaller-scale, in practice WNPC is almost always faster than WNB because (as mentioned) even in big data problems $\log_2 n/p < \log_2 10^6 < 20$, whereas letting $t$ to have the reasonable value of $t = 10$, we have $t^3 = 1000$. It is necessary to remind that this value is a permanent upper bound and as $t$ is decided based on the

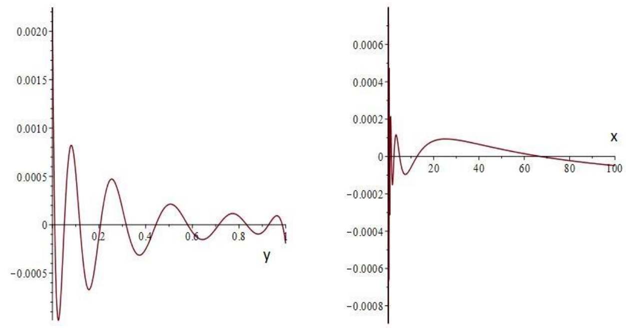



**Figure 3.4. The error of the approximating functions $\sum_{i=0}^{10} a_i B_{(i,10)}(y)$ (the left plot) and $\sum_{i=0}^{10} a_i B_{(i,10)}\left(\frac{x}{x+1}\right)$ (the right plot) from the main function $g_1\left(\frac{y}{1-y}\right) = g(x)$.**

expected accuracy level, it would be kept fixed for every problem whether it is a small scale or a big-data problem.

Thereafter, the worst complexity belongs to SVM because of its data-dependent additional term $O(p \cdot n^2)$ (to $O(n \cdot m)$). Even considering a very low-scale problem in which there exist seven authors (i.e. $p = 7$) and, for each author, there are only 2 training documents (i.e. $n = 7 \times 2$), this additional term would be as large as $p \cdot n^2 = 1372$. Therefore, almost always, the mentioned additional term is expected to be greater than the discussed upper bounds in level 2 classifiers.

### 3.5.5 Implementation and code complexity

All of the PNB, WNB, and WNPC classifiers are implemented in Python v.3.6.1. The only utilized external library is the SpaCy[33] open-source library and Scikit-learn[34]. The former provides us with the required syntactic parser for sn-gram whereas the latter provides SVM (sklearn.svm.SVC) and Multinomial Bayes (sklearn.naive_bayes.MultinomialNB) classifiers. The PNB algorithm code is accessible online via the link http://dmls.iust.ac.ir/CogLing/NaiveStylometry.zip.

However, about the code complexity, given the sn-grams, it is the same as discussed in the previous subsection. However, evidently, the complexity of constructing sn-grams shall be added to the classifier complexity. Although the complexity of SpaCy syntactic parser is not mathematically discussed in state-of-the-art, it is experimentally proven to be the best among other implemented (available) syntactic parsers (Stent, Choi, St, St, & York, 2015). We, moreover, should know about the complexity of constructing sn-grams from a parsed document. Although a recursive depth-first traversal algorithm is presented in (Sidorov et al., 2014) for construction of sn-grams from syntactic parse trees, alternatively, we can also utilize its non-recursive version for having linear

[33] https://spacy.io/

[34] http://scikit-learn.org/

complexity in, both, time ($O(t+g)$) and space ($O(t)$), when $t$ stands for the number of nodes in the extracted syntactic parse tree, and $g$ stands for the number of (grammatical) edges. Albeit, the breadth-first search could, also, be utilized for constructing sn-grams, because the order of sn-grams is not important and they are treated as being in a bag. However, accordingly, the total complexity of the proposed experiments is as low as

$$O(classifier + n \cdot s \cdot (t+g) + SpaCy\_parser) \qquad \text{Equation 4.2}$$

where $n$ stands for the number of total train documents and $s$ stands for the average number of sentences of each document and $SpaCy\_parser$ for the (low) computational complexity of SpaCy syntactic parser. Please note that, we can utilize the dependency parser proposed in (Sagae & Lavie, 2005) to make the parser complexity, linear, and therefore the total complexity as

$$\begin{aligned} &O(classifier + n \cdot s \cdot (t+g) + n \cdot s \cdot w) \\ &= O(classifier + n \cdot s \cdot (t+g+w)) \end{aligned} \qquad \text{Equation 4.3}$$

where $w$ stands for the average number of words of a sentence.

For, even more improvement in time-complexity, GPU programming can be utilized, in the presence of which, other parsers such as (M. Lewis, Lee, & Zettlemoyer, 2016) are much more appropriate. However, these improvements are left to be done in future works of this study.

Now that we have fully covered the theoretical novelties on the Text Mining field (including the Uncertainty-handling NLP platform and text classifiers required for the proposed pTER system), it would be the turn to propose the novel theories which are come up for the prediction system that is the CF-related novel techniques proposed in this section.

## 3.6 Generalized Significance Weights

In subsection 1.1.6, we discussed on different SWs, proposed on the state-of-the-art. A complete list of SWs is presented in the left column of Table 3.5. Before starting to analyze the mentioned SWs, it is worthy to point out an interesting point about them: Most of the mentioned studies are presented without any reference to the previously presented studies (or at least not all of them). It means that the proposed survey in subsection 1.1.6 is the first comprehensive one in this field.

However, in the following, we intend to propose a generalization of them toward more alleviation in the so-called cold-co-start problem. In this regard, firstly, subsection 3.6.1 addresses the logic behind the existing SWs in the state-of-the-art and provides an insight on how the idea of their generalization can result in the better performance of them. Then, section 3.6.2 proposes the intended methodology for evaluating them.

### 3.6.1 The logic behind the existing Significance Weights

This study is the first one that (as a side product) provides a survey of SWs as well. However, here we analyze the logic behind the mentioned SWs and then propose their generalized versions.

Looking at the Herlocker SW, or similarly (Jung, Park, & Lee, 2004), we see that the idea is based on the following rules

1- The weighting value lies on $[0,1]$.

**Table 3.5. Standard and generalized versions of the existing SWs in state-of-the-art.**

| | Standard version | Generalized version |
|---|---|---|
| Herlocker | $\frac{\min(\lvert u_i \cap u_j\rvert, y)}{y}$ | $\max\left(0, z+(1-z)\frac{min(\lvert u_i \cap u_j\rvert, y)}{y}\right)$ |
| Sigmoid | $\frac{1}{1+e^{-\frac{\lvert u_i \cap u_j\rvert}{2}}}$ | $\max\left(0, \frac{1}{z+0.5}\left((z-0.5)+\frac{1}{1+e^{-y\times\lvert u_i \cap u_j\rvert}}\right)\right)$ |
| Koren | $\frac{\lvert u_i \cap u_j\rvert}{\lvert u_i \cap u_j\rvert + y}$ | $\max\left(0, z+(1-z)\frac{\lvert u_i \cap u_j\rvert}{\lvert u_i \cap u_j\rvert + y}\right)$ |
| Ali | $\begin{cases} \frac{\lvert u_i \cap u_j\rvert}{y}. & SM \geq 0 \\ \frac{\lvert u_i \cap u_j\rvert}{y+\max(\lvert u_i \cap u_j\rvert . y)}. & SM < 0 \end{cases}$ | $\begin{cases} \max\left(0, z+(1-z)\frac{\lvert u_i \cap u_j\rvert}{y}\right). & SM \geq 0 \\ \max\left(0, z+(1-z)\frac{\lvert u_i \cap u_j\rvert}{y+max(\lvert u_i \cap u_j\rvert, y)}\right). & SM < 0 \end{cases}$ |
| Arctangent | $\frac{2}{\pi}\arctan(\lvert u_i \cap u_j\rvert)$ | $\max\left(0, z+(1-z)\left(\frac{2}{\pi}arctan(y\cdot\lvert u_i \cap u_j\rvert)\right)\right)$ |
| Candillier | $\frac{\lvert u_i \cap u_j\rvert}{\lvert u_i \cup u_j\rvert}$ | $\max\left(0, z+(1-z)\frac{\lvert u_i \cap u_j\rvert}{\lvert u_i \cup u_j\rvert}\right)$ |
| Zheng | $\frac{2\lvert u_i \cap u_j\rvert}{\lvert u_i\rvert + \lvert u_j\rvert}$ | $\max\left(0, z+(1-z)\frac{2\lvert u_i \cap u_j\rvert}{\lvert u_i\rvert + \lvert u_j\rvert}\right)$ |
| Sun | $\frac{\lvert u_i \cap u_j\rvert}{\sqrt{\lvert u_i\rvert \times \lvert u_j\rvert}}$ | $\max\left(0, z+(1-z)\frac{\lvert u_i \cap u_j\rvert}{\sqrt{\lvert u_i\rvert \times \lvert u_j\rvert}}\right)$ |

2- It should assign a small weighting value when two users have few co-rated items and greater or equal values for more co-rated items; therefore, it should be a monotonically increasing function.

The sigmoid, the Koren, and the second formula in the Ali GSW have moreover a stronger criterion than 2. They suggest a strictly increasing function instead. Looking at their SWs, it is seen that they obey two extra criteria

3- The function should have a horizontal asymptote

4- It should go toward its asymptote with a high initial speed and a negative acceleration (by speed, we mean the first derivative of the function, and by acceleration, its second derivative).

The reason of the above rules is: after reaching some number of co-rated items, we are sufficiently confident about the similarity result; so, incrementing the number of co-rated items would be less effective after the cold-start conditions. The first formula of Ali, however, has none of the above criteria and has sufficed to be ascending. While introducing Ali, we mentioned the corresponding reason that is related to more competency of positive similarities than negative ones. However, in the multivariate category, the SWs have the property 1 (being bounded in $[0,1]$), as well as being ascending by an increment of $|u_i \cap u_j|$ (as the common ground of the properties 2 and 4), plus the following property:

5- Their speed should be controlled by a factor that is related to the number of rated items by each of the two considered users.

The mentioned factor in Candillier SW is $\frac{1}{|u_i \cup u_j|}$, in Zheng SW is $\frac{2}{|u_i|+|u_j|}$, and in Sun SW is $\frac{1}{\sqrt{|u_i| \times |u_j|}}$. The logic behind this normalizing factor is: assuming a high number of common ratings between two users, this high value results in the high significance of similarity measure only if it is high in comparison with the total ratings that each of them have. As an example, if two users have commonly rated 50 movies and each of them have watched/rated only those movies the measured similarity (e.g. by Pearson correlation) will be super meaningful, but if each of them have watched/rated 500 movies whereas only 50 of them have been the same, then the similarity measure would be less meaningful, as their taste seem to be meaningfully different.

As mentioned above, the left column of Table 3.5 lists the surveyed SWs. Obviously, there can be more and more functions satisfying the mentioned properties of SWs. For example, the arctangent function which behaves similar to the univariate SWs, and also, can be generalized in the similar mentioned manner. You can see its standard and generalized version in the same Table. Figure 3.5 provides an intuition about the presented generalizations.

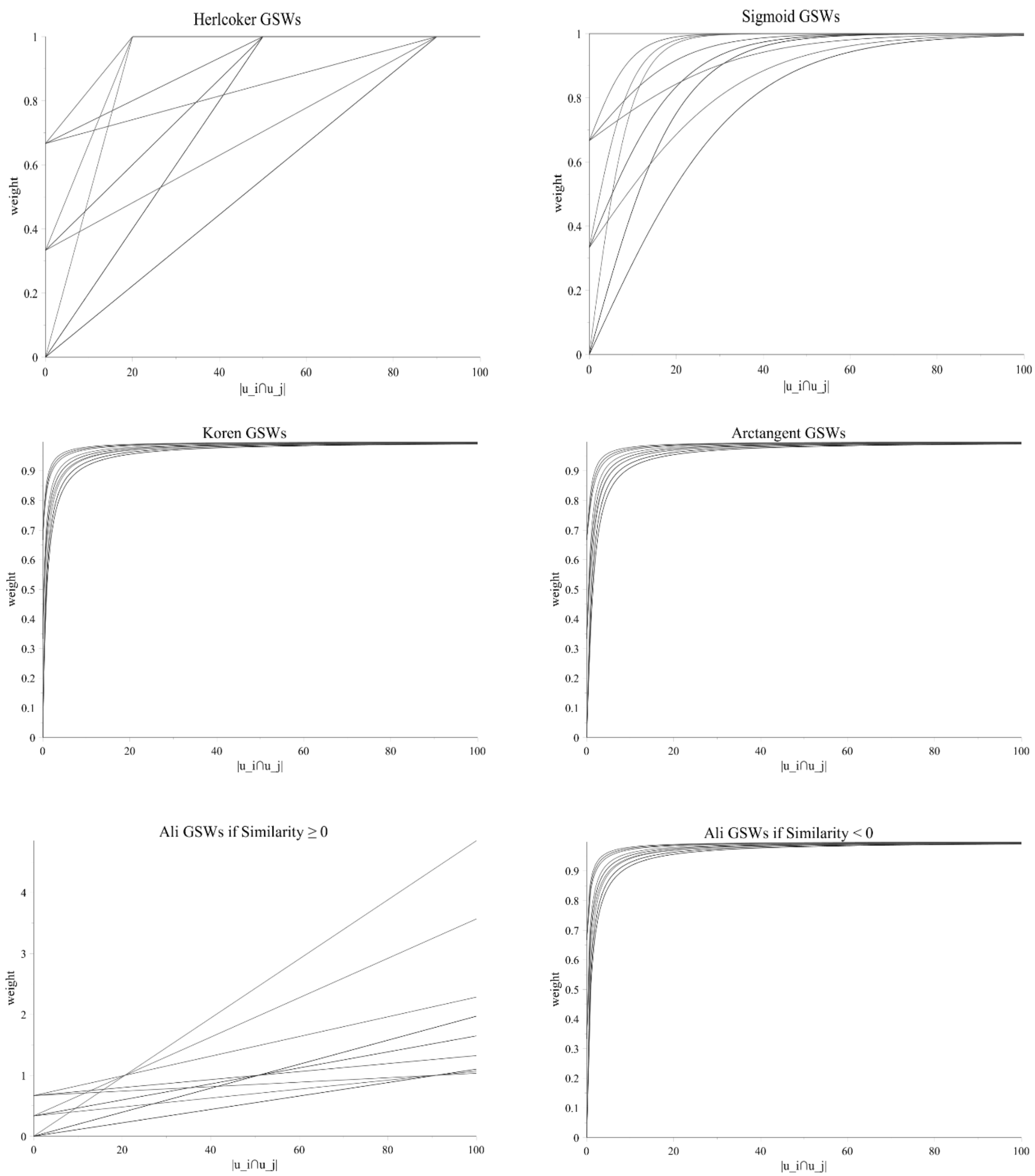



**Figure 3.5. Some generalized versions of different univariate SWs and their flexible/various shapes/instances.**

Note that, in this research, we only refer to the "cold-start user" (Park et al., 2006) (or simply cold-user (Nguyen, Denos, & Berrut, 2007): users with few ratings); because this situation results in few co-rated items in user-pairs. More specifically, we only consider the cold-start situation in which nevertheless of users' ratings (few or many), the number of co-rated items in user-pairs are few. We call this condition as cold-start-co-user, or simply "cold-co-user".[35]

There are differences between the above-enumerated SWs. For example, the weight value in Sigmoid SW begins from 0.5 ($Sig(0) = 0.5$), whereas in the others the y-intersect is 0. In other words, Sigmoid SW assigns 62.2% ($sigmoid(1)$) of significance (which is a really high value) for two users with only 1 co-rated item, whereas the other SWs assign a very low degree of Significance (i.e. given $y = 50$, $SW(1) = \frac{1}{50}$ of significance by Herlocker and Ali#1, $\frac{1}{51}$ by Koren, and $\frac{1}{100}$ by Ali#2). This inspires a more basic question that, in general, what should the y-intersect of a SW be? Sigmoid SW chooses 0.5 whereas the others choose 0. But a suggestion for solving this issue can be generalizing the SW to have flexible y-intersect. Moreover, some of the proposed SWs have a stretched plot along x-axis whereas some others have a compressed form.

In this subsection, we propose a generalization for the enumerated SWs, while keeping satisfaction of their mentioned rules. You can see the generalized forms in the right column of Table 3.5.

The parameter $z$ changes the y-intercept of the function while the parameter $y$ is for stretching or compressing the plot along the x-axis. Note that, the parameter y is not added to Herlocker, Koren, Ali, and multivariate SWs, because they already have their own scaling factor, as described before (albeit the scaling factors of univariate SWs are logically different from scaling factors in multivariate category).

About the parameter $z$, we should also add the following point: as mentioned in subsection 1.1.6, there exist a SW (M. Wang & Ma, 2016) in which the first positive (nonzero) significance value is assigned to $x = 2$ (by $x$, we mean $|u_i \cap u_j|$). In other words, that SW has a virtual negative y-intercept. We call it virtual negative, because if the proposed linear SW was continued for $x < 2$, it intersected the y-axis in a negative point. However, formulating their SW we have

$$f(x) = \begin{cases} \max\left(0, \dfrac{x}{y}\right), & 0 < x < y \\ 1, & x \geq y \end{cases} \qquad \text{Equation 3.13}$$

Correspondingly, to cover such possible cases for the $z$ parameter, we add the maximization operator to all the GSWs, as it can be seen in the Table 3.5 (right column).

Nevertheless, we expect that for every problem (e.g. every type of dataset), for every GSW there, exists a deal of $(y, z)$ pair for which the GSW work better than its standard versions. Indeed, a main goal of this research is proposing more effective pairs and therefore more alleviation to cold-co-user (and similarly cold-co-item) problem.

---

[35] In fact, based on applying SWs for refining similarity of "users" or "items" in CF, we are only considering cold-co-user or (similarly defined) cold-co-item conditions. Because the main duty of SWs is "considering the number of co-rated items or co-rating users" and not, merely, considering the cold-user or cold-item.

As mentioned, all of the proposed functions not only obey the enumerated rules/ objectives but also have the potency of providing more flexible and therefore more effective SWs (Figure 3.5) than their standard versions.

Now, the question is how to find an effective $(y, z)$ pair? For a given problem or dataset, we define a cost function for each $(y, z)$ pair. The cost function is the mean (absolute or squared) error of the prediction experiments on the whole dataset. Then, we seek the optimal $(y, z)$ pair values by a fuzzy optimization algorithm (which will be presented in chapter 4). Then, the final suggestion would be the yielding optimal $(y, z)$ pair which in average, outputs the least error / cost. This idea would be explained in the following subsection, with more details.

### 3.6.2 Evaluation methodology

As mentioned in subsection 1.1.6, the main mission of CF is predicting a rating value for one of the non-rated items of one user. As explained in the Equation (the 3rd strategy), the prediction is the aggregation of the rating value of the $|U|$ nearest neighbors to the user $u$.

$$\hat{r}_{u,i} = \bar{r}_u + k \sum_{u' \in U} simil(u, u') \times \left(r_{u',i} - \bar{r}_{u'}\right)$$

In the previous subsection, we explained the functionality of the well-known SWs

$$simil_{weighted}^{y,z}(u, u') = w^{y,z}(u, u') \times simil_{normal}(u, u') \qquad \text{Equation 3.14}$$

Therefore, for significance-weighted CF, we can update the recommendation formula as

$$\hat{r}_{u,i} = \bar{r}_u + k' \sum_{u' \in U} simil_{weighted}^{y,z}(u, u') \times \left(r_{u',i} - \bar{r}_{u'}\right), \qquad \text{Equation 3.15}$$
$$k' = \frac{1}{\sum_{u' \in U} \left| simil_{weighted}^{y,z}(u, u') \right|}$$

In the experiments, we apply the GSWs, mentioned in the right column of 3.5, on the Equation 3.15 to propose the best $(y, z)$ parameter pairs by which Equation 3.15 provides the most accurate prediction. For experimenting, we choose a dataset in which we expect most of the user-pairs ($u$ and $u'$ in Equation 3.15) to be in cold-co-start condition, so that we ensure the priority of the proposed GSWs in this condition.

Nevertheless, for the evaluation, we apply the prediction algorithm, denoted in Equation 3.15, on a standard dataset[36]. Then, the $(y, z)$ by which the least error is resulted would be proposed as the optimized value. Equation 3.16 provides a schematic view from the process of optimizing $(y, z)$ parameters, in this research

[36] In section 4.5 (the corresponding experiments), we utilize the standard

$$\min_{(y,z)}\{Cost_{(y,z)}\}, y \in R^{+}, z \in R$$

$$Cost_{(y,z)} = \text{Total Error}\left(\bigcup_{i \in \{\text{Error Measures}\}} \{Error_{\text{i}}^{(\text{y,z})}\}\right) \qquad \text{Equation 3.16}$$

Here, we provide a short definition of the concepts, used in the above equation.

**Error measures**

By $Error$, we mean the difference of "our prediction" value for test set records[37] from the "real value," rated by the user himself. In the dataset, 80% of the data[38] is proposed as the test set and the remained 20% as the training set (we keep only 20% of data for training to welcome cold-co-user problem which we are going to alleviate). The two following "Error Measurements" are used to measure the accuracy / error of the prediction experiments: The well-known Mean Absolute Error (MAE) (Hyndman & Koehler, 2006)

$$MAE = \frac{1}{|TS|}\sum_{i=1}^{|TS|}\left|r_{u,i} - \hat{r}_{u,i}\right| \qquad \text{Equation 3.17}$$

and Root Mean Square Error (RMSE) (Chai & Draxler, 2014)

$$RMSE = \sqrt{\frac{1}{|TS|}\sum_{i=1}^{|TS|}\left(r_{u,i} - \hat{r}_{u,i}\right)^2} \qquad \text{Equation 3.18}$$

where $TS$ represents the test set, $r_{u,i}$ (refer to Equation 1.1) is the recommended rating by algorithm and $\hat{r}_{u,i}$ is the real rating of the user in the test set. Therefore, we can substitute $\{\text{Error Measures}\}$ in Equation 3.16, with {MAE, RMSE}.

**Total error**

From the one hand, as the final optimization output, we should propose one $(y, z)$ pair for each GSW, and from the other hand, the best parameter pair for minimizing Mean Absolute Error and Root Mean Square Error, most probably, contradict each other. Considering that we are going to propose a GSW which provides both appropriate Mean Absolute Error and Root Mean Square Error, we should compute a resultant total value for these two.

An important point about the utilized error measures is: both of the Mean Absolute Error and Root Mean Square Error are estimators of standard deviation. Although in this application, they do not represent a real standard deviation, the fact that they are of the same type, is enough reason for not

[37] By a record, we mean the triple of (1) a user, (2) an item, and (3) the rating of that user to that item.

[38] Among the test set queries, a proportion of them are not utilized because, while predicting user's rating of the query by $n$ neighbors, the targeted user should have $n$ neighbors who have assigned a rating to that movie; otherwise, that record is excluded from the experiments.

normalizing them before unification. Thus, very straightforwardly, we can set the Total Error as their Euclidean norm.

$$\text{Total Error}(\{Err_1, Err_2\}) = \sqrt{Err_1^2 + Err_2^2}. \qquad \text{Equation 3.19}$$

**Neighbors**

We repeat the optimization for different neighbor numbers ($|U|$ in the Equation 3.15) to see the effect of neighbors increment on the efficiency of SWs / GSWs. We repeat the experiments for neighbor numbers $\{1,2,\dots,N\}$. However, once $N$ is determined, for having a fair comparison among them, all the test cases in the experiments must be performable by all the $N$ experiments. Thus, all the test cases in which the corresponding item (whose rating is subject of estimation) has less than $N$ raters are filtered. This makes the test set to only contain items with at least $N$ ratings in the experiments. But, those filtered / removed less-often rated items are very welcome for the experiments, because while rating such less-rated items, the number of possible neighbors of the active user is reduced (to only whom have rated that item).

Thus, we set $N$ to be 10 for being few enough for not losing the less-often rated items, and correspondingly, to welcome cold-co-user problem, which we are going to alleviate.

## 3.7 A linear-complexity Fuzzy Optimization Algorithm

As mentioned in section 1.1.6, considering the possibility of real-time requirement to finding the optimal parameters of the GSW, because as mentioned in section 1.1, there is no Metaheuristic Optimization Algorithm in the state-of-the-art which (1) does not need numerous iterations, (2) does not fall into local optimums, and (3) provides linear computational complexity, in this section, we propose a novel linear-complexity fuzzy optimization technique.

In general, an optimization algorithm should be adopted based on the time complexity of evaluating/calling the corresponding cost function and the scale of the corresponding domain. If the domain consists of finite and not super large points (compared to the available hardware computational capacity) and the corresponding time complexity is linear, then even an exhaustive search algorithm can be fast- and accurate-enough for finding the optimum point. However, when the time complexity increases or the domain is infinite or consists of a finitely super large number of points, an exhaustive idea cannot be adopted anymore, because it would be super time consuming or infeasible. Thus, optimization algorithms were necessary to appear.

In a very general view, numerical optimization algorithms can be categorized into two categories. The first category is related to classical algorithms in which the cost function and the domain constraints are considered to have some known characteristics. Some examples for this category are: Linear Programming (Dantzig, 2016) for linear cost functions with polytope constrained space, Convex Optimization (Boyd & Vandenberghe, 2010) algorithms for concave / convex functions (while maximization / minimization) with convex constraint set, Quadratic Programming (Dostál, 2009) for quadratic cost function with linear constraints, Fractional Programming (Stancu-Minasian, 2012) for the cases in which the cost function is a ratio of concave/convex functions and the constraints are convex, Combinatorial optimization (Ding-Zhu & Pardalos, 2013) for the cases in which the domain includes finite points, especially when the exhaustive search is not feasible.

The second category of the numerical optimization algorithms is related to heuristic and MOA (Boussaïd et al., 2013) in which (as the name implies) the goal is not reaching a proven optimum point/solution, but it is approaching a solution that is expected to be near enough to the optimum point. This category, itself, is divided into two main subcategories (Boussaïd et al., 2013). The first subcategory includes single-solution-based MOAs which are the MOAs that start from a single initial solution and move away from it toward the optimum point. Some examples of this subcategory are Simulated Annealing (Aarts, Korst, & Michiels, 2014), Tabu search (Glover & Laguna, 2013), GRASP (greedy randomized adaptive search procedure) (Resende & Ribeiro, 2010), variable neighborhood search (Hansen & Nenad Mladenović, 2014), guided local search (Voudouris, Tsang, & Alsheddy, 2010), iterated local search (Lourenço, O., & T., 2010). The second subcategory of MOAs includes population-based MOAs which deal with a group or population of solutions (rather than dealing with a single solution) among which the best one is adopted. The most well-known methods of this subcategory are related to Evolutionary Computation (Jun Zhang et al., 2011) (e.g. genetic algorithms, evolution strategies, evolutionary programming, genetic programming) and Swarm Intelligence (Merkle & Middendorf, 2014) (e.g. Ant colony, Particle Swarm Optimization, Bacterial Foraging Optimization, Bee colony).

However, the optimization problem, in the previous section, has some constraints which restrict the choice of optimization methods. On the one hand, adopting the local optimums is quite unwelcome, as the goal of this section is "freeing up" the potential of SWs by their generalization. On the other hand, we seek for an optimization method which can also cover the real-time applications of Memory-based CF. We are interested in real-time applications because we are proposing dataset-adaptive GSWs. Definitely, no Social Network or rating platform has a uniform treatment. Therefore, the claimed dataset-adaptive GSWs must be (easily) updatable. More in detail, in a recommender system, after a real user assigns a new rating to an item, the system can update the previous version of the Total Error values. Then, if the optimizer can be re-trained in real-time, it can immediately update the best $(y, z)$ pair by which the weighted similarities become updated as well. The potency of being trained in real-time, restricts us to choose low time-complexity optimizers, and if the addressed system deals with big data, we would be restricted to choose linear time-complexity optimizers. This latter constraint should be considered beside the fact that the time-complexity of simulations, required for Memory-based CF error evaluation is very high.

Therefore, the addressed problem has the following 3 restrictions:

1. The cost function of the addressed optimization problem (Memory-based CF error in Equation 3.16) does not have any analytic formula and requires rather time-consuming simulations to be evaluated.
2. The global minimum value for the cost function is required.
3. The optimization algorithm has to have linear time-complexity.

Considering the restriction one, there is no formula with known characteristics –which is expected in classical optimization. Thus, among the two main categories (classical optimization algorithms vs. MOAs), the classical methods are not expected as being appropriate for this optimization problem. Moreover, considering the same restriction (number 1), the high-iteration optimization methods, whether in single-solution-based optimizations (e.g. simulated annealing, iterated local search) or in population-based optimizations (e.g. Evolutionary Computing, Swarm Intelligence) should be avoided, because they would be impractically time-consuming.

Considering the restriction two, the approaches, which are mainly in single-solution-based MOAs (such as Tabu search, GRASP, or Hill Climbing) that potentially fall in local optimums should be avoided. Finally, by considering the last (third) restriction, unfortunately, the list of the selectable optimizers becomes clear, and (as mentioned before) there remains no optimizing algorithm which fits all the 3 abovementioned restrictions, at the same time.

Thus, we have to develop such "linear time-complexity" "low-population" "global MOA" before utilization. We propose an SM technique satisfying the mentioned restrictions. SM is a sub-category of population-based MOAs[39], which approximates models (metamodels or emulators) that are aimed to be as similar as possible to the real (expensive) cost function while having a cheap computational cost and being trainable/constructible by few evaluations of the real (expensive) cost function (Bartz-Beielstein, 2016).

However, as mentioned, similar to their other counterparts, SMs have also the drawback of not being appropriate for real-time applications which require linear time-complexity. In other words, despite their cheap time-complexity, SMs are not such cheap for being trained. Although there is a wide variety of approaches to SM (as it can be hierarchically seen in Figure 1.2 in (Tenne & Goh, 2010)), the time-complexity of the training phase of all of them is higher than linear ($O(n)$), and therefore, in this section, we propose a SM with linear time complexity in training phase.

The idea of the proposed algorithm is inspired by the study of (Chakraborty, Guha, & Dutta, 2016) that deals with the optimization problems in which evaluating/calling the cost function has very high time complexity. For example, optimizing the net profit of an investment inter-industry company, by tuning the investment proportion in each industry. Each cost-function-call in this optimization problem equals with testing the net profit of the company when $x_1\%$ of the fund is assigned for the 1st industry, $x_2\%$ for the 2nd, and … $x_n\%$ is assigned for the nth industry. Evidently, evaluating such cost function takes months or years and calling it, even once, is not feasible. The authors of (Chakraborty et al., 2016) (as a part of their study) utilize the idea of using the experts' knowledge in the format of some fuzzy rules such as

*If '$x_1$' is 'high' and '$x_2$' is 'moderate' and … '$x_n$' is 'low'*
*then 'NetProfit' is 'almost high'*

where 'high', 'moderate', 'low', and … are fuzzy variables. Then, utilizing a Tsukamoto inference scheme (Ross, 2010), the experts' fuzzy rules are converted into one function, as the estimated

[39] As mentioned in last part of introduction section (footnote 3), not all the scholars explicitly enumerate SMs under the umbrella term, population-based optimizations. The reason is explained there.

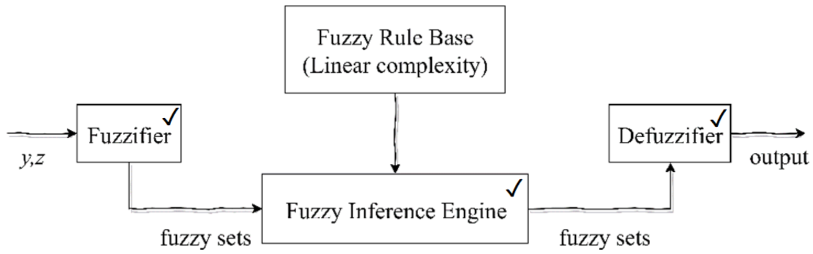



**Figure 3.7. Four components of a fuzzy inference system.**

cost function (i.e. SM). The very important property of the mentioned approach (Chakraborty et al., 2016) to SMs is: In addition to (the expected) cheapness of the SM computational cost, it has cheap time complexity in training phase (on which we discuss after explaining the subsequently inspired idea).

The mentioned idea inspired us if we can achieve some automatically driven rules for the Total Error in Equation 3.16 (similar to the rules that market-experts propose for the cost function, considered in (Chakraborty et al., 2016)). Then, we likewise can come up with such cheaply trainable SM, which would be resolving for the mentioned real-time problem.

However, automatic generation of such rules is not challenging, at all. A very simple technique that we adopt for this purpose is evaluating the cost function (Total Error) on a moderately-wide (e.g. 10×10) grid on the feasible space of the parameters $(y, z)$, like what in the exemplary grid depicted in Figure 3.6.

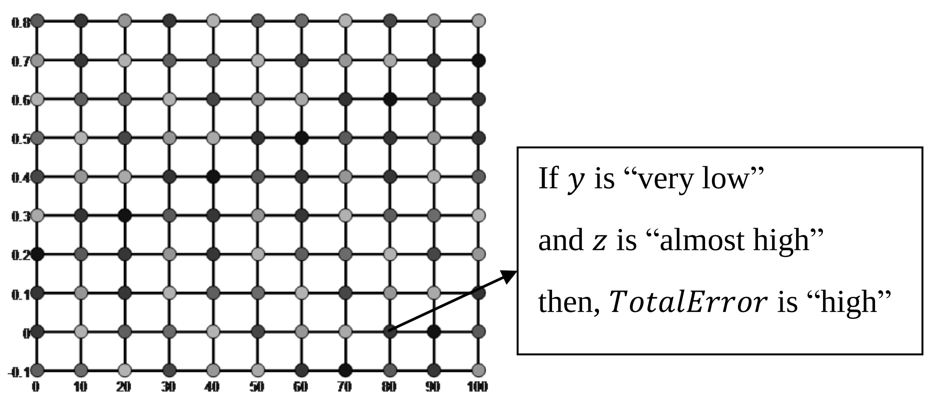



**Figure 3.6. An exemplary grid on the search space of $(y, z)$. Dark (bright) areas represent high (low) Total Error for the corresponding $(y, z)$ variable.**

In the below, we will discuss how fuzzy input and fuzzy output variables are defined, so that the fuzzy SM can linearly be trained.

In general, a fuzzy inference scheme is comprised of 4 components, 3 of which not requiring training phase (marked by ✓ in Figure 3.7). The only component which requires training is the Fuzzy Rule Base because the rules of a fuzzy inference scheme must have the following 3 properties: completeness, consistency, and continuity (L.-X. Wang, 1997). This means that after defining the initial rules, they should be checked to have the mentioned 3 properties, and correspondingly be updated (what we call training) until satisfying all the 3 conditions. However, as we explain below, we coordinate the mentioned rules, fuzzy input variables (for $(y, z)$), and fuzzy output variables (for Total Error) so that the fuzzy Rule Base has the 3 necessary properties from the very beginning.

The components specified by ✓ mark does not require training for real-time optimization experiments. The time complexity of constructing the Fuzzy Rule Base (as the only component which plays role in training) is linear.

We choose the grid points of each GSW so that it's corresponding GSWs have a y-intersects among $\{-0.1, 0, 0.1, 0.2, \dots, 0.8\}$. It is because, logically, we believe that a y-intersect more than 0.8 is meaningless, as the similarity value of two users with only one common rating does not desire to be very near to 1. For example, consider two users whose common watched/rated movies is only one. Therefore, the similarity value of them is computed based on the score by which each of them has rated that single movie. Such similarity is not trustable enough, because it is very likely that, that movie has special characteristic (e.g. being very popular, being artfully comedy …) considering which most of the people (notwithstanding their similarity) rate it very positive. Reciprocally, those users may have rated it reversely but it might be because that movie has a bipolar nature and two users, despite their similarity, may belong to different poles (e.g. fans of Barcelona / Real-Madrid) of that bipolar. Thus, assigning a very high significance to them is unfair.

On the other hand, we believe that, logically, a y-intersect less than -0.1 is meaningless, because the similarity value of two users with two or three common ratings should have, at least, a low degree of significance and does not desire to be 0 (-0.1 itself is related to $n = 0$ that never happens because two under-comparison users have always at least one common rating). For example, if two users have commonly rated two or three movies and the rated movies are very similar or very different, although on one side of the coin, there exist the probability that all of the rated movies are special, the other side of the coin should not be neglected as well. As the inferred similarity or dissimilarity, at least exists for an aspect of their character (at least a minor aspect) and it should be considered by (at least) a little positive weight value.

In other words, we believe that {-0.1, 0, 0.1, 0.2, …, 0.8} represents the list of {super-low, very low, low, almost low, …, super high} y-intersects by which the variable $z$ can be calculated.

About the stretch variable $y$, we assign it so that the GSW function arrives in y-value 0.99 (or 1 for Herlocker's GSW) exactly when $x$ ($|u_i \cap u_j|$) is equal with each of the values {10, 20, …, 100} in ten different rules. By this technique, we are tuning the maximum value for GSW functions (we assume 0.99 as the maximum GSW value, in the asymptotic functions). Logically, the maximum value of a GSW represents "infinitely significant" degree. We believe that, even in the optimistic view, the similarity value of two users whose co-rated items are less than 10 cannot be considered as infinitely significant. Reciprocally, even in the pessimistic view, we believe that when the number of co-rated items of two users is more than 100 (e.g. two users who have similarly or differently rated 100 movies) the similarity of them should be considered as fully-meaningful. In other words, we believe that {10, 20, …, 100} represents the list of {super-low, very low, low, almost low, …, super high} values for $x$ value on which the GSW becomes maximum (or 0.99), and for the corresponding $y$ variable as well.

These two tricks for adopting the list of training $(y, z)$ pairs is adopted because a regularly gridded $(y, z)$ plane may not be effective enough, as small (large) changes in $(y, z)$ may result in large (small) changes in GSW functions, contrary to the regular grid expectation. Thus, we set $(y, z)$s in the described way to uniformly cover the feasible area of GSWs (Figure 3.8 shows the corresponding GSWs for Herlocker and Sigmoid).

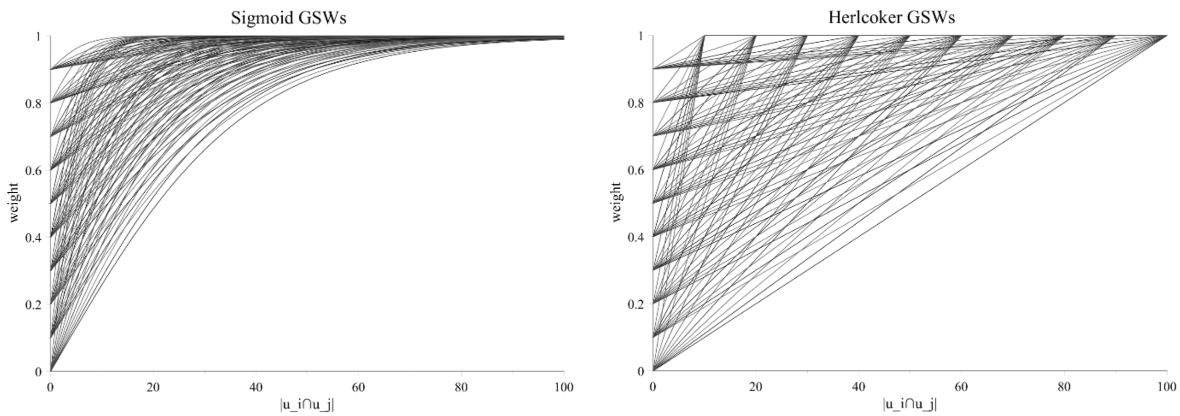



**Figure 3.8. Sigmoid and Herlocker GSWs with y-intercepts in {0, 0.1, …, 0.9}, landing x value in {10, 20, …, 100}.**

Then, using a fuzzifier, we transform each point of the mentioned grid to a fuzzy rule. For this purpose, we use triangular fuzzifier, examples of which can be seen in Figure 3.9 in which each membership function represents one linguistic variable.

Please note that although the minimum and maximum points of the above-mentioned ranges are important, the number of partitions depends on the computing power of the machine that the target real-time optimizer is being implemented on. Figure 3.9 depicts exemplary variables for the case that the number of partitions is 8.

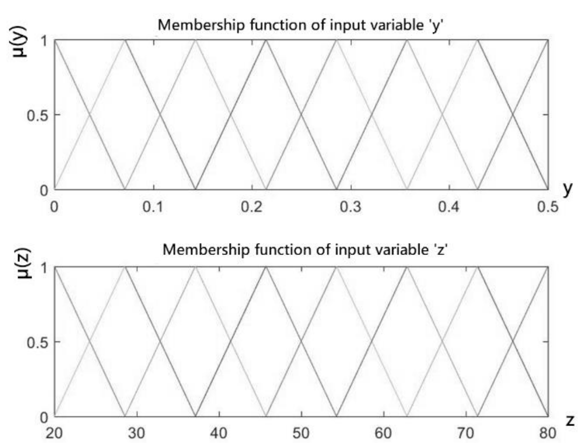


**Figure 3.9. Example triangular membership functions utilized for fuzzifying $(y, z)$ values. The triangles (as explained) are not, necessarily of the same size.**

As it can be seen in the depicted examples, each triangular membership function has the abovementioned specified points (after calculating the corresponding $y$ and $z$) as its maximum, and its two neighbors as its left and right minimums.

The fuzzy variables for $(y, z)$ are described above. However, for the variables representing Total Error, we utilize the following straightforward strategy.

We set the domain of the Total Error to the interval specified by the minimum and maximum value among the evaluated Total Errors (of the abovementioned $(y, z)$ points in Figure 3.6). Then, we average jumps of Total Errors while moving from all of the nodes (in Figure 3.6) to their right, above, and right-above nodes (if any) and the calculated average-jump is set as $\beta$ in Figure 3.10. Then we extend, a bit, the domain of Total Error (equally to the right and left) so that it can be completely partitioned by $\beta$-length steps. The triangular membership functions of Total Error are different from the $(y, z)$ memberships as the beginning and ending points of all of them are the same and they only have different max points. We explain the reason of this difference while describing why the fuzzy Rule Base of the proposed system has the necessary properties.

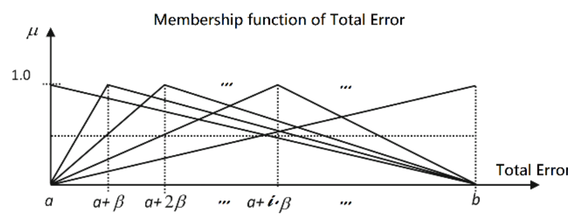



**Figure 3.10. Example membership functions for fuzzy variables of Total Error. The distances of max nodes are equal.**

Although we do not name the corresponding linguistic variables of each membership function, each of them logically represents one linguistic variable. For example, we could name the 8 membership functions of the variable $y$ as super low, very low, low, almost low, almost high, high, very high, super high. We could also suppose $n$ linguistic variables for the $n$ membership functions in Figure 3.10.

Using such hypothetical names, the bold point in Figure 3.6 can be fuzzifier to the following linguistic rule

If '$y$' is 'super low' and '$z$' is 'very high' then 'Total Error' is 'moderately very high'

However, such names are quite useless and we presented the above example just for clarifying the idea. In practice, we utilize rules of the following type

If '$y$' is $Y_1$ and '$z$' is '$Z_7$' then 'Total Error' is '$T_{19}$'

Now, we briefly describe why the constructed fuzzy Rule Base (comprised of the abovementioned rules) is complete, consistent, and continuous. Note that we have 100 rules (fewer or more rules are also possible when by varying the partitions number as described above), one for each of the

$10 \times 10$ $y$ and $z$ fuzzy variables, in each of which one fuzzy variable $y$ and one fuzzy variable $z$ is fully fired (has the membership degree of 1). This implies that the proposed fuzzy Rule Base is, from the one hand, complete, and from the other hand, consistent. Completeness is clear because, for each $(y, z)$ fuzzy variable pair, one rule is defined. Consistency is also established because for each rule one and only one $(y, z)$ fuzzy variable pair is fired and therefore each $(y, z)$ fuzzy variable pair is fired once and the existence of two rules with the same antecedent is impossible.

Continuity is also established because of the form of triangular membership functions of $\text{Total Error}$ (Figure 3.10). They are designed so that while having a justifiable distance between (the max value of) two neighbor fuzzy variables ($\beta$), all of the fuzzy variables have an intersection with each other which guarantees continuity without the irrational arrangement of fuzzy variables.

After designing a standard fuzzy Rule Base, 3 components (among the 4) of the fuzzy inference system depicted in Figure 3.7 (input fuzzy variables, output fuzzy variables, and Fuzzy Rule Base) are ready. The only remained component of the proposed fuzzy inference scheme is fuzzy inference engine. We use Mamdani inference engine as it is compatible with the mentioned fuzzy Rule Base, which is a smart but simple inference engine that requires linear time complexity for inference (Ross, 2010).

The fuzzy inference system depicted in Figure 3.7 is an SM that not only is linearly evaluable but also is linearly constructible. The only time-consuming (with linear complexity) component of constructing the fuzzy inference system is its fuzzy Rule base and the only time consuming (with linear complexity) step of constructing the fuzzy Rule base is computing the average jump (required for adjusting $\text{Total Error}$ fuzzy variables) that can be done by linear time complexity. Please be reminded that the input $(y, z)$ fuzzy variables in non-real-time applications are computed once. But in real-time applications (as explained above) as every user rated a previously-unrated item, the whole Total Error values are linearly updated and the proposed optimizer can linearly optimize the $(y, z)$s by considering this additional data.

It is also notable that the proposed SM is not the first fuzzy inference system utilized as SM. For example, (Akbarzadeh-T, Davarynejad, & Pariz, 2008) propose a fuzzy SM for being utilized before the evolutionary optimization. But, the main characteristic of the proposed fuzzy SM is its linear time-complexity for being trained, the characteristic which its state-of-the-art (fuzzy and non-fuzzy) counterpart optimizers are deprived of.

After having a trained fuzzy SM, we utilize Exhaustive Search algorithm on it for approximating the best $(y, z)$ (as the estimated global optimum). In section 4.5, we apply the proposed algorithm to the chosen Dataset (Movielens 100k) and provide discussions and analyses on the results.

The next chapter is dedicated to the Experiments and Results, engaged with the theories developed in this chapter.

# 4 Experiments and Results

In the previous chapter, we proposed our novel theories in each of the required fields of science to the proposed pTER system. In this chapter, we provide the evaluating experiments related to each of them, one by one.

We start by the WordNet Fuzzifier algorithm. However, considering that the logic of the algorithm is theoretically proven, we evaluate the efficiency of the output of the algorithm. For this purpose, in section 4.1, we apply BoS, BoWS, BoFS, and BoFWS Text Mining models on the standard English WordNet and fuzzified English WordNet (subsection 3.1.3) and compare the results on the Sentiment Analysis problem for proving the superiority of the fuzzified English WordNet over the standard English WordNet. Then, in section 4.2 we deal with creating and proving the efficiency of the Interval Fuzzy Synsets (presented in section 3.2). The evaluation is done by comparison of the interval membership degrees, produced with the algorithm vs. the version produced by native English speakers.

After that, sections 4.3 addresses the experiments related to Poisson Naïve Bayes classifier and section 4.4 addresses similar experiments to section 4.3 but by means of the two Weibull Naïve Bayes classifiers proposed in section 3.4. At the end, section 4.5 addresses the experiments and evaluations related to GSWs and their optimization by the linear-complexity optimizer, proposed in section 3.7.

## 4.1 Fuzzified English WordNet and Bag of Fuzzy Synsets

In this section, we are going to present the evaluation of the newly produced FWN. We would like to emphasize that our algorithm has been theoretically proven. This is the reason why we call the validation addressed in this section as validation of FWN and not a validation of the proposed algorithm.

Although FWN is produced by the same proven algorithm, we validate it to check the fulfillment of the assumed conditions, described in section 3.1: the sufficient size of the corpus and the precision of the WSD algorithm. To this end, we apply the BoFS and BoFWS models to Sentiment Analysis, as a very well-known sub-field of Text Mining, in order to see whether it produces an improvement in accuracy over the BoS and BoWS model. This would prove the usefulness of the extra information provided by FWN.

**What problem to solve:** We have to choose a Text Mining problem/sub-scope and an appropriate algorithm of that field to be solved once by BoFS / BoFWS and once by BoS / BoWS. As said, our published FWN, as well as the potential outputs of the algorithm for other languages, can be employed not only in Text Mining problems but in general, in all the scientific disciplines which make use of WordNet or other WLDs. Nevertheless, in order to validate it, we will choose an appropriate algorithm from a Text Mining sub-scope (Sentiment Analysis due to the scope of this dissertation) and see the difference in performance when applied to BoS / BoWS models compared to BoFS / BoFWS.

**Sentiment Analysis Algorithm.** For performing the experiments, the most important prerequisite is to choose the algorithm by which Sentiment Analysis is done. There are hundreds of proposed

Sentiment Analysis algorithms in state-of-the-art; some of them specific for Sentiment Analysis purposes (such as (Reforgiato Recupero et al., 2015) and many others) and some others general classifiers which can also be adopted for Sentiment Analysis.

One of the proposed text classifiers (which at the same time is simple, low-complexity, and accurate) is the Support Vector Machine with Naïve Bayes features, the so-called NBSVM, proposed by (S. Wang & Manning, 2012). In (S. Wang & Manning, 2012), they utilize unigrams and bigrams to feed NBSVM. In our experiments, also we adopt NBSVM as the classifier and instead of unigrams and bigrams utilize BoS and BoFS for the mentioned comparison purpose.

**Dataset.** In the experiments, we use the IMDB dataset, proposed in 2011 by (A. Maas, Daly, Pham, & Huang, 2011) as a sentiment-tagged (positive/negative) large dataset of informal movie reviews from the Internet Movie Database. It is comprised of 25k positive and 25k negative reviews and has been utilized by hundreds of novel research studies (A. L. Maas et al., 2011). It is one of the most well-known datasets in *document-level Sentiment Analysis*.

**Experiments and Results.** Before addressing the experiments, it is worthy to establish the evaluation metric by which we shall qualify the results. Considering that, in document-level Sentiment Analysis, no retrieval is done and, correspondingly, the main task consists in classifying the documents to one of the possible sentiments, the only common meaningful measure for evaluation will be accuracy that is $\#correct\_classifications/\#items$.

However, we feed NBSVM once by BoS / BoWS and once by BoFS / BoFWS and compare the accuracies. Please note that we have two versions of FWN (v.83 and v.93) and, correspondingly, feed BoFS and BoFWS, in separated experiments, by each of the proposed two versions. Figure 4.1 represents the accuracy of all the 6 tested Sentiment Analysis experiments. 80% of the data is assigned for training and the reaminder-20% for testing. For the purpose of reaching more confidence on the meaningfulness of the results, 10-fold cross validation strategy is also adopted. We measure the p-values for the null hypothesis that the results, corresponding to the fuzzy versions are equal to the results corresponding to the crisp versions. As you can see in Table 4.1, all the resulting p-values are quite low, satisfying the 98% confidence on the superiority of the FWN-fed Sentiment Analysis results over their crisp version. Specifically, the best result that belongs to BoFWS v.93 satisfies 99.7% confidence in its superiority.

**Discussion and analysis.** The most interesting fact which can be observed in the results of Table 4.1 is the superiority of the BoFWS models over BoFS models. We expected this superiority, as mentioned in section 3.1 We believe that the fuzzy membership degrees, provided by FWN (and more generally in fuzzified WLDs) are more effective when they assign weights to each of the wordsenses independently. By independence, we do not mean wordsenses independency from synsets, as wordsense would be meaningless without having a synset. We mean bagging and considering all the wordsenses as members of one feature (i.e. their corresponding synset) decreases the resolution of the proposed modeling, whereas considering each of the wordsenses of a synset, as individual features of text contains more resolution which can help the machine to, better, model the text.

**Table 4.1. The p-values related to the null-hypothesis that the results corresponding to the fuzzy versions are equal to the results corresponding to the crisp versions.**

| p-value Table | P(Fuzzy v.93 = crisp) | F (Fuzzy v.83 = crisp) |
|---|---|---|
| Synset | 0.015 | 0.004 |
| Word-Sense | 0.003 | 0.019 |

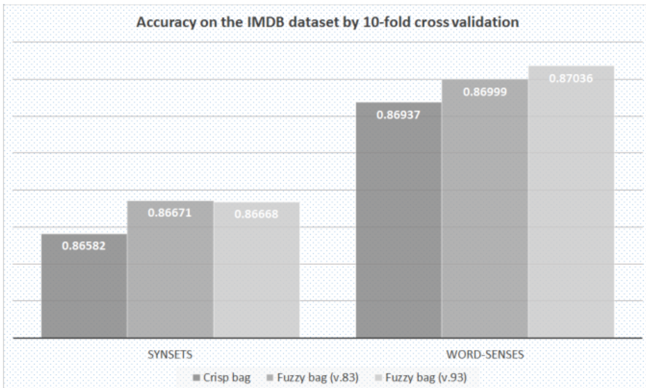



**Figure 4.1. Accuracy of the NBSVM, fed by BoS, BoFS v.83, BoFS v.93, BoWS, BoFWS v.83, and BoFWS v.93 on IMDB dataset, letting 80% of the dataset for train and the remainder-20% for test, after 10-fold cross validation.**

The other interesting fact to investigate is the relative effectiveness of v.83 and v.93. For this purpose, we measure the p-value of the null-hypothesis that fuzzy version 1993 results are equal to fuzzy version 1983 results. This p-value is 0.06 for BoFWS (i.e. 94% of confidence for the superiority of v.93 over v.83); but in BoFS results, it is as high as 0.46 (i.e. the minor difference between v.93 and v.83 results (Table 4.1) is not meaningful).

Thus, we can say that in the lower-accuracy results (related to synsets) both of the v.83 and v.93 outperform their crisp counterpart while having no meaningful difference between each other. But, in the higher-accuracy results (related to wordsenses), in addition to the fact that both of the v.83 and v.93 results outperform their crisp counterpart, v.93 result outperforms its v.83 counterpart, as well.

As mentioned in subsection 1.1.2, the superiority of the BoFWS.v.93 over BoFWS.v.83 was expected, before. In 1993, Dubois et al. (D. Dubois et al., 1993) proved that their proposed transformation is the maximally informative transformation function, and therefore, it was expected to function better than their 1983 version (D. Dubois & Prade, 1983).

Nevertheless, as mentioned, in section 3.3, BoFWS (as well as the other utilized Text Mining models) are auxiliary models, only to show that the models that utilize fuzzified WordNets function better than their counterparts that utilize the standard WordNets to conclude that the fuzzy version has more information. Correspondingly, it is expected that fuzzification of the more advanced Text Mining models, such as FRED (Gangemi et al., 2017), will result in providing pioneer accuracies in comparison with the state-of-the-art.

Albeit, even now, with the utilized low time-complexity algorithm and the very naïve BoFWS Text Mining model, the yielded accuracy on IMDB (87%) is still near to the results, outputted by very high time-complexity algorithms such as deep learning networks. For example, the results of the LSTM-based sentiment analysis algorithm (Johnson & Zhang, 2016) on IMDB (implemented and reported by (McCann, Bradbury, Xiong, & Socher, 2017)) provides 94.1%.

Considering that, one the one hand, the utilized algorithm here (NB-SVM) has very low time-complexity (unlike the LSTM deep learning algorithm), and, on the other hand, the utilized fuzzy Text Mining model (as mentioned) is an auxiliary model, this 7% of distance in accuracy is a promising sign for the performance of the fuzzified version of the state-of-the-art Text Mining models, such as FRED.

The next section addresses our experiments on the Interval version of FWN, proposed in section 3.2.

## 4.2 Interval Fuzzy Synsets

In this section, we apply the proposed algorithm in section 3.2, on an English corpus with context-categorized data. The Open American National Corpus (OANC (Fillmore et al., 1998)) has textual data on 8 different categories: technical (37%), journal (34%), telephone (18%), travel guides (7%), non-fiction (2%), face-to-face (1%), letters (0.6%), and fiction (0.4%). We also utilize the well-known UKB (Agirre E., Lopez de Lacalle O., 2014) graph-based WSD algorithm.

Before going on, we recall that, in the algorithm proposed in section 3.2, it is necessary that the frequency of all of the (M) wordsenses of each analyzed synset in all of the N categories are large enough (M × N large enough frequencies, while numbers greater than 30 are considered[40] as "Enough"). Applying our algorithm on all of the 8 categories of the OANC corpus (setting N=8), we see no Synset satisfying the mentioned circumstance. Removing the 3 smallest categories of OANC (fiction, letters, and face-to-face), in the 5 remained categories, we find 9 Synsets, satisfying the mentioned circumstance (N=5, and S=9): Small.a.01, Large.a.01, Area.n.01, Finally.r.01, Make.v.01, Difficult.a.01, Practice.n.01, Entire.s.01, and Individual.a.01. Thus, we execute our algorithm and compute the interval-valued $\mu_S(x)$ functions for these synsets.

As described, the achieved algorithm results are based on the context-uncertainty. For evaluating the results, we compare them by subject-uncertainty. This comparison is because (1.a) each subject/individual while his judgment, considers context-uncertainty as an obvious type of uncertainty because he, as a native speaker of that language, is aware of the wordsense compatibility in many of the contexts, and moreover he is requested to express his judgment, uncertainly. (1.b) The contexts that are neglected by each subject while judgment, are covered by the others who are nearer to those contexts. (1.c) Thus generally, subject-uncertainty includes context-uncertainty information as well. (2) The results of subject-uncertainty are trustable as they are yielded from human brain judgments (the data is outlier-excluded). Thus, we expect that the algorithm results (assuming accurately representing the context-uncertainty) are near to the

[40] Sometimes 30 is considered as a golden number in Statistics (Campbell, 2011). Although there are right hesitations about this (Kar & Ramalingam, 2013), yet it can be chosen as a rule of thumb.

crowdsourced results (subject-uncertainty) which (I) are trustable and (II) include the context-uncertainty information.

We made a survey by 31 native English subjects[41], proposing a compatibility assignment example, the target synset, its WordNet definition, some usage examples of its wordsenses, and asking them to assign [lower, upper] values to the compatibility of each wordsense with the synset definition, between 0 and 100. We asked compatibility, because from the one hand the membership degree of a member is its compatibility with its set (D. J. Dubois & Prade, 1980), and from the other hand the compatibility is more sound for common people. Then, based on the "average of intra over inter" tradeoff strategy, for each wordsense, the average of the 31 [lowers, upper] memberships is assigned as its crowdsourced interval. After running the experiments and before using the data, we mapped up their "max compatibility" outputs so that at least one of the wordsenses of each synset has the max-compatibility of 100; it is because we postulate that WordNet synsets have been designed, taking care of the linguistics subtleties so that each synset has, at least, one wordsense of 100% match-possible with the synset definition, as a valid possibility within the interval.

We assign the final error of the algorithm ($E_S$) for a synset ($S$) as the averaged Hausdorff distances (Munkres, 1999) of algorithm & crowdsourced intervals over its (M) wordsenses.

$$E_S = \frac{\sum_{i=1}^{M} d_H([l_a,u_a],[l_c,u_c])}{M}, d_H([l_i,u_i],[l_j,u_j]) = \max(|l_i - l_j|, |u_i - u_j|) \qquad \text{Equation 4.1}$$

Please note that the Hausdorff distance outputs absolute errors (of the left or right sides of intervals); thus, $E_S$ can be considered as the Mean Absolute Error. The $E_S$ values for the 9 synsets are as follows: (small.a.01, 0.226); (Large.a.01, 0.126); (Area.n.01, 0.245); (Finally.r.01, 0.167); (Make.v.01, 0.173); (Difficult.a.01, 0.074); (Practice.n.01, 0.206); (Entire.s.01, 0.121); (Individual.a.01, 0.083). The Mean Absolute Error over the 9 synsets is **0.158**. As it can be seen, the low distances show an acceptable conformity of our algorithm results to the crowdsourced subject-uncertainty based results.

After fully addressing the validating experiments on the Uncertainty-handling NLP-platform, it is the turn to address the experiments of the second phase of the pTER project, that is, the Uncertainty-handling text classifiers. We start with the experiments of the PNB text classifier in the next section and continue with the WNB and WNPC Weibull-based text classifiers.

## 4.3 Poisson Naïve Bayes Classifier

In this section, we propose the Authorship Attribution experiments, related to the PNB classifier, proposed in the first part of subsection 3.4.1.

*Dataset.* Selecting the dataset is equal to selecting the sub-problem to be solved by the proposed methods. In one view, Authorship Attribution problems can be categorized regarding the size of documents. There is a spectrum of very short documents such as tweets to very large documents such as books or novels.

[41] Based on the central limit theorem we force our survey participants be more than 30 so that the mean of the results of the survey follows a normal distribution (Mendel, 2007).

Considering that, as the feature extraction phase, we choose the state-of-the-art sn-gram model mainly known by the well-known study of Sidorov et al. (Sidorov et al., 2014), we adopt the same dataset, utilized in (Sidorov et al., 2014). The dataset comprises of novels from the Project Guttenberg dataset from which 13 books are selected by them. In this section, since we are going to compare our results with the same results reported in (Sidorov et al., 2014), we utilize the same dataset. However, in the following section, the last version of this dataset is utilized, which at the time of writing this dissertation is available online on the webpage of the corresponding author[42] (Sidorov et al., 2014), which includes different novels written by 7 authors.

It is necessary to note that, although this dataset may be considered too small, on the one hand, the study of (Sidorov et al., 2014) is the most well-known state-of-the-art research conducted on Authorship Attribution of large documents, and on the other hand, despite our negotiations with the corresponding author and, even, despite his support, due to unavailability of the corresponding graduated student, we could not get access to the required codes for the comparison purposes. Thus, we have to suffice to this rather small 91-book database. However, in the following, we explain the k-fold cross-validation strategy by which we approach more significant results.

*K-fold cross-validation.* Although the dataset is a standard dataset utilized in a very well-known study of this field, because the number of total books ($13\times7=91$) is not large enough to provide confidence on the significance of the results difference, we apply 13-fold cross-validation[43] to multiply the number of classifications by 13 and therefore provide more confidence on significance of the comparison results.

*Evaluation Metric.* Considering that the addressed problem is Authorship Attribution and does not have any type of retrieval (such as what in authorship verification or …), the recall, precision, and f-score measures are not meaningful. Therefore, the standard accuracy measure (i.e. $\frac{|\text{correct classifications}|}{|\text{test dataset}|}$) is adopted as the evaluation metric. In the 2-11 train-test strategy, the denominator is 1001 (7 authors × 11 test books × 13-fold cross-validation), and in the 8-5 strategy, it is 455 ($7\times5\times13$).

*Train/Test proportion.* For the experiments of this section, considering the 8-train/5-test proportion of the experiments in (Sidorov et al., 2014), we train the Authorship Attribution algorithm by 24 books belonging to 3 authors[44] (each one 8 books) and test the Authorship Attribution system on 15 books belonging to 3 authors (5 for each one). Table 4.2 Represents the accuracy (i.e. $\frac{\text{correct detections}}{\text{all books}}$) of our Authorship Attribution experiments in Author Attribution of the 15 books.

[42] http://www.cic.ipn.mx/~sidorov/

[43] Three authors had more than 13 books. We sorted the books by their title name and truncated the books located at the end to make the authors all having 13 books.

[44] Booth Tarkington, George Vaizey, and Louis Tracy

As it can be seen, the results prove the initially proposed hypothesis: The unacceptable results of Poisson-based Naïve Bayes in Authorship Attribution, as opposed to its good results in Text Classification, are due to the fact that the Text Mining models or Stylome fingerprints utilized in Authorship Attribution studies are not robust enough to model authors' Stylome. If a robust enough fingerprint is used instead, PNB gives the same good results in Author Attribution as it gives in Text Classification.

**Table 4.2. Comparison of the time complexity and the accuracy of the Stylometry Results reported in** (Sidorov et al., 2014) **and the proposed system (PNB).**

| Features number | Classifiers → / **Features proportion (%) ↓** | **SVM** $O(p \cdot n^3 \cdot m)$ | **J48 (C 4.5)** $O(n \cdot m^2)$ | **MB** $O(n \cdot m)$ | **PNB** $O(n \cdot m)$ |
|---|---|---|---|---|---|
| **400** | 0.22 | 100 | 87 | 100 | 96.41 |
| **1000** | 0.55 | 100 | 87 | 80 | 93.33 |
| **4000** | 2.20 | 100 | 67 | 40 | 92.31 |
| **7000** | 3.85 | 100 | 67 | 53 | 93.85 |
| **11000** | 6.05 | 100 | 67 | 40 | 94.36 |
| Average | **2.58** | **100** | **75** | **62.6** | **94.05** |

In the presented results, PNB is as accurate as all the other ones, having higher than 92% accuracy. On the other hand, PNB has a better scalability, due to its linear complexity. Looking at the data in Table 4.2 for the other classifiers, they either have high computational complexity (SVM) and therefore are non-appropriate when the number of authors or books increase, or they have rather a low accuracy. J48 classifier results are, on average, 19% less accurate (while having even more time complexity) and multinomial Bayes results are, on average, 31.5% less accurate than PNB. Moreover, the better accuracy of Multinomial Bayes when the number of features is low, based on the proposed experiments in the next section, seems to be related to the random choice of the train/test proportions of the dataset. It is because if the reported results of them, in Table 9 of their study (Sidorov et al., 2014), is tested on such 13-fold dataset, the best results in Table 4.2 (belonging to SVM), likely, would decrease, and in that case, the comparison of the average accuracy of PNB and other experimented methods in (Sidorov et al., 2014) would become much fairer.

However, based on the proposed experiments, PNB is recommended for the Author Attribution applications in which real-time applications are important. An example of this can be real-time Social Network analysis.

## 4.4 Weibull Naïve Bayes Classifiers

In this section, firstly, we provide the experiments specification including the compared methods and the uncertainty criteria quantification. Then, the numerical results achieved by the experiments related to each of the separately-discussed uncertainty cases are reported, and finally, analytic discussions on each of the corresponding experiments are provided. Please note that the dataset, the k-fold cross-validation strategy, and the evaluation metric are the same as reported in the previous section.

*State-of-the-Art Comparisons.* (Sidorov et al., 2014) test J48 Decision Tree, Multinomial Bayes, and SVM as the appropriate classifiers to their proposed feature model (sn-gram) and report SVM as the best. However, as seen in the previous section, PNB outperforms the decision tree classifier (J48) in, both, accuracy and computational complexity while dealing with the addressed problem. Correspondingly, in the mentioned competitor classifiers list by (Sidorov et al., 2014), we substitute the decision tree by PNB and use it beside Multinomial Bayes and SVM for comparison purposes.

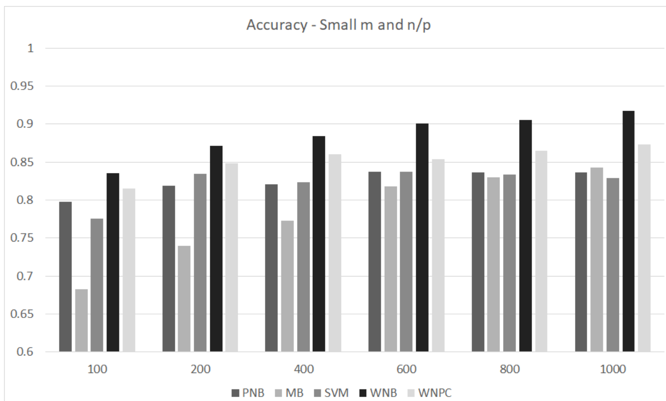
*Features number.* We repeat the experiments of each classifier on each dataset, for different feature numbers. As in (Sidorov et al., 2014), feature number stands for the number of the most frequent sn-grams of the authors by which the classifier is trained. But, instead of the spectrum chosen by (Sidorov et al., 2014), since we reproduce the experiments and are free to re-choose them, we provide the more meaningful features number of {100, 200, 400, …, 1000, 2000, 4000, …, 10000}, instead of {400, 700, 1000, 4000, 7000, 11000} utilized there.

However, for sorting the most frequent sn-grams, for each $F_i^{(j)} = \left(f_{i,1}^{(j)}, f_{i,2}^{(j)}, \dots, f_{i,n_i}^{(j)}\right)$ (i.e. the frequency sequence of the feature $j$ in each of the $n_i$ documents labeled as $c_i$), we assign the sorting score $\sum_{l=1}^{n_i} \log\left(f_{i,l}^{(j)}\right)$ so that if a non-important sn-gram is repeated frequently in a document, only because of its context (e.g. the sn-gram "Peter-said") it does not falsely increase the rank of that trivial sn-gram.

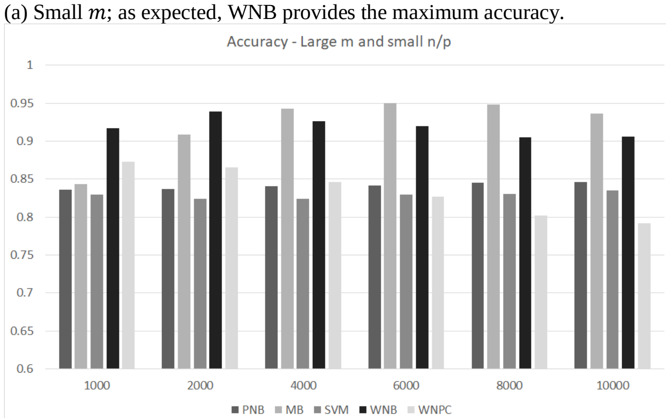
*Uncertainty criteria.* In section 3.4, we addressed small and large $m$ and $n/p$ values. However, for practical purposes, a quantitative criterion is necessary. For the $m$ parameter, getting inspired from the feature numbers reported in (Sidorov et al., 2014), we consider the feature numbers ($m$) 100-1000 as small values and the feature numbers 1000-10000[45] as large values.

[45] Values larger than 10k are also definitely possible. However, based on the (Sidorov et al., 2014) results, we know that 10k fingerprints are quite enough for Stylometry purposes (even sometimes misleading due to the role of the rare fingerprints).

For specifying the sufficient and insufficient training data (small and large $n/p$), for quantifying the concept of sufficiency in the proportion of training to the whole dataset, we utilize the concept of SW, which is addressed in subsections 1.1.6 and 3.6. SW shows how meaningful is the information of the intersection of two sets[46] to represent the similarity of them. Considering that the intersection of Training Set (TS) (with $n/p$ elements) and the whole Data Set (DS) (with $n$ elements) is TS itself, we can use $SW(TS, DS)$ to show how much TS information can represent its similarity with DS. One of the simplest and most recent proposed SWs in state-of-the-art is $\frac{s_1 \cap s_2}{\sqrt{|s_1| \cdot |s_2|}}$, proposed by Sun et al. (Q. Sun et al., 2016). We use this $SW_{\text{Sun}}$ for assay the mentioned sufficiency concept.



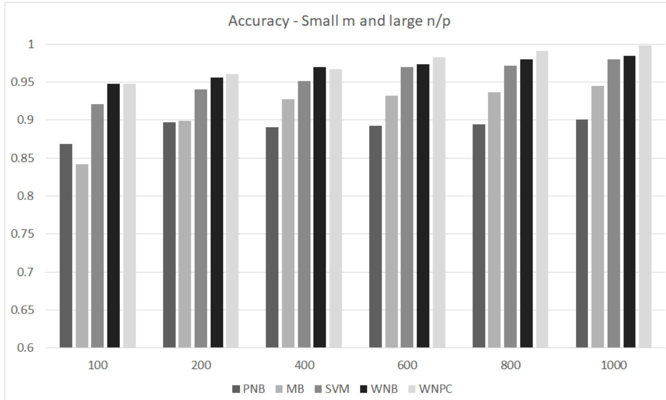

(a) Small $m$; as expected, WNB provides the maximum accuracy.



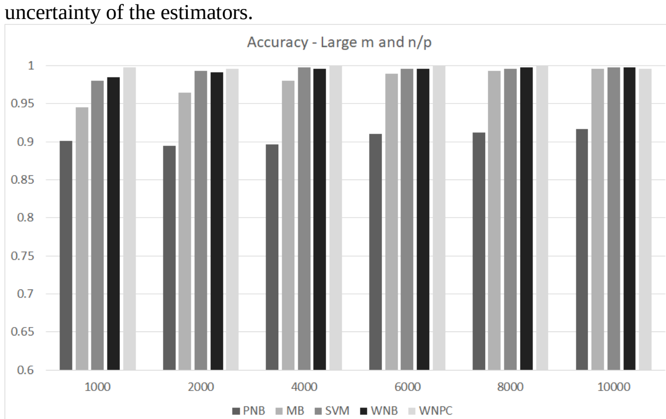

(b) Large $m$; as recommended, Multinomial Bayes should be revisited in such cases. This recommendation is in the absence of the main suggestion for this type of uncertainty that is Weibull Bayes classifier, suggested as future work.

**Figure 4.2. The Authorship Attribution accuracy when the system is trained by 2 and tested by 11 books. The results are average of 1001 predictions (7 authors, 11 test books, and 13-fold cross validation).**

[46] In the field of Recommender Systems, the mentioned two sets are rating information of two users.




(a) Small $m$; as expected, WNPC improves WNB by handling the intrinsic uncertainty of the estimators.




(b) Large $m$; although there is no uncertainty, the proposed classifiers function as well as their counterparts. Even in this equal accuracy case, WNPC is recommended due to its log-linear complexity.

**Figure 4.3. The Authorship Attribution accuracy when the system is trained by 8 and tested by 5 books. The results are average of 455 predictions (7 authors, 5 test books, and 13-fold cross validation).**

In the proposed experiments, for the "insufficient TS case," we assign 2 documents for TS (and 11 for test) because training by two items is the minimum requirement and if we use one item for training, then some Weibull estimators cannot function. The corresponding $SW_{\text{Sun}}$ in this case is 39%. For "sufficient TS case," we adopt the same adopted strategy by Sidorov et al. (Sidorov et al., 2014) that is assigning 8 documents for TS (and 5 for test). The corresponding $SW_{Sun}$ for this case is 78%. As a rule of thumb, we set $SW_{Sun}^{-1}(50\%) = (n/p)_{threshold}$ as the threshold to specify whether $n/p$ is sufficient (greater than $SW_{Sun}^{-1}(50\%)$) or not (smaller than $SW_{Sun}^{-1}(50\%)$).

Results and Discussion. Figure 4.2 plots the accuracy results related to the small $n/p$ values (2-11 train/test strategy). Figure 4.2 (a) illustrates only the results of low $m$ values. As expected, WNB excels in its counterparts. WNPC does not function well, because as mentioned in section 3.4, when there is insufficient training data, only MoM provides good estimations and the other Weibull parameter estimators do not provide very appropriate results.

Figure 4.2 (b) plots the results for the case with insufficient training data but large $m$ values. As mentioned, the main suggestion for this type of uncertainty is the non-naïve version of WNB (i.e.

Weibull Bayes). However, because Weibull Bayes is a rather complex mathematical research requiring a separate future work, as recommended in section 3.4, the already existing non-naïve Multinomial Bayes classifier should be "revisited" because it avoids the too many ($\binom{m}{2}$) naïve assumptions of conditional independence. As expected, the experiments confirm the mentioned argumentation. Multinomial Bayes provides results, even, superior to WNB that is equipped with the very flexible Weibull probability distribution.

Figure 4.3 represents the results related to the large values of $n/p$ (8-5 train/test strategy). Figure 4.3 (a) is dedicated to the small values of $m$. The results confirm the expectation provided by Bounhas et al. (Bounhas et al., 2014) and as expected the proposed "possibilistic" classifier, WNPC, successfully handles the intrinsic uncertainty of estimators by utilizing the advantages of all, at once.

Figure 4.3 (b) represents the results related to large values of $m$ and $n/p$. Due to the absence of uncertainty, the results show the almost equal accuracy of the SVM, WNB, and WNPC. However, considering that WNPC has the best computational complexity, still, for this uncertainty-less case, WNPC is recommended.

After fully covering the experiments related to the Uncertainty-handling text classifiers, now, it is the turn to address the Uncertainty-handling CF and the proposed GSWs as well as the fuzzy optimization algorithm utilized for optimizing them, in the next section.

## 4.5 Generalized Significance Weights and Linear-complexity Fuzzy Optimizer

In this section, we present the corresponding experiments related to the proposed GSWs in section 3.6 by means of the proposed linear-complexity algorithm presented in section 3.7, which are conducted on the Movielens 100k dataset (Harper & Konstan, 2015).

We consider 5 types of plots:

1. The plots in which the efficiency of the different GSWs proposed in section 3.6 are compared, to check which of them (in average) works better and adopt it for the next experiments (for avoiding the plot redundancy, occurred in the case that we examined the 8 proposed GSWs in each of the following plots).
2. The plot by which the performance of the proposed MOA is proven to be almost as accurate as an Exhaustive Search algorithm.
3. The plot in which the effect of SW generalizations is presented; the Total Error plot for the examined SW, before and after generalization, is utilized to represent the alleviation of the error measure, achieved by generalization.
4. The plot in which variations of GSW efficiency is examined while changing the train/test dataset proportion, to prove the sustainability of the examined GSW.
5. The plot in which the efficiency of the examined GSW is checked while adopting different Similarity Measures.

However, for being able to compare the efficiencies (capability of each SW for improving the accuracy or alleviating the Total Error), we compute improvement percentage of the SWs than the case in which no SW is used. The utilized evaluation metric is:

$$\text{Total Error alleviation (\%)} = \frac{Error(CF_{Normal}) - Error(CF_{weighted})}{Error(CF_{Normal})} \times 100. \qquad \text{Equation 4.2}$$

The Equation 4.2, in the ideal case (if $Error(CF_{weight})$ was zero) outputs 100% of improvement and in the neutral case ($Error(CF_{Normal}) = Error(CF_{weight})$) the formula outputs 0% of improvement[47].

Having the evaluation metric clarified, in the following subsections we propose the abovementioned comparisons. We investigate this improvement for different neighbor numbers (1-10) to provide a better intuition to the provided evaluations. Note that the parameters of GSWs are optimized, separately, for each neighbor number. In other words, each neighbor number provides an individual optimization problem, and the plots, proposed in the following subsections represent the improvement achieved by each SW / optimized GSW.

As mentioned, in each of the following subsections one factor is under examination, because if we examine all of them in each subsection, it would need many plots in each step that is out of the

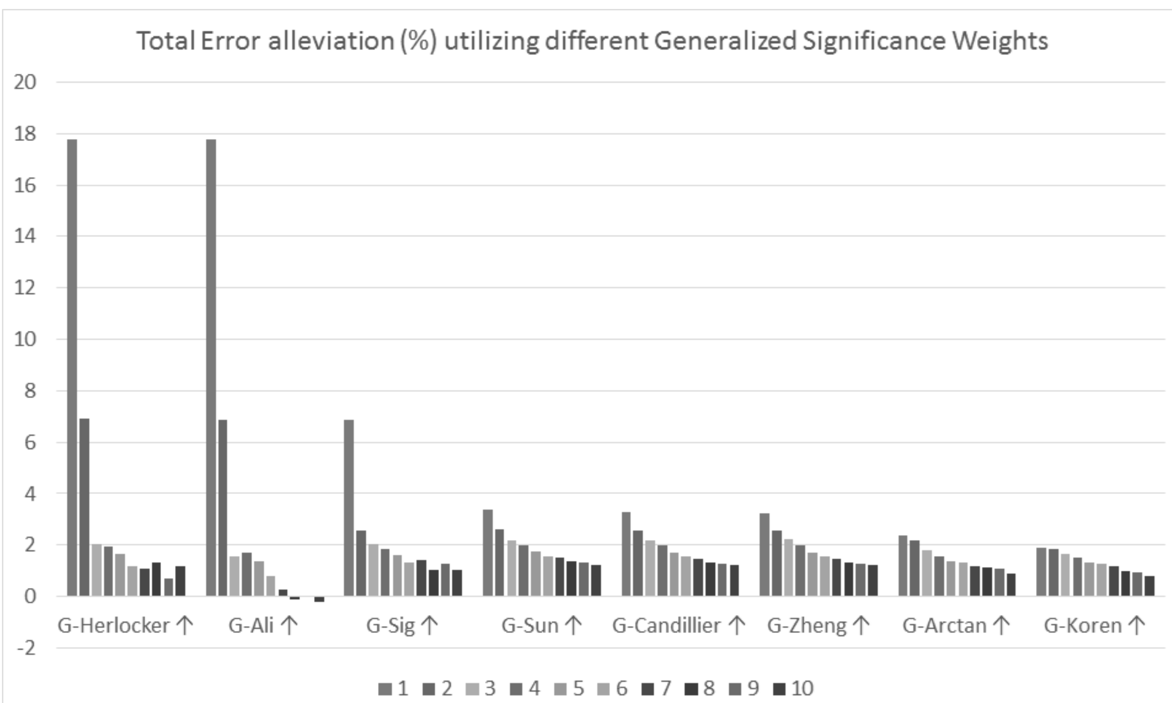



**Figure 4.4. Total Error alleviation (%) while adopting different GSWs. Colors represent the number of neighbors.**

[47] Considering that we always expect an improvement by the proposed GSWs and never expect the ideal case, we expect a number, greater than zero and less than 100 for improvement percentage.

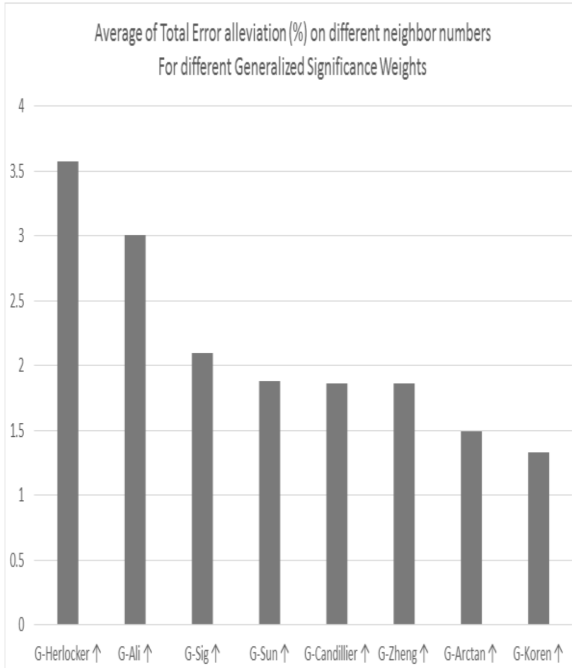



**Figure 4.5. Total Error alleviation (%) while adopting different GSWs, the alleviation % is averaged over the neighbor numbers 1-10.**

size of this section. In the conducted experiments, $simil_{normal}(u, u')$ by default, is set to Pearson similarity measure as the successful similarity measures in most of the mentioned SW studies; the default train-test proportion is set to 20-80 as explained in section 3.1; and the default GSW (as mentioned above) will be determined after analyzing the results of the following subsection. In the following subsections, we address the abovementioned plots, one by one.

### 4.5.1 Significance Weight Functions

In this subsection, we compare the accuracy of the 8 proposed GSW.

As it can be seen, in Figure 4.4, the best performance belongs to Generalized Herlocker SW. Specifically, it has a much improvement in Total Error when the number of neighbors is few (18% for n=1 and 6% for n=2). The other fact is Herlocker GSW preserves a minimum Total Error alleviation of 1% even when the number of neighbors is many.

Considering that the Figure 4.4 compacts all the experiments related to the 10 neighbors, at once, visually comparing them is not convenient. Therefore, we also depict the Figure 4.5 which shows the average Total Error alleviation (%) via 1-10 neighbors.

As it can be seen in Figure 4.5, the expectation of observing better performance in Generalized Herlocker can be numerically verified. Thus, we would adopt Herlocker GSW for the remainder of experiments.

### 4.5.2 Accuracy of the Optimization

For proving the accuracy of the proposed (linearly-trainable) fuzzy SM, we plot the Total Error alleviation (%) for 2 optimization methods. Once, we apply Exhaustive Search on a dense grid (31×31=961 points (for each neighbor number) on the search domain, specified in section 3.7, on the main computationally-expensive cost function (Equation 16) to approximately represent the global optimum solutions. Once again, we apply the same Exhaustive Search on the proposed fuzzy SM and depict the Total Error alleviation (%) yielded by each of them, to represent that the error caused by the proposed fuzzy SM is almost equal to errors of the Exhaustive Search algorithm. As it can be seen in Figure 4.6, the Total Error alleviation (%) of the proposed MOA is almost the same as the one adopted by the Exhaustive Search. It should be considered that the Exhaustive Search is done on the testing dataset (that guarantees the best solution) while the MOA is trained on the training dataset and the depicted results are the test results on the unseen test dataset, as occurred in the real application.

The high accuracy of the proposed algorithm should be considered, besides the fact that the proposed optimizer is from the one hand computationally cheap (like every other SM) and from the other hand linearly trainable -unlike others- and it requires a few numbers of training experiments.

### 4.5.3 Effect of the Generalization

In this subsection, we depict the improvement percentage achieved by the standard and the generalized versions of Herlocker SW to check the improvements, while generalization.

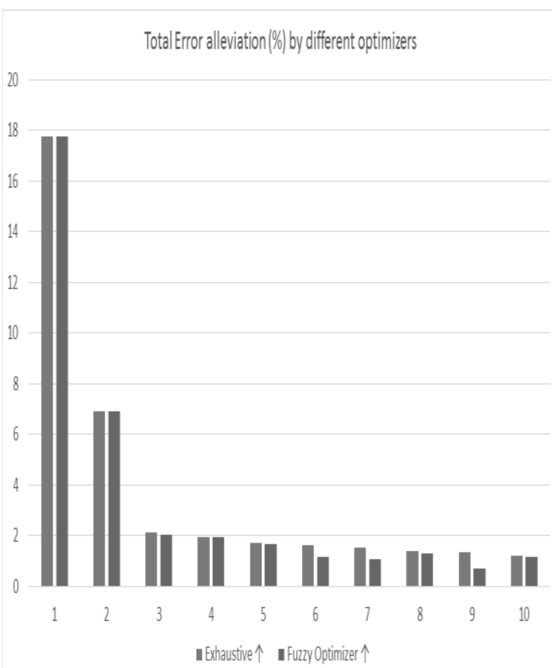



**Figure 4.6. Comparing the performance of applying Exhaustive Search and the proposed fuzzy SM on generalized Herlocker for optimizing its parameters. Both of the errors are almost equal for both of the optimizing algorithms.**

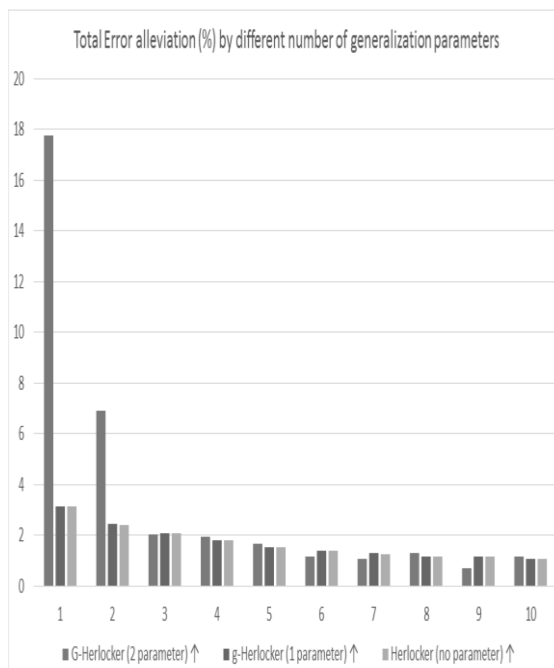



**Figure 4.7. Comparing the performance of the Herlocker GSW while dealing with different proportions of train / test set from the main dataset.**

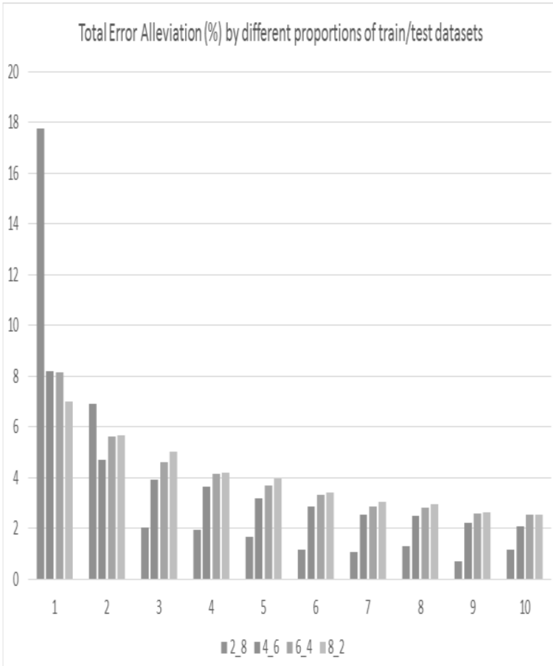



Figure 4.8. The improvement achieved by standard Herlocker (2002), univariate Herlocker (2004) and the proposed generalized Herlocker

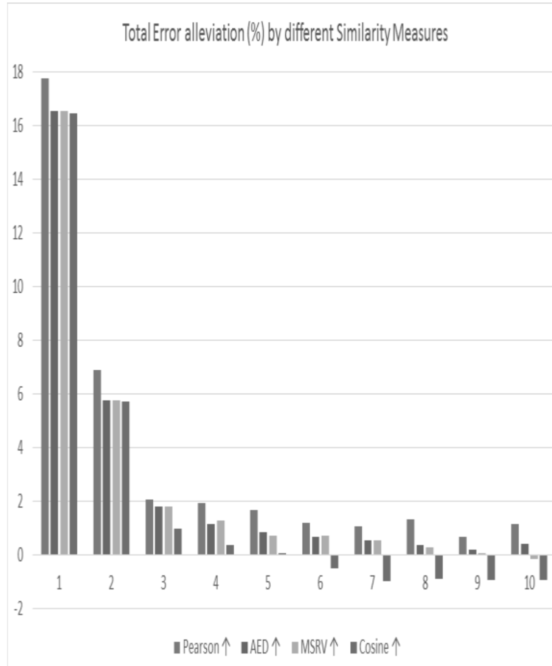



Figure 4.9. The performance of the Herlocker GSW while dealing with different Similarity Measures: Pearson, Adjusted Euclidean Distance (AED), Moment Similarity of Random Variables (MSRV), Cosine.

Figure 4.7 plots the Total Error alleviation (%), for standard Herlocker (2002) (Jon Herlocker et al., 2002), Flexible Herlocker (2004), and the proposed Generalized Herlocker. As it can be seen, in Figure 4.7, the two previously proposed versions of Herlocker (parameter-less in 2002 (Jon Herlocker et al., 2002), and univariate in 2004 (McLaughlin & Herlocker, 2004)) have almost a same performance, whereas, the proposed generalization (bivariate GSW) proposes a meaningfully better performance. It should be noted that the better performance occurs when the number of neighbors is low, and when dealing with many neighbors, the improvement is less meaningful.

This fact proves the initially discussed hypothesis, namely, the existing SWs have significant potency for causing more improvement in Memory-based CF, especially in the cases, which cold-co-start problem is severe.

### 4.5.4 Effect of the Train/Test Proportion

In this subsection, we check the performance of the Herlocker GSW while dealing with different proportions of train/test dataset from the main dataset. We experiment for 4 different cases: 20-80, 40-60, 60-40, and 80-20, when the first number stands for the proportion of the train set and the latter the proportion of the test set (which would be the remainder of what training set occupied).

As it can be seen in Figure 4.8, on the one hand, in all of the cases, we have a meaningful improvement in Total Error alleviation, and from the other hand, the improvement achieved by the less proportion of training set is more when the neighbor numbers is few, while for the many neighbor numbers, the more improvement occurs when we have more proportion for training.

This fact shows that the proposed algorithm improves the CF accuracy, not only while the availability of only very few predictor neighbors (sever cold-co-start), but also when the number of available predictor-neighbors increases, the proposed optimizer will gain the potency of even more alleviation if more information (training data) is available. In other words, not only GSWs can assist in reducing the cold-co-start problem, but also they still are helpful when the severity of the cold-co-start problem is less. It should also be noted that the improvement decreases by a decrement of cold-co-start severity.

### 4.5.5 Effect of the Similarity Measures

In this subsection, we check the performance of the Herlocker GSW while dealing with different Similarity Measures. As mentioned, in the introduction, we check 4 different Similarity measures: Pearson, Cosine, Adjusted Euclidean Distance, and Moment Similarity of Random Variables (addressed in section 1.1). Figure 4.9 proposes the comparison results.

As it can be seen from the results, the best performance occurs for Pearson Similarity Measure. Indeed, it was expected because it is the most common Similarity Measure in the corresponding state-of-the-art. The other important point is Cosine is not a trustable Similarity Measure when dealing with GSWs. It is because of the presence of even (although low but yet) negative performance values when dealing with Cosine Similarity Measure. However, as depicted, (except some cases of Cosine Similarity Measure), all the utilized Similarity Measures have a meaningful improvement on CF.

# 5 Concluding Remarks and Future Trends

In this dissertation, we propose the necessary foundations for the probabilistic/possibilistic Text-based Emotion Rating (pTER) system. Correspondingly, in this chapter, as a conclusion, we first provide a summary form the contribution achievements of this dissertation and then address the future works of this dissertation, which most of them are already planned and/or under progress.

## 5.1 Concluding remarks

This dissertation includes research studies in three dimensions of computer science research:

(1) Sections 4.1 to 4.3 provide an Uncertainty-sustainable platform for NLP. More in details, section 4.1 and 4.2 provide fuzzifier algorithms for every WordNet-like Lexical Database (WLD) and section 4.3 presents two auxiliary Text Mining models to evaluate the precision and the validity of the proposed fuzzifier algorithms.

(2) Section 4.4 presents Uncertainty-sustainable classifiers that (as the name implies) have high sustainability in uncertain circumstances that normally occur while dealing with Social Networks sparse information.

(3) Sections 4.5 introduces the cold-co-start problem which frequently happens in Sentiment Prediction paradigm and, moreover, upgrades the state-of-the-art of cold-start alleviating methods by proposing Generalized Significance Weights (GSW) which mainly alleviate this sub-problem. Then, as the final ring of its contribution chain, this section presents a linear complexity fuzzy Metaheuristic Optimizing Algorithm (MOA) for finding the optimal parameters of the proposed GSWs in real-time.

The abovementioned three phases in this dissertation, are the necessary foundations for a successful Uncertainty-sustainable Sentiment Prediction system. In the following, we provide the related concluding remarks of these three contribution categories. respectively.

### 5.1.1 Uncertainty-sustainable Text Mining platform

In section 3.1, firstly, we propose an algorithm for the automatic generation of fuzzy membership functions for definite synsets of the existing WLDs. The proposed WLD-fuzzifier algorithm is mainly based on the definition of possibility and its relationship with membership functions. The validity of its results is proven, mathematically, by the Probability and Possibility Theorem methods.

Moreover, we apply the proposed algorithm to the English language to generate the fuzzified version of WordNet, called FWN, and publish it online. Thereafter, we apply the presented FWN to Sentiment Analysis, for experimentally proving the FWN efficiency. To this end, we introduce two auxiliary fuzzy Text Mining models, i.e. Bag of Fuzzy Synsets and Bag of Fuzzy WordSenses (BoFS and BoFWS) in section 3.3, which can be fed by the proposed FWN. In section 4.1, we apply the proposed Text Mining models to the Sentiment Analysis process and prove the superiority of the mentioned fuzzy models over their crisp counterpart.

It is necessary to note that neither BoFS nor BoFWS are considered as the contributions of this dissertation, and they are utilized, only, to illustrate the superiority of utilizing the additional information provided by FWN over the case in which this information is absent.

As a parallel study of this phase, in section 3.2, we discuss the information lack in the already existing standard fuzzy synsets from 3 different aspects: method-uncertainty, context-uncertainty, and subjects-uncertainty. Then, we propose an algorithm for constructing interval version of fuzzy synsets, based on the context-uncertainty (resulting interval-valued membership degrees). Comparison of the interval results with crowd-sourced data (subject-uncertainty), in section 4.2, shows an acceptable conformity (0.158 distance over 1) for the algorithm to the crowd-sourced results.

We would like to remind that (as the title of this subsection inspires) utilizing this platform, is expected to make its applications in Text Mining applications to be more sustainable under uncertainty-dealing circumstances. Therefore,

### 5.1.2 Uncertainty-sustainable Text Classification

In this phase of the dissertation, we propose two Machine Learning based classifiers. It is necessary to note that because of the request of the H2020 IDENTITY European project, we applied our classification research studies on the Authorship Attribution field of Computer Science. However, due to the very high similarity of the different aspects of Text Classifiers the proposed classifier can easily be used in the Sentiment Classification, as well.

The proposed classifier is a fast and accurate classifier, called Poisson Naïve Bayes (PNB) for Author Attribution. PNB has linear time complexity while providing highly accurate results. PNB revisits Poisson-based classifier and shows that the low accuracy of the previous applications of this classifier in Author Attribution is not because of the classifier itself, but it is because of the Text Mining model utilized as the Stylome fingerprint, in those studies. PNB utilizes sn-gram, a new successful Text Mining model that is proven to be a robust Stylome fingerprint. As expected, Authorship Attribution experiments (on the Project Guttenberg) prove that PNB provides accurate results despite its linear complexity. For a fair comparison, we perform the compared experiments (e.g. SVM, C4.5, and Multinomial NB) on a k-fold cross-validation, so that the superiority of the PNB would become clearer.

While the experiments of this study prove the high performance of PNB, the second research of this phase covers mathematical extensions and elaborations of the proposed system:

In the second study of this phase, we provide proposals for different uncertain circumstances, occurring in Authorship Attribution problem. Considering $m$ as the number of features of a document, $n$ as the total number of documents (for the entire authors), and $p$ as the number of the entire classes (authors), we divide such uncertain circumstances into 3 categories (I) insufficient features and insufficient training documents per class (small $m$ and $n/p$), (II) sufficient features but insufficient documents per class (large $m$ and small $n/p$), and (III) insufficient features but sufficient documents per class (small $m$ and large $n/p$). Then, we illustrate why each of the mentioned suggestions or the proposed methods in Table 5.1 fits the addressed type of uncertainty.

**Table 5.1. A structured view of different possible uncertain circumstances in Authorship Attribution and the fittest classifiers for each of them.**

| | $m\downarrow$ | $m\uparrow$ |
|---|---|---|
| $\frac{n}{p}\downarrow$ | **Weibull Naïve Bayes (WNB)** | **Weibull Bayes (**Accuracy↑) |
| | Accuracy↑ | **Multinomial Bayes (MB)** |
| | | Accuracy↑ & Complexity↓ |
| $\frac{n}{p}\uparrow$ | **Weibull Naïve Possibilistic (WNPC)** | **Weibull Naïve Possibilistic (WNPC)** |
| | Accuracy↑<br>Complexity↓ | Accuracy = State-of-the-Art<br>Complexity↓ |

As represented in Table 5.1, for the uncertainty type I, for the first time (to the best of our knowledge), we apply the Weibull Naïve Bayes (WNB) classifier for text classification. For the uncertainty type II, we analyze why WNB is not expected to function well and propose Weibull Bayes for this type of uncertainty. However, because Weibull Bayes is complex enough to require a separate study, we mention it as a future work to be addressed, and for now, we explain why revisiting the non-naïve Multinomial Bayes (MB) is expected to still have the best accuracy while dealing with the uncertainty type II. For the case III, we present a "possibilistic" version of WNB (the so-called WNPC) that handles the intrinsic uncertainty of the WNB classifier and illustrate why such intrinsic uncertainty exists in WNB only under the uncertainty type III circumstance.

The computational complexity of the proposed methods is also analyzed. Multinomial Bayes, WNPC, and WNB provide the low computational complexities of $O(m \cdot n)$, $O\left(m \cdot n \times \log\left(\frac{n}{p}\right)\right)$, and $O(m \cdot n \times t^3 \cdot \log_2(A))$, respectively, where $t$ and $A$ are data-independent factors related to function approximation. Considering that $\log\left(\frac{n}{p}\right)$ and $t^3 \cdot \log_2(A)$ are either data independent or have practical small upper bounds, all of the proposed classifiers are appropriate for the real-time applications, as well.

We utilize the addressed classifiers besides one of the most well-known state-of-the-art feature extraction models of Authorship Attribution (i.e. sn-grams (Sidorov et al., 2014)) to reach a complete Authorship Attribution system required for the experiments. Considering that the most recent and well-known study on the sn-gram model is (Sidorov et al., 2014), we compare the results of our experiments with the results yielded by the addressed classifiers in (Sidorov et al., 2014). The conducted experiments confirm the analyzed expectations.

It is notable that the mentioned uncertainty-dealing circumstances potentially occur in both small- and large-scale problems. Regarding the factor $n/p$, this fact is because $n/p$ represents the number of training documents "per" author (and not the entire training documents) and potentially happens even in large-scale problems. Considering the factor $m$, the mentioned fact is because, regardless of the existence or lack of numerous features in the training documents, if the uncertainty caused by the smallness of $m$ is managed, the small values of $m$ are quite appropriate for increasing the classification speed.

### 5.1.3 Uncertainty-sustainable Collaborative Filtering

The third phase of the methods proposed in this dissertation addresses a special case of the cold-start problem occurring in Memory-based CF. Traditional similarity measures, which are utilized in Memory-based CF, are vulnerable under the cold-start problem, especially by cold-user which is a sub-category of them (The condition in which users' ratings are few). One of the proposed methods for solving this problem is using Significance Weights (SWs), which assign a low mass to the similarity value of two users when they have few common data and high mass vice versa.

In the previous two decades, many SWs have been proposed in the state-of-the-art. In this phase of the dissertation, we first define a sub-category of the cold-user problem (cold-co-user)[48] and illustrate that the SWs are mainly focusing on either cold-co-user or cold-co-item sub-problem. Then, we survey the proposed SWs, in the state-of-the-art, and discuss on their potential excellence if they become generalized by involving y-intercept parameter ($y$) as well as scale parameter ($z$). Next, for becoming able to find the globally optimal $y$ and $z$ in real-time Memory-based CF applications, we propose a linear time complexity MOA based on SMs. The refreshing $(y, z)$ output can be utilized to specify the optimal SW for the target application, online.

The conducted experiments on Movielens 100k dataset, for various numbers of nearest-neighbors (1-10), various proportions of train/test, various Similarity Measures, various SWs, prove the meaningful superiority of the GSWs over the standard SWs in the presence of the cold-co-start problem. We also compare the results, obtained by the proposed MOA with the results obtained by the Exhaustive Search optimizer, and prove their nearly equal results, while the proposed optimizer has linear time complexity. Meanwhile, it is worthy to note that, generally, the proposed MOA is recommended for any similar problem, in which, (1) the cost function is expensive and time-consuming to be computed, (2) local optimums are avoided, and (3) linear time-complexity is required.

The mentioned experiments, eventuate that SWs must be substituted by GSWs in their future applications, especially considering the linear time-complexity of the proposed MOA that makes GSWs applicable even in real-time applications. Although the experiments of real-time optimization are left as a future work of this study, they already can be performed without the requirement to any additional theoretical contribution. We will address this potency in subsection 5.2.1.

## 5.2 Future studies

As mentioned in section 2, this dissertation provides the necessary foundations for probabilistic/possibilistic (Uncertainty-handling) Text-based Emotion Rating (pTER) system. Although (as expected) the mentioned three phases of this research represented in this dissertation are sufficient for such a Sentiment Prediction system to be developed, yet, as mentioned in section 1.2, we have planned several research papers in two main categories: Scalar pTER and the interval pTER systems, each of which enhancing the project from one point of view. Half of the fundamental (and foundational) papers of this category (i.e. scalar pTER) are covered in this

[48] Similarly cold-co-item

dissertation. However, still, there are five remained enriching research papers that can make the scalar pTER system more robust. Additionally, the interval pTER system comprises ten research papers, which are either under-progress or defined but in-queue research papers of the mentioned researchers in section 1.2. It is necessary to note that, evidently, the in-queue research papers (albeit defined), due to the nature of science and research, are subject to modification based on the state-of-the-art requirements. However, the defined papers have a meaningful distance with the borders of science, right now.

In the following, we provide a more detailed description of the defined future works of the two proposed pTER systems.

### 5.2.1 Scalar pTER studies

Regarding that five (over the six) research studies proposed in this dissertation belong to scalar pTER project, there would remain five under-progress planned future works for scalar pTER project, each of which are being addressed by one master student.

## *Fuzzy FRED*

In section 3.1, the fuzzified WordNet and fuzzy WLDs has been proposed. Based on them, in section 3.3 two auxiliary Text Mining models (i.e. BoFS and BoFWS) has been proposed in the corresponding section. As mentioned, the superiority of BoFS & BoFWS over their crisp versions inspires that: When FWN is effective in improving such naïve bag-models that model the input text with high information loss, a fortiori, the information potency of FWN shall be effective in improving advanced Text Mining models that preserve the input information, as much as possible. Proposing novel Text Mining models that have the potency of being fed by fuzzified WLDs is one of the planned future works of the pTER system.

Recently, a novel graph-based Text Mining model, called FRED, has been proposed for converting the natural language text to RDF and OWL (Draicchio, Gangemi, Presutti, & Nuzzolese, 2013)(Gangemi et al., 2017). Considering that FRED has received broad attention by the Text Mining society, we found it as one of the best candidates to be upgraded by the fuzzified WLDs.

Mr. Tavana is a master holder of Logic science and a master researcher of a second master course on the Artificial Intelligence in FUM[49]. The first research of his master thesis is defined on the fuzzified FRED project.

## *Conceptualized WordNet*

Another defined research line on the "Uncertainty-handling platform" phase of the scalar pTER system goes one step further than the progress of fuzzified WLDs and focuses on the "concepts" in WordNet. Although already some important research studies have been conducted on the "concepts" in Natural Languages and Ontologies, yet, there is no conducted study for conceptualizing WLDs, that is, extracting umbrella concepts for several synsets to provide a more comprehensive view while working with them. Mr. Mohammadian is a master student of Artificial

---

[49] Ferdowsi University of Mashhad

Intelligence in IUST[50] whose master thesis is defined on conceptualizing WordNets. Providing such a more comprehensive view can handle the uncertainty caused by the sparsity of the synsets, which especially occur in the Social Networks extracted information.

### *Fuzzified WordNet in domain-ontology-based Sentiment Analysis*

After enriching the comprehensive Uncertainty-handling platform for NLP, it is the turn to enrich the Uncertainty-handling Sentiment Analysis methods.

Recently, a novel fuzzy domain-ontology-based Sentiment Analysis (Ali, Kwak, & Kim, 2016) has been proposed and attracted the attention of several researchers in fuzzy Text Mining society. However, the utilized fuzzy ontology is not fed by fuzzy WLDs and simply has fuzzy components. Considering our developed rich platform for Uncertainty-handling in NLP, we suggest the upgrade of the mentioned system by the proposed fuzzy WordNet in this dissertation.

Mr. Karrabi is a master student in FUM who passes the master course of Artificial Intelligence. This research has been defined as the first research of his master thesis.

### *Fuzzy Sentilo*

Recently, a very popular upper-ontology-based Sentiment Analysis system is provided, which is called Sentilo (Reforgiato Recupero et al., 2015). Sentilo is fed by FRED which is also the other popular research of the Sentilo presenters. Considering that the FRED system is going to be fuzzified in the first phase of the scalar pTER project, correspondingly, the fuzzified version of the Sentilo system is the most straightforward research for Uncertainty-handling ontology-based Sentiment Analysis.

Mr. Moradi is a 1$^{st}$-year master student of IUST who is already working on his seminar on "A survey of the Sentiment Analysis methods with cognitive approach." His next year research, as the first "research-track" paper of his master thesis, is defined on fuzzy Sentilo.

### *pTER system*

The very last research of the pTER system is the pTER itself. The pTER paper is the final product of the proposed scalar pTER system. Although by means of the presented foundational studies of this dissertation pTER already can be developed, yet, we let this very novel idea to be disseminated as an academic paper, only after gaining its total potency to receive the highest possible attention from the society. Thus, it is not included in the scope of this dissertation. After the other abovementioned studies will be added to the scalar pTER, this research will be conducted by Mr. Kalamati after defending his Artificial Intelligence master thesis in FUM.

#### 5.2.2 Interval pTER studies

In this subsection, we address the ten defined/planned studies on the Interval pTER project. Please note that the majority (five over six) of the proposed research papers in this dissertation are enumerated in the "scalar" pTER project. However, it is notable that the mentioned five papers,

[50] Iran University of Science and Technology

not only provide foundations of the scalar pTER project but also extending them in another dimension of uncertainty would result in another generation of Uncertainty-handling Sentiment Prediction systems, that are, interval-handling systems.

Disregarding the abovementioned fundamental role of the five (over six) conducted research studies in this dissertation, considering that in the interval pTER system, the author of this dissertation has one research paper (among the 10 defined papers) that is already published (Hossayni, Rajati, et al., 2016), and therefore the other nine research studies on the interval pTER system are engaged with the other students, in the following, we provide a brief explanation for each of them.

### *Probabilistic Interval Fuzzy sets*

As mentioned in section 3.2, one of the proposed approaches to WordNet fuzzification is interval fuzzification. However, the algorithm has the drawback of requiring enough frequency of synset wordsenses, in all of the categories, that forces either using several corpora as input (high computational cost) or having few numbers of context-categories (low contextual information) to increase the per-category information. Thus, for making the algorithm useful for practical purposes (the applications of WLDs such as NLP and Knowledge Engineering), the algorithm should be upgraded by utilizing method-uncertainty, in which even using low wordsense frequencies, we can construct the interval fuzzy synsets. However, because the already existing theory in Type-2 Fuzzy Sets does not seem sufficient for constructing a generalizable algorithm of what proposed in section 3.2, Mr. Alizadeh as a part of his Ph.D. dissertation (in IUST) has introduced a new mathematical concept, based on the probabilistic fuzzy theory proposed by (Meghdadi & Akbarzadeh-T, 2001). He also provides mathematical proves on the specifications of his proposed model, based on which the mentioned generalization algorithm can be conveniently conducted.

### *Interval fuzzifier for WordNet-like Lexical databases*

As mentioned, as a future work of section 3.2 (Hossayni, Rajati, et al., 2016), we propose the generalization of the interval-WordNet idea. In this regard, in a parallel study, Mr. Alizadeh has proposed an algorithm for constructing the PI-F version of WLDs of any language, and correspondingly, constructs the PI-F synsets of English WordNet. For proving the accuracy and validity of the constructed PI-F WordNet, he proposes an auxiliary Text Mining model (Bag of Interval Fuzzy Synsets) based on the PI-F synsets and applies it to the Sentiment Analysis problem. The superiority of the Sentiment Analysis accuracy in the presence of PI-F synsets, over the case in which T1-F synsets is utilized, is expected to prove the more information support, provided by the constructed PI-F synsets in comparison with T1-F synsets.

### *Fuzzy ConceptNet*

In the in-progress papers of the scalar pTER, we enumerated the “Conceptualized WordNet” paper, as an in-progress paper of Mr. Mohammadian. However, his next (in queue) study is defined on the application of the idea proposed in Fuzzy WordNet and PI-F WordNet papers on his in-progress study to develop the Uncertainty-handling version of Conceptualized WordNet.

### *An Interval Approach to Relation Extraction*

As mentioned in the scalar pTER planned works, the in-progress paper of Mr. Tavana is the fuzzified version of the FRED relation extraction (i.e. text to RDF/OWL). The second (in queue) research paper of Mr. Tavana is defined on the utilization of PI-F WordNet and Fuzzy and PI-F conceptualized WordNet, to make the FRED relation extraction system, more capable in managing the uncertainties.

### *Fully Uncertainty Managing Fuzzy Text Classifier*

In section 3.4, we introduce WNPC. However, getting inspired by the very popular research of (Bounhas et al., 2014), we became motivated to provide a more Uncertainty-handling version of the WNPC. The proposed classifier by (Bounhas et al., 2014) can handle the interval-valued features in the testing phase and uncertain classes in the training phase. Mr. Alizadeh in his parallel planned work has presented a version of WNPC which can handle the uncertain classes, as well as the uncertain (interval), attributes both in the training and the testing phases of the classification. However, because handling the fine-grained classes in the test phase (and outputting the compatibility degree of the item with each class) is equal to fuzzy classification, we call this classifier as Fully-uncertainty Management “Fuzzy” Text Classifier.

### *Unsupervised Ontology-based Interval-output Sentiment Analysis*

The first (in progress) paper of the master thesis of Mr. Karrabi, as mentioned in the previous subsection is a fuzzified version of the study of (Ali et al., 2016). However, due to the provided interval fuzzy potency by the research papers of Mr. Alizadeh and Mr. Mohammadian in providing the PI-F Synsets and Concepts, the second paper of Mr. Karrabi is defined on the PI- Fuzzification of the mentioned study, for providing the capability to handle more uncertainty.

### *Sentiloin: The first uncertain Cognitive Sentiment Analysis tool*

Intervals and, one step further, Random Variables are generally welcome in the management of Uncertainty-handling. However, until now, in the abovementioned studies, the PI-F WordNet and PI-F ConceptNet were subject to being utilized in different planned research papers. We call this approach as the top-down approach in the utilization of intervals for uncertainty management, that is, the utilization of interval-Uncertainty-handling platforms as a more informative source.

However, there is also another approach which may be called bottom-up approach. Right now, the Sentilo (Reforgiato Recupero et al., 2015) Sentiment Analysis system provides fine-grained values for the sentiment of a sentiment holder about a subject, as the output of the Sentilo system. We can upgrade the Sentilo so that (in addition to feeding the existing PI-F lexical sources) its output becomes interval-valued and it can also consider the method uncertainty (explained in section 3.2) as its output. This paper is the subject of the second (in queue) study of Mr. Moradi.

### *Sentiment Variables and Category-based Collaborative Filtering*

For more Uncertainty-handling in the Collaborative Filtering (CF) systems, Mr. Kalamati, as the first paper of his master thesis in FUM, proposes a new method by which all of the Memory-based CF system items are categorized under small number of classes (e.g. genres) and ratings of users on different categories are modelled by a parametric probabilistic model using only two parameters. In this regard, ratings of each user about each category are modeled by a Beta Random Variable, which has a high potency in representing Sentiment and Emotions. The parameters of the mentioned Probability Distribution Function are estimated by a Bayesian estimator (with linear time complexity). In his study, it is going to be shown that utilizing the mentioned method and correspondingly compressing the information by the proposed probabilistic model provides a satisfactory accuracy while outstandingly keeping the complexity to be low, thanks to the mentioned uncertainty management. Please note that Random Variables are one step more ahead than intervals in uncertainty management. Random Variables can utilize their full capacity or a part of their potency, for example, by getting converted to confidence intervals or expected value or etc.

#### *Interval pTER*

The last defined future work of the interval pTER project is the interval pTER, itself. In this future work that is an in-parallel progressing paper of Mr. Kalamati (which is defined to be conducted after defending his master thesis) the output of the Sentiment Analysis phases of the studies of this project are delivered as input to the interval-based Uncertainty-handling (the abovementioned study) and considering that in both of the phases more uncertainties are handled, it is expected that the accuracy of this Sentiment Prediction system is increased in comparison with the scalar pTER research.

### 5.2.3 Sentiment Answering

One of the main motivations of this dissertation has been the application of Sentiment Prediction tools in Question Answering. Considering the Question Answering problem, some industries, especially the ones dealing with Recommender Systems field of study, are especially interested in detecting the “sentiment seeking” questions of users in the shopping platform websites or in Social Networks and answer the predicted version of their sentiments to them.

Such a system would be of very high interest to them, because when the hesitations of a customer about an intended purchase is resolved, the expected value of the daily purchases will increase, which is very welcome for companies. We call the act of detecting the Sentiment Questions (Somasundaran, 2007) of users and answering them as Sentiment Answering.

By Sentiment Questions, we mean the questions that are seeking for the quality of a product (that can be interpreted as: “what will be my opinion after using this product”). Detection of such questions can be done either by means of Machine Learning (Srba & Bielikova, 2016) tools or Semantic tools (Bogdanova, Santos, Barbosa, & Zadrozny, 2015). However, when such questions are recognized, it will be the turn of utilizing Question Answering.

Based on a philosophic point of view, Question Answering searches can be done either via Social Search or Library Search (A. Trias i Mansilla & de la Rosa i Esteva, 2013b). Focusing on the web applications, when an online user searches for an answer of his question, he can utilize online search engines (e.g. Google, Bing, Yahoo search, etc.) or he can ask his question in social networks or the other interaction platforms available in Web 2.0.

As illustrated, Sentiment Answering systems address the online management of Social Searches. However, this Social Search management can be done either centrally or peer2peer. In the following, we address each of them separately.

### *Central Sentiment Answering systems*

If the Sentiment Answering task is going to be done by a central system, then it can be considered as a supervised Social Search, in which a Big Brother takes care of the questions of others and serves them by answering their questions. What scalar and interval pTER studies propose best fit the mentioned central Sentiment Answering system. Such system can take advantage of a User/Item opinion matrix, partially filled in by Sentiment Analysis algorithms, and then, the Sentiment Prediction task can be well-covered, centrally, by means of CF techniques or the other mentioned techniques in section 1.1.1.

### *Peer2Peer Sentiment Answering Systems*

Supervised Question Answering has the cost/drawback of accepting the supervision of a Big Brother. However, despite the many advantages of such Big Brothers, there are recent trends to avoid them and keep their privacy, at a higher level (Power, 2016). Two of the most well-known Social-Search-based Question-Answering systems are "AskNext" (Albert Trias i Mansilla & de la Rosa i Esteva, 2012) and "Question Waves" (A. Trias i Mansilla & de la Rosa i Esteva, 2013a).

Question Waves algorithm defines an analogy between the iterations that a question receives in online or real-world Social Networks between the friends of the questioner, since the time it is propagated (asked) until the time it is returned back to him, likely with an answer, attached.

The Question Waves system is agent-based and therefore peer2peer. In the case that the agent of each online user has a User/Item opinion matrix, every agent can perform the task of Sentiment Answering, if it has enough knowledge about the asked Sentiment Question. The local User/Item opinion matrix of each agent, although due to its locality is more limited, it has two advantages. The first advantage, as mentioned, is removal of a Big Brother, and the second advantage is: Local knowledge of the owner (the corresponding user) of each agent potentially can be verified by him. In other words, an online user can verify or even explicitly charge the inner knowledge of his agent, only once, and thereafter, his agent can actively participate in the peer2peer Sentiment Answering task.

This peer2peer Question-Waves-based Sentiment-Prediction task is the subject of a new line or future research studies. It is necessary to note that even such a peer2peer Question-Waves-based Sentiment-Prediction system can take advantage of the entire Uncertainty-handling platform, proposed in this dissertation.

After fully addressing the planned (to be done by the already-involving students of this project) future works of this dissertation, it is the turn to address the other (unplanned) future works, which are discussed in the following subsection.

### 5.2.4 Other Future studies suggested to the society

In this subsection, we describe the future works of this study which are neither in-progress nor planned as an in-queue paper of any student of this project.

As a future work of the “WordNet Fuzzifier” study (section 3.1), we suggest that the WLD-fuzzifier algorithm proposed in this dissertation is used to fuzzify every other WLD in any language to increase the Text Mining efficiency in those languages.

As a future work of the WNPC (section 3.4), the main suggested future work is Weibull Bayes classifier. We, moreover, suggest utilizing 3-parameter Weibull distribution for even better accuracy improvement. Application of the proposed classifiers to the other Text Mining problems (e.g. Sentiment Analysis) and/or other Authorship Attribution problems (e.g. author verification) is another possible future work. Moreover, applying the same proposed algorithms on the other standard datasets for double confirming the results is suggested to the interested researchers for conducting future lines of this study. Another research line for the future study is proposing a classifier that can handle the uncertainty related to the smallness of the document itself. Please note that the Machine Learning approach to Authorship Attribution cannot handle the uncertainty when all of the (1) the document size, (2) the training documents of each author ($n/p$), and (3) the features number ($m$) are low. It is because, in such circumstances, there is almost no data by which the system is trained. Therefore, we would suggest a hybrid approach using the Cognitive Linguistics as well as Machine Learning techniques for the mentioned future work.

For the "GSW" study (sections 3.6 and 3.7), we suggest its real-time applications in Social Networks or Recommender Systems, similar to what in its offline (i.e. periodical usage) applications. We introduce Total Error to be considered as the cost function of $(y, z)$. Then, the Total Error is computed for a number of $(y, z)$ pairs, in non-linear time (training step). Next, the proposed MOA inputs the costs of a number of $(y, z)$ pairs and finds the approximate globally optimal $(y, z)$, in linear time (testing step). But, unlike the offline (periodical usage) applications, after this step we suggest that as soon as a new item or post is rated / liked / disliked by a user, the prediction error of Memory-based CF for that one prediction can heuristically update the already existing Total Errors, in linear time (real-time-training step). Then, based on the updated train data, the MOA outputs an updated optimal $(y, z)$, again in linear time (real-time-testing step). Thus, after training the system with its initial data, once, all of the next real-time training and testing steps have linear time complexity. Figure 5.1 provides the block-diagram of the proposed real-

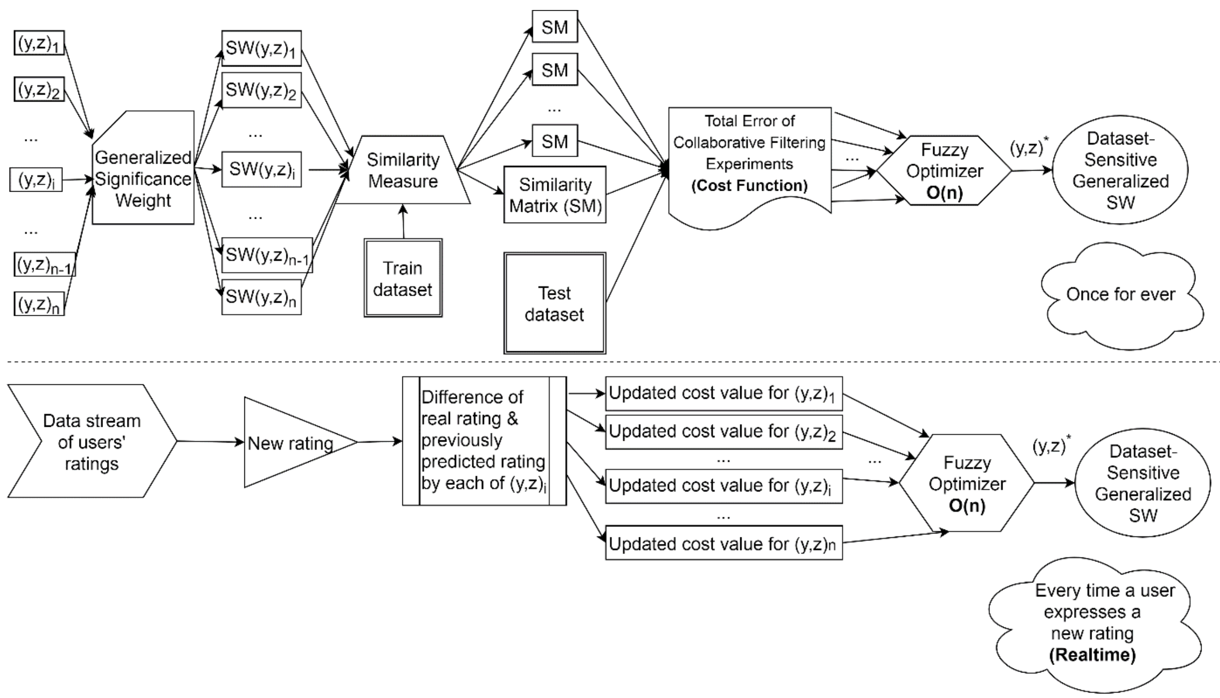



**Figure 5.1. The block diagram of the method described to be implemented as Future Work (this method is already implementable by the proposed theory in sections 3.6 and 3.7).**

time system.

As another future study of sections 3.6 and 3.7, we suggest taking advantage of other datasets and other similarity measures to prove the efficiency of the presented method, more strongly.

# 6 Bibliography


Aarts, E., Korst, J., & Michiels, W. (2014). Simulated Annealing. In *Search Methodologies* (pp. 265–285). Springer US.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, *17*(6), 734–749.

Agirre, E., & Soroa, A. (2009). Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the ACL* (pp. 33–41).

Agirre E., Lopez de Lacalle O., S. A. (2014). Random Walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics, 40:1*, *40*(1).

Ahn, H. J. (2008). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences*, *178*(1), 37–51.

Airoldi, E. M., Anderson, A. G., Fienberg, S. E., & Skinner, K. K. (2006). Who wrote ronald Reagan's radio addresses? *Bayesian Analysis*, *1*(2), 289–320.

Akbarzadeh-T, M. R., Davarynejad, M., & Pariz, N. (2008). Adaptive fuzzy fitness granulation for evolutionary optimization. *International Journal of Approximate Reasoning*, *49*(3), 523–538.

Alahmadi, D. H., & Zeng, X. J. (2015). ISTS: Implicit social trust and sentiment based approach to recommender systems. *Expert Systems with Applications*, *42*(22), 8840–8849.

Alahmadi, D. H., & Zeng, X. J. (2016). Twitter-based recommender system to address cold-start: A genetic algorithm based trust modelling and probabilistic sentiment analysis. *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, *2016–Janua*, 1045–1052.

Ali, F., Kwak, K. K.-S., & Kim, Y.-G. Y. (2016). Opinion mining based on fuzzy domain ontology and Support Vector Machine : A proposal to automate online review classification. In *Applied Soft Computing Journal* (Vol. 47, pp. 235–250).

Ali Ghazanfar, M., Prugel-Bennett, A., Ghazanfar, M., & Prugel-Bennett, A. (2010). Novel Significance Weighting Schemes for Collaborative Filtering: Generating Improved Recommendations in Sparse Environments. In *DMIN'10, the 2010 International Conference on Data Mining*.

Altheneyan, A. S., & Menai, M. E. B. (2014). Naïve Bayes classifiers for authorship attribution of Arabic texts. *Journal of King Saud University - Computer and Information Sciences*, *26*(4), 473–484.

Aslantürk, O., Sezer, E. A., Sever, H., & Raghavan, V. (2010). Application of cascading rough set-based classifiers on authorship attribution. In *Proceedings - 2010 IEEE International Conference on Granular Computing, GrC 2010* (pp. 656–660).

Ayogu, I. I., & Olutayo, V. A. (2016). Authorship Attribution using Rough Sets based Feature Selection Techniques. *International Journal of Computer Applications*, *152*(6), 975–8887.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)* (Vol. 0, pp. 2200–2204).

Bagchi, S. (2015). Performance and quality assessment of similarity measures in collaborative filtering using mahout. *Procedia Computer Science*, *50*, 229–234.

Bartz-Beielstein, T. (2016). A Survey of Model-based Methods for Global Optimization. *Bioinspired Optimization Methods and Their Applications*, 1–18.

Basile, P., Degemmis, M., Gentile, A. L., Lops, P., & Semeraro, G. (2007). The JIGSAW algorithm for word sense disambiguation and semantic indexing of documents. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 4733 LNAI, pp. 314–325).

Bimson, K. D., Hull, R. D., & Nieten, D. (2016). The Lexical Bridge: A Methodology for Bridging the Semantic Gaps between a Natural Language and an Ontology. In *Semantic Web* (pp. 137–151).

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, *46*, 109–132.

Bogdanova, D., Santos, C., Barbosa, L., & Zadrozny, B. (2015). Detecting Semantically Equivalent Questions in Online User Forums. *Proceedings of the 19th Conference on Computational Language Learning*, 123–131.

Bond, F., & Paik, K. (2012). A Survey of WordNets and their Licenses. In *Proceedings of the 6th Global WordNet Conference (GWC 2012)* (pp. 64–71).

Borin, L. (2005). Mannen är faderns mormor: Svenskt associationslexikon reinkarnerat. *LexicoNordica*, *12*, 39–54.

Borin, L., & Forsberg, M. (2009). All in the family: A comparison of SALDO and WordNet. In *Proceedings of the Nodalida 2009 Workshop on WordNets and other Lexical Semantic Resources - between Lexical Semantics, Lexicography, Terminology and Formal Ontologies. NEALT Proceedings Series*.

Borin, L., & Forsberg, M. (2010a). Beyond the synset: Swesaurus–a fuzzy Swedish wordnet. In *Proceedings of the symposium: Re-thinking synonymy: semantic sameness and similarity in languages and their description*.

Borin, L., & Forsberg, M. (2010b). From the people's synonym dictionary to fuzzy synsets-first steps. In *Proc. LREC 2010 workshop Semantic relations. Theory and Applications.*

Bounhas, M., Ghasemi Hamed, M., Prade, H., Serrurier, M., & Mellouli, K. (2014). Naive possibilistic classifiers for imprecise or uncertain numerical data. *Fuzzy Sets and Systems*, *239*, 137–156.

Boussaïd, I., Lepagnot, J., Siarry, P., BoussaïD, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*, *237*, 82–117.

Boyd, S., & Vandenberghe, L. (2010). *Convex optimization*. *Optimization Methods and Software* (Vol. 25).

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th conference on Uncertainty in Artificial Intelligence* (Vol. 461, pp. 43–52).

Brocardo, M. L., Traore, I., Saad, S., & Woungang, I. (2013). Authorship verification for short messages using stylometry. In *2013 International Conference on Computer, Information and Telecommunication Systems, CITS 2013*.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modelling and User-Adapted Interaction*, *12*(4), 331–370.

Cambria, E. (2016). Affective Computing and Sentiment Analysis. *IEEE Intelligent Systems*, *31*(2), 102–107.

Cambria, E., & Hussain, A. (2012). *Sentic Computing - Techniques, Tools, and Applications*. Dordrecht,: Springer Science & Business Media.

Cambria, E., & White, B. (2014). Jumping NLP curves: A review of natural language processing research. *IEEE Computational Intelligence Magazine*, *9*(2), 48–57.

Campbell, J. (2011). Why is 30 the "Magic Number" for Sample Size? *Food Science and Technology. United Kingdom*.

Candillier, L., Meyer, F., & Fessant, F. (2008). Designing specific weighted similarity measures to improve collaborative filtering systems. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 5077 LNAI, pp. 242–255).

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, *7*(3), 1247–1250.

Chakraborty, D., Guha, D., & Dutta, B. (2016). Multi-objective optimization problem under fuzzy rule constraints using particle swarm optimization. *Soft Computing*, *20*(6), 2245–2259.

Chaski, C. E. (2013). Best practices and admissibility of forensic author identification. *Journal of Law and Policy*, *21*(1993), 333–376.

Chen, C. C., Wan, Y.-H., Chung, M.-C., & Sun, Y.-C. (2013). An effective recommendation method for cold start new users using trust and distrust networks. *Information Sciences*, *224*, 19–36.

Chen, L., Chen, G., & Wang, F. (2015). Recommender systems based on user reviews: the state of the art. *User Modeling and User - Adapted Interaction*, *25*(2), 99–154.

Chen, L., & Wang, F. (2013). Preference-based clustering reviews for augmenting e-commerce recommendation. *Knowledge-Based Systems*, *50*, 44–59.

Chen, L., & Wang, F. (2017). Explaining Recommendations Based on Feature Sentiments in Product Reviews. In *Proceedings of the 22nd International Conference on Intelligent User*

*Interfaces - IUI '17* (pp. 17–28).

Chen, X., Qin, Z., Zhang, Y., & Xu, T. (2016). Learning to Rank Features for Recommendation over Multiple Categories. *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '16*, 305–314.

Chen, X., Zheng, Z., Yu, Q., & Lyu, M. R. (2014). Web service recommendation via exploiting Location and QoS information. *IEEE Transactions on Parallel and Distributed Systems*, *25*(7), 1913–1924.

Choi, K., & Suh, Y. (2013). A new similarity function for selecting neighbors for each target item in collaborative filtering. *Knowledge-Based Systems*, *37*, 146–153.

Consoli, S., & Reforgiato Recupero, D. (2015). Using fred for named entity resolution, linking and typing for knowledge base population. In *Semantic Web Evaluation Challenge (SemWebEval 2014 at ESWC 2014)* (pp. 40–50).

Crawford, E., Gingerich, M., & Eliasmith, C. (2015). Biologically Plausible, Human-scale Knowledge Representation. *Cognitive Science*.

Dabeer, O. (2012). Estimating and Shaping Opinions in Twitter. *2012 Information Theory and Applications Workshop*.

Dantzig, G. B. (2016). *Linear Programming and Extensions*. *Princeton university*. Princeton university press.

Das, P., Tasmim, R., & Ismail, S. (2016). An experimental study of stylometry in Bangla literature. In *2nd International Conference on Electrical Information and Communication Technologies, EICT 2015* (pp. 575–580).

de Gemmis, M., Lops, P., Semeraro, G., & Basile, P. (2008). Integrating tags in a semantic content-based recommender. In *Proceedings of the 2008 ACM conference on Recommender systems - RecSys '08* (p. 163).

de Melo, G., Baker, C. F., Ide, N., Passonneau, R. J., & Fellbaum, C. (2012). Empirical Comparisons of MASC Word Sense Annotations. In *Lrec 2012 - Eighth International Conference on Language Resources and Evaluation* (pp. 3036–3043).

Delgado, M., & Moral, S. (1987). On the concept of possibility-probability consistency. *Fuzzy Sets and Systems*, *21*(3), 311–318.

Diao, Q., Qiu, M., Wu, C.-Y., Smola, A. J., Jiang, J., & Wang, C. (2014). Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '14*, 193–202.

Diederich, J., Kindermann, J., Leopold, E., Paass, G., Informationstechnik, G. F., & Augustin, D.-S. (2000). Authorship Attribution with Support Vector Machines. *Applied Intelligence*, *19*, 2003.

Ding-Zhu, D., & Pardalos, P. M. (2013). *Handbook of Combinatorial Optimization*. Springer Science & Business Media.

Dong, R., O’Mahony, M. P., Schaal, M., McCarthy, K., & Smyth, B. (2013). Sentimental product recommendation. *Proceedings of the 7th ACM Conference on Recommender Systems - RecSys ’13*, 411–414.

Dong, R., O’Mahony, M. P., Schaal, M., McCarthy, K., & Smyth, B. (2016). Combining similarity and sentiment in opinion mining for product recommendation. *Journal of Intelligent Information Systems*, *46*(2), 285–312.

Dostál, Z. (2009). *Optimal quadratic programming algorithms. With applications to variational inequalities*.

Draicchio, F., Gangemi, A., Presutti, V., & Nuzzolese, A. G. (2013). FRED: From natural language text to RDF and OWL in one click. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 7955 LNCS, pp. 263–267).

Dubois, D. J., & Prade, H. (1980). *Fuzzy Sets and Systems: Theory and Applications*.

Dubois, D., & Prade, H. (1982). On several representations of an uncertain body of evidence. In *Fuzzy information and decision processes* (pp. 167–181).

Dubois, D., & Prade, H. (1983). Unfair coins and necessity measures: Towards a possibilistic interpretation of histograms. *Fuzzy Sets and Systems*, *10*(1–3), 15–20.

Dubois, D., Prade, H., & Sandri, S. (1993). On possibility/probability transformations. *Proceedings of Fourth IFSA Conference*, 103–112.

Eder, M. (2015). Does size matter? Authorship attribution, small samples, big problem. *Digital Scholarship in the Humanities*, *30*(2), 167–182.

El Manar El Bouanani, S., & Kassou, I. (2014). Authorship Analysis Studies : A Survey. *International Journal of Computer Applications*, *86*(12), 22–29.

Elming, Jakob and Johannsen, Anders and Klerke, Sigrid and Lapponi, Emanuele and Alonso, Hector Martinez and Sogaard, A. (2013). Down-stream effects of tree-to-dependency conversions. In *Hlt-Naacl* (pp. 617–626).

Erk, K. (2010). What Is Word Meaning, Really? (And How Can Distributional Models Help Us Describe It?). In *In Proceedings of the 2010 Workshop on GEometrical Models of Natural Language Semantics, ACL 2010, Uppsala, Sweden, 16 July 2010* (pp. 17–26).

Erk, K., & McCarthy, D. (2009). Graded word sense assignment. In *EMNLP ’09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (Vol. 1, pp. 440–449).

Erk, K., McCarthy, D., & Gaylord, N. (2009). Investigations on word senses and word usages. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP* (Vol. 1, pp. 10–18).

Erk, K., McCarthy, D., & Gaylord, N. (2013). Measuring word meaning in context. *Computational Linguistics*, *39*(3), 511–554.

Evans, J. S. B. T., & Stanovich, K. E. (2013). Dual-Process Theories of Higher Cognition: Advancing the Debate. *Perspectives on Psychological Science*, *8*(3), 223–241.

Faridani, S. (2011). Using canonical correlation analysis for generalized sentiment analysis, product recommendation and search. *Proc. RecSys*, 355–358.

Fellbaum, C. (1998). WordNet: An Electronic Lexical Database. *British Journal Of Hospital Medicine London England 2005*, *71*(3), 423.

Feltoni Gurini, D., Gasparetti, F., Micarelli, A., & Sansonetti, G. (2018). Temporal people-to-people recommendation on social networks with sentiment-based matrix factorization. *Future Generation Computer Systems*, *78*, 430–439.

Fillmore, C. J., Ide, N., Jurafsky, D., & Macleod, C. (1998). An American National Corpus: A Proposal. In *the First International Language Resources and Evaluation Conference* (pp. 965–970).

Fridman, L., Stolerman, A., Acharya, S., Brennan, P., Juola, P., Greenstadt, R., … Gomez, F. (2015). Multi-modal decision fusion for continuous authentication. *Computers and Electrical Engineering*, *41*(C), 142–156.

Galitsky, B. (2017). Improving relevance in a content pipeline via syntactic generalization. *Engineering Applications of Artificial Intelligence*, *58*, 1–26.

Galitsky, B., & McKenna, E. (2008). Sentiment extraction from consumer reviews for providing product recommendations. *US Patent App. 12/119,465*.

Gangemi, A. (2013). A comparison of knowledge extraction tools for the semantic web. In *The Semantic Web: Semantics and Big Data: 10th International Conference, ESWC 2013 (Lecture Notes in Computer Science)* (Vol. 7882 LNCS, pp. 351–366).

Gangemi, A., Draicchio, F., Presutti, V., Nuzzolese, A. G., Reforgiato, D., & Nazionale, S. C. (2013). A Machine Reader for the Semantic Web. In *Proceedings of the 2013th International Conference on Posters & Demonstrations Track* (pp. 149–152).

Gangemi, A., Nuzzolese, A., Presutti, V., Draicchio, F., Musetti, A., & Ciancarini, P. (2012). Automatic Typing of DBpedia Entities. In *The Semantic Web – ISWC 2012 SE - 5* (Vol. 7649, pp. 65–81).

Gangemi, A., Presutti, V., & Reforgiato Recupero, D. (2014). Frame-based detection of opinion holders and topics: A model and a tool. *IEEE Computational Intelligence Magazine*, *9*(1), 20–30.

Gangemi, A., Presutti, V., Reforgiato Recupero, D., Nuzzolese, A. G., Draicchio, F., & Mongiovì, M. (2017). Semantic Web Machine Reading with FRED. *Semantic Web*, *8*(6), 873–893.

García-Cumbreras, M. Á., Montejo-Ráez, A., & Díaz-Galiano, M. C. (2013). Pessimists and optimists: Improving collaborative filtering through sentiment analysis. *Expert Systems with Applications*, *40*(17), 6758–6765.

Garnaev, A., Baykal-Gursoy, M., & Poor, H. V. (2014). Incorporating attack-type uncertainty into network protection. *IEEE Transactions on Information Forensics and Security*, *9*(8), 1278–

1287.

Garnaev, A., & Trappe, W. (2016). A bandwidth monitoring strategy under uncertainty of the adversary's activity. *IEEE Transactions on Information Forensics and Security*, *11*(4), 837–849.

Gavrilova, M. L., & Yampolskiy, R. (2011). Applying biometric principles to avatar recognition. *Lecture Notes in Computer Science*, *6670*, 140–158.

Glover, F., & Laguna, M. (2013). Tabu Search. In *Handbook of combinatorial optimization* (pp. 3261–3362). Springer New York.

Goikoetxea, J. and E. A. and A. S. (2015). Random Walks and Neural Network Language Models on Knowledge Bases. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1434–1439).

Gonçalo Oliveira, H., & Gomes, P. (2011). Automatic Discovery of Fuzzy Synsets from Dictionary Definitions. In *22nd International Joint Conference on Artificial Intelligence* (pp. 1801–1806).

Grieve, J. (2007). Quantitative authorship attribution: An evaluation of techniques. *Literary and Linguistic Computing*, *22*(3), 251–270.

Gunawardana, A., & Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research*, *10*, 2935–2962.

Gurini, D. F., Gasparetti, F., Micarelli, A., & Sansonetti, G. (2013). A sentiment-based approach to twitter user recommendation. In *CEUR Workshop Proceedings* (Vol. 1066).

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. (2009). The WEKA data mining software: An update. In *SIGKDD Explorations* (Vol. 11, p. f).

Halvani, O., Steinebach, M., & Zimmermann, R. (2013). Authorship verification via k-nearest neighbor estimation. *Notebook PAN at CLEF*.

Hansen, P., & Nenad Mladenović. (2014). Variable neighborhood search. In *Search methodologies* (pp. 313–337). Springer US.

Harper, F. M., & Konstan, J. A. (2015). The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.*, *5*(4), 19:1--19:19.

Heo, J. H., Salas, J. D., & Kim, K. D. (2001). Estimation of confidence intervals of quantiles for the Weibull distribution. *Stochastic Environmental Research and Risk Assessment*, *15*(4), 284–309.

Herlocker, J., & Konstan, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (p. 8). ACM.

Herlocker, J., Konstan, J. a., & Riedl, J. (2002). An Empirical Analysis of Design Choices in Neighborhood-Based Collaborative Filtering Algorithms. *Information Retrieval*, *5*, 287–310.

Hirst, G. (2009). Ontology and the Lexicon. In S. Staab & R. Studer (Eds.), *Ontology and the*

*Lexicon* (pp. 269–292). Springer Berlin Heidelberg.

Homem, N., & Carvalho, J. P. (2011). Authorship identification and author fuzzy "fingerprints." In *Annual Conference of the North American Fuzzy Information Processing Society - NAFIPS*.

Hossayni, S.-A., Akbarzadeh-T, M. R., Reforgiato Recupero, D., Gangemi, A., & de la Rosa i Esteva, J. L. (2016). Fuzzy Synsets, and Lexicon-Based Sentiment Analysis. In D. M. Deng Y., Denecke K., Declerck T., Recupero D.R. (Ed.), *EMSA-RMed* (Vol. 1613). CEUR Workshop Proceedings (Scopus code 21100218356).

Hossayni, S.-A., Rajati, M.-R., Del Acebo, E., Reforgiato Recupero, D., Gangemi, A., Akbarzadeh-T, M.-R., & De La Rosa I Esteva, J. L. (2016). Towards Interval Version of Fuzzy Synsets. *Frontiers in Artificial Intelligence and Applications*, *288*, 297–302.

Hurford, J. (2003). Why Synonymy is Rare: Fitness is in the Speaker. *Ecal03*.

Hyndman, R. R. J., & Koehler, A. A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679–688.

Ivasic-Kos, M., Pobar, M., & Ribaric, S. (2016). Two-tier image annotation model based on a multi-label classifier and fuzzy-knowledge representation scheme. *Pattern Recognition*, *52*, 287–305.

Jakob, N., & Weber, S. (2009). Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion* (pp. 57–64).

Jamali, M., & Ester, M. (2009). TrustWalker: a random walk model for combining trust-based and item-based recommendation. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 397–406.

Jockers, M. L., & Witten, D. M. (2010). A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing*, *25*(2), 215–223.

John Ross Quinlan. (2014). *C4.5 : programs for machine learning*. *Morgan Kaufmann series in machine learning*. Morgan Kaufmann Publishers Inc.

Johnson, R., & Zhang, T. (2016). Supervised and Semi-Supervised Text Categorization using LSTM for Region Embeddings. In *ICML*.

Jun Zhang, J., Zhi-hui Zhan, N., Ying Lin, H. S. H., Ni Chen, Y., Yue-jiao Gong, Y., Jing-hui Zhong, Y., … Yu-hui Shi, J. (2011). Evolutionary Computation Meets Machine Learning: A Survey. *IEEE Computational Intelligence Magazine*, *6*(4), 68–75.

Jung, K. Y., Park, D. H., & Lee, J. H. (2004). Hybrid collaborative filtering and content-based filtering for improved recommender system. *Computational Science Iccs 2004 Pt 1 Proceedings*, *3036*, 295–302.

Juola, P. (2008). Authorship Attribution. *Foundations and Trends® in Information Retrieval*, *1*(3), 233–334.

Kalloubi, F., Nfaoui, E. H., & El Beqqali, O. (2016). Microblog semantic context retrieval system based on linked open data and graph-based theory. *Expert Systems with Applications*, *53*, 138–148.

Kann, V., & Rosell, M. (2005). Free construction of a free Swedish dictionary of synonyms. In *Proc. 15th Nordic Conf. on Comp. Ling.–NODALIDA (5)* (pp. 1–6).

Kar, S. S., & Ramalingam, A. (2013). Is 30 the Magic Number ? Issues in Sample Size. *National Journal of Commuity Medicine*, *4*(1), 175–179.

Kawamae, N. (2011). Predicting future reviews: sentiment analysis models for collaborative filtering. *Proceedings of the Fourth ACM International …*, 605–614.

Kibriya, A. M., Frank, E., Pfahringer, B., & Holmes, G. (2004). Multinomial Naive Bayes for Text Categorization Revisited. *Australasian Joint Conference on Artificial Intelligence*, 488–499.

Kilgarriff, A. (1997). "I don't believe in word senses." *Computers and the Humanities*, *31*(2), 25.

Kim, H. K., Kim, H., & Cho, S. (2017). Bag-of-concepts: Comprehending document representation through clustering words in distributed representation. *Neurocomputing*, *266*, 336–352.

Kim, J., Yoo, J., Lim, H., Qiu, H., Kozareva, Z., & Galstyan, A. (2013). Sentiment Prediction Using Collaborative Filtering. In *Icwsm* (pp. 685–688).

Kim, S.-B., Seo, H.-C., & Rim, H.-C. (2003). Poisson Naive Bayes for Text Classification with Feature Weighting. In *Proceedings of the sixth international workshop on Information retrieval with Asian languages* (pp. 33–40).

Kim, S. B., Han, K. S., Rim, H. C., & Myaeng, S. H. (2006). Some Effective Techniques for Naive Bayes Text Classification. *IEEE Transactions on Knowledge and Data Engineering*, *18*(11), 1457–1466.

Kim, S., Yoon, J., & Song, M. (2001). Automatic Extraction of Collocations From Korean Text. *Computers and the Humanities*, *35*, 273–297.

Kiryakov, A. K., & Simov, K. I. (1999). Ontologically supported semantic matching. *Proceedings of NoDaLiDa 1999*, 91–102.

Kobel, A., Rouillier, F., & Sagraloff, M. (2016). Computing Real Roots of Real Polynomials . . . and now For Real ! In *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation* (pp. 303–310).

Koren, Y. (2010). Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *4*(1).

Koukourikos, A., Stoitsis, G., & Karampiperis, P. (2012). Sentiment analysis: A tool for rating attribution to content in recommender systems. *CEUR Workshop Proceedings*, *896*, 61–70.

Krishna, P. V., Misra, S., Joshi, D., & Obaidat, M. S. (2013). Learning Automata Based Sentiment Analysis for recommender system on cloud. *2013 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 1–5.

Kwon, H.-J., & Hong, K.-S. (2009). Moment Similarity of Random Variables to Solve Cold-start

Problems in Collaborative Filtering. *2009 Third International Symposium on Intelligent Information Technology Application*, 584–587.

Kwon, H., Kwon, H., & Hong, K. (2011). Personalized Emotional Prediction Method for Real-Life. *Lecture Notes in Computer Science*, *6781*, 45–52.

León Aráuz, Pilar Gómez-Romero, J., & Bobillo, F. (2012). A Fuzzy Ontology Extension of WordNet and EuroWordnet for Specialized Knowledge. In *Proceedings of the 10th terminology and knowledge engineering conference* (pp. 139–154). Madrid.

Leung, C. W., Chan, S. C., & Chung, F. (2006). Integrating Collaborative Filtering and Sentiment Analysis : A Rating Inference Approach. *ECAI 2006 Workshop on Recommender Systems*, 62–66.

Leung, C. W. K., Chan, S. C. F., Chung, F. L., & Ngai, G. (2011). A probabilistic rating inference framework for mining user preferences from reviews. *World Wide Web*, *14*(2), 187–215.

Levi, A., Mokryn, O. O., Diot, C., & Taft, N. (2012). Finding a Needle in a Haystack of Reviews : Cold Start Context-Based Hotel Recommender System. *Proceedings of the 6th ACM Conference on Recommender Systems - RecSys '12*, 115–122.

Lewis, J. A. (2014). *Forensic Document Examination: Fundamentals and Current Trends*. *Forensic Document Examination: Fundamentals and Current Trends*.

Lewis, M., Lee, K., & Zettlemoyer, L. (2016). LSTM CCG Parsing. In *Proceedings of NAACL-HLT* (pp. 221–231).

Li, H., Cui, J., Shen, B., & Ma, J. (2016). An intelligent movie recommendation system through group-level sentiment analysis in microblogs. *Neurocomputing*, *210*, 164–173.

Li, T., Mei, T., Kweon, I. S., & Hua, X. S. (2011). Contextual bag-of-words for visual categorization. *IEEE Transactions on Circuits and Systems for Video Technology*, *21*(4), 381–392.

Lin, C., Xie, R., Guan, X., Li, L., & Li, T. (2014). Personalized news recommendation via implicit social experts. *Information Sciences*, *254*, 1–18.

Lin, F., & Sandkuhl, K. (2008). A survey of exploiting WordNet in ontology matching. *IFIP International Federation for Information Processing*, *276*, 341–350.

Liu, H., He, J., Wang, T., Song, W., & Du, X. (2013). Combining user preferences and user opinions for accurate recommendation. *Electronic Commerce Research and Applications*, *12*(1), 14–23.

Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2014). A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, *56*, 156–166.

Lops, P., Degemmis, M., & Semeraro, G. (2007). Improving Social Filtering Techniques Through WordNet-Based User Profiles. In *User Modeling* (pp. 268–277).

Lops, P., Musto, C., Narducci, F., De Gemmis, M., Basile, P., & Semeraro, G. (2010). MARS: a MultilAnguage Recommender System Pasquale. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems* (pp. 24–31).

Lou, P., Zhao, G., Qian, X., Wang, H., & Hou, X. (2016). Schedule a rich sentimental travel via sentimental poi mining and recommendation. *Proceedings - 2016 IEEE 2nd International Conference on Multimedia Big Data, BigMM 2016*, 33–40.

Lourenço, H. R., O., M., & T., S. (2010). Iterated Local Search: Framework and Applications. In *Handbook of Metaheuristics* (pp. 363–397).

Lu, J., Wang, G., & Moulin, P. (2014). Human identity and gender recognition from gait sequences with arbitrary walking directions. *IEEE Transactions on Information Forensics and Security*, *9*(1), 51–61.

Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: A survey. *Decision Support Systems, 74*, 12–32.

Luyckx, K. (2011). *Scalability Issues in Authorship Attribution*. Brussel: ASP - Academic & Scientific Publishers.

Luyckx, K., & Daelemans, W. (2008). Authorship attribution and verification with many authors and limited data. In *Belgian/Netherlands Artificial Intelligence Conference* (pp. 335–336).

Ma, H., King, I., & Lyu, M. R. (2007). Effective Missing Data Prediction for Collaborative Filtering. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 39–46).

Maas, A., Daly, R., Pham, P., & Huang, D. (2011). Learning word vectors for sentiment analysis. *Proceedings of the 49th …*.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (pp. 142–150).

Madalli, D., Sulochana, A., & Singh, A. K. (2016). COMAT: core ontology of matter. *Program*, *50*(1), 103–117.

Maltz, D. A. (1994). *Distributing Information for Collaborative Filtering on Usenet Net News*. Cambridge.

Maltz, D., & Ehrlich, K. (1995). Pointing the way: active collaborative filtering. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 202–209).

Manjula, D., Aghila, G., & Geetha, T. V. V. (2003). Document knowledge representation using description logics for information extraction and querying. In *Proceedings ITCC 2003, International Conference on Information Technology: Computers and Communications* (pp. 189–193).

Martinez, D., Otegi, A., Soroa, A., & Agirre, E. (2014). Improving search over Electronic Health Records using UMLS-based query expansion through random walks. *Journal of Biomedical Informatics*, *51*, 100–106.

McCann, B., Bradbury, J., Xiong, C., & Socher, R. (2017). Learned in Translation: Contextualized Word Vectors. In *Advances in Neural Information Processing Systems*.

McLaughlin, M. R., & Herlocker, J. L. (2004). A Collaborative Filtering Algorithm and Evaluation

Metric that Accurately Model the User Experience. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 329–336). ACM.

Meghdadi, A. H., & Akbarzadeh-T, M.-R. (2001). Probabilistic Fuzzy Logic and Probabilistic Fuzzy Systems. In *The 10th IEEE International Conference on Fuzzy Systems* (pp. 1127–1130).

Mendel, J. M. (1999). Computing With Words, When Words Can Mean Different things to Different People. *Proceedings of the 3rd International ICSC Symposium on Fuzzy Logic and Applications, Rochester, NY*.

Mendel, J. M. (2007). Computing with words and its relationships with fuzzistics. *Information Sciences*, *177*(4), 988–1006.

Mendel, J. M., & Wu, D. (2010). Interval type-2 fuzzy sets. In *Perceptual Computing* (pp. 35–63). Wiley online library.

Merkle, D., & Middendorf, M. (2014). Swarm intelligence. In *Search Methodologies* (pp. 213–242).

Meyniel, F., Sigman, M., Mainen, Z. F. Z. F. F., Acerbi, L., Vijayakumar, S., Wolpert, D. M., … Sigman, M. (2015). Confidence as Bayesian Probability: From Neural Origins to Behavior. *Neuron*, *88*(1), 78–92.

Mihalcea, R. (2011). Word sense disambiguation. In *Encyclopedia of machine learning* (pp. 1027–1030).

Miller, G. a. (1995). WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39–41.

Miller, G. a., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography*, *3*(4), 235–244.

Montaner, M., López, B., Rosa, J. L. de la, & De La Rosa, J. L. (2003). A taxonomy of recommender agents on the internet. *Artificial Intelligence Review*, *19*(4), 285–330.

Mourão, F., Rocha, L., Araújo, C., Meira, W., & Konstan, J. (2017). What surprises does your past have for you? *Information Systems*, *71*, 137–151.

Muhammad, K., Lawlor, A., Rafter, R., & Smyth, B. (2015). Great explanations: Opinionated explanations for recommendations. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9343*, 244–258.

Munkres, J. (1999). *Topology, ed*. MIT press.

Murphy, K. P. (2006). Binomial and multinomial distributions. *Bernoulli*, (5), 1–16.

Navigli, R. (2006). Meaningful Clustering of Senses Helps Boost Word Sense Disambiguation Performance. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics joint with the 21st International Conference on Computational Linguistics* (pp. 105–112).

Neal, T., Sundararajan, K., Fatima, A., Yan, Y., Xiang, Y., & Woodard, D. (2017). Surveying Stylometry Techniques and Applications. *ACM Computing Surveys*, *50*(6), 1–36.

Nguyen, A.-T., Denos, N., & Berrut, C. (2007). Improving new user recommendations with rule-based induction on cold user data. *Proceedings of the 2007 ACM Conference on Recommender Systems - RecSys '07*, 121–128.

Nwobi, F. N., & Ugomma, C. A. (2014). A Comparison of Methods for the Estimation of Weibull Distribution Parameters. *Metodološki Zvezki - Advances in Methodology and Statistics*, *11*(1), 65–78.

Osgood, C. E. (1952). the Nature and Measurement of Meaning. *Psychological Bulletin*, *49*(3), 227.

Otegi, A., Arregi, X., Ansa, O., & Agirre, E. (2014). Using knowledge-based relatedness for information retrieval. *Knowledge and Information Systems*, *44*(3), 689–718.

Otegi, A., Xavier, A., & Eneko, A. (2011). Query Expansion for IR using Knowledge-Based Relatedness. In *Proceedings of 5th International Joint Conference on Natural Language Processing* (pp. 1467–1471).

Palomino-Garibay, A., Camacho-González, A. T., Fierro-Villaneda, R. A., Hernández-Farias, I., Buscaldi, D., & Meza-Ruiz, I. V. (2015). A random forest approach for authorship profiling. In *CEUR Workshop Proceedings* (Vol. 1391).

Papadimitriou, A., Symeonidis, P., & Manolopoulos, Y. (2012). A generalized taxonomy of explanations styles for traditional and social recommender systems. *Data Mining and Knowledge Discovery*, *24*(3), 555–583.

Pappas, N., & Popescu-Belis, A. (2013). Sentiment analysis of user comments for one-class collaborative filtering over ted talks. *[SIGIR2013]Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 773.

Pappas, N., & Popescu-Belis, A. (2016). Adaptive sentiment-aware one-class collaborative filtering. *Expert Systems with Applications*, *43*, 23–41.

Parand, K., Hossayni, S., & Rad, J. (2016). An operation matrix method based on Bernstein polynomials for Riccati differential equation and Volterra population model. *Applied Mathematical Modelling*, *40*(2), 993–1011.

Park, S.-T., Pennock, D., Madani, O., Good, N., & DeCoste, D. (2006). Naïve filterbots for robust cold-start recommendations. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '06*, 699–705.

Patil, T. R. (2013). Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification. *International Journal Of Computer Science And Applications*, *6*(2), 256–261.

Pennock, D. M., Lawrence, S., & Giles, C. L. (2000). Collaborative Filtering by Personality Diagnosis : A Hybrid Memory- and Model-Based Approach. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence* (pp. 473–480).

Pirasteh, P., Hwang, D., & Jung, J. E. (2014). Weighted Similarity Schemes for High Scalability in User-Based Collaborative Filtering. *Mobile Networks and Applications*, 1–11.

Platt, J. C. (1998). Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. *Advances in Kernel Methods*, 185–208.

Poria, S., Cambria, E., Gelbukh, A., Bisio, F., & Hussain, A. (2015). Sentiment Data Flow Analysis by Means of Dynamic Linguistic Patterns. *IEEE Computational Intelligence Magazine*, *10*(4), 26–36.

Pouramini, J., & Minaei-Bidgoli, B. (2016). A New Synthetic Oversampling Method Using Ontology and Feature Selection in Order to Improve Imbalanced Textual Data Classification in Persian Texts. *Bulletin de La Société Royale Des Sciences de Liège*, *85*, 358–375.

Power, D. J. (2016). “Big Brother” can watch us. *Journal of Decision Systems*, *25*, 578–588.

Presutti, V., Draicchio, F., & Gangemi, A. (2012). Knowledge extraction based on discourse representation theory and linguistic frames. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 7603 LNAI, pp. 114–129).

Raghavan, S., Kovashka, A., & Mooney, R. (2010). Authorship Attribution Using Probabilistic Context-Free Grammars. In *Proceedings of the ACL 2010 Conference Short Papers* (pp. 38–42).

Reforgiato Recupero, D., Presutti, V., Consoli, S., Gangemi, A., Nuzzolese, A. G., Reforgiato Recupero, D., … Nuzzolese, A. G. (2015). Sentilo: Frame-Based Sentiment Analysis. *Cognitive Computation*, *7*(2), 211–225.

Reforgiato Recupero, D., Presutti, V., Consoli, S., Gangemi, A., & Nuzzolese, A. G. (2015). Sentilo: Frame-Based Sentiment Analysis. *Cognitive Computation*, *7*(2), 211–225.

Ren, F., & Wu, Y. (2013). Predicting user-topic opinions in twitter with social and topical context. *IEEE Transactions on Affective Computing*, *4*(4), 412–424.

Rennie, J. D. M., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML)-2003* (Vol. 20, pp. 616–623).

Resende, M. G. C., & Ribeiro, C. C. (2010). Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations, and Applications. In *Handbook of Metaheuristics* (Vol. 146, pp. 283–319).

Resnick, P., & Varian, H. R. (1997). Recommender Systems. *Communications of the ACM*, *40*(3), 56–58.

Rhodes, D. (2015). Author Attribution with CNN’s. In *CS224N Projects* (pp. 1–8).

Ricci, F., Rokach, L., & Shapira, B. (2004). *Introduction to recommender systems*. (F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor, Eds.), *ACM Transactions on Information Systems* (Vol. 22). Boston, MA: Springer US.

Rocha, A., Scheirer, W. J., Forstall, C. W., Cavalcante, T., Theophilo, A., Shen, B., … Stamatatos,

E. (2017). Authorship Attribution for Social Media Forensics. *IEEE Transactions on Information Forensics and Security*, *12*(1), 5–33.

Rosa, R. L., Rodríguez, D. Z., & Bressan, G. (2015). Music recommendation system based on user's sentiments extracted from social networks. *IEEE Transactions on Consumer Electronics*, *61*(3), 359–367.

Rosaci, D., & Sarné, G. M. L. (2013). Recommending multimedia web services in a multi-device environment. *Information Systems*, *38*(2), 198–212.

Ross, T. J. (2010). *Fuzzy Logic with Engineering Applications*. John Wiley & Sons.

Rovenchak, A. A. (2011). A naïve conception of the uncertainty principle in the multiparametric attribution of texts. *Glottometrics*, *21*, 65–72.

Rutkowski, L. (2013). Artificial Intelligence and Soft Computing 12th International Conference, ICAISC 2013 Zakopane, Poland, June 9-13, 2013 Proceedings. In *12th International Conference, ICAISC 2013 Zakopane, Poland, June 9-13, 2013 Proceedings*.

Sagae, K., & Lavie, A. (2005). A classifier-based parser with linear run-time complexity. In *Proceedings of the Ninth International Workshop on Parsing Technology - Parsing '05* (pp. 125–132).

Sahlgren, M., & Cöster, R. (2004). Using bag-of-concepts to improve the performance of support vector machines in text categorization. In *Proceedings of the 20th international conference on Computational Linguistics* (pp. 487–493).

Savchenko, A. V. (2013). Probabilistic neural network with homogeneity testing in recognition of discrete patterns set. *Neural Networks*, *46*, 227–241.

Schaalje, G. B., Fields, P. J., Roper, M., & Snow, G. L. (2011). Extended nearest shrunken centroid classification: A new method for open-set authorship attribution of texts of varying sizes. *Literary and Linguistic Computing*, *26*(1), 71–88.

Schafer, J., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In *The Adaptive Web. Lecture Notes in Computer Science* (pp. 291–324). Springer.

Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 253–260). CONF, ACM.

Scheirer, W. J., Jain, L. P., & Boult, T. E. (2014). Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(11), 2317–2324.

Schelter, S., Boden, C., & Markl, V. (2012). Scalable similarity-based neighborhood methods with MapReduce. *Proceedings of the Sixth ACM Conference on Recommender Systems - RecSys '12*, 163.

Segarra, S., Eisen, M., & Ribeiro, A. (2015). Authorship Attribution Through Function Word Adjacency Networks. *IEEE Transactions on Signal Processing*, *63*(20), 5464–5478.

Semeraro, G., Degemmis, M., Lops, P., & Basile, P. (2007). Combining Learning and Word Sense

Disambiguation for Intelligent User Profiling. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence* (pp. 2856–2861).

Semeraro, G., Lops, P., & Degemmis, M. (2005). WordNet-based user profiles for neighborhood formation in hybrid recommender systems. In *Proceedings - HIS 2005: Fifth International Conference on Hybrid Intelligent Systems* (pp. 291–296).

Sen, P. K., & Singer, J. M. (1993). *Large Sample Methods in Statistics: An Introduction with Applications*. CRC Press.

Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton university press.

Sharma, C., & Bedi, P. (2018). Community based Hashtag Recommender System (CHRS) for twitter. *Journal of Intelligent and Fuzzy Systems*, *34*(3), 1511–1519.

Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., & Chanona-Hernández, L. (2014). Syntactic N-grams as machine learning features for natural language processing. *Expert Systems with Applications*, *41*(3), 853–860.

Simperl, E., & Luczak-Rösch, M. (2014). Collaborative ontology engineering: A survey. *Knowledge Engineering Review*, *29*(1), 101–131.

Skinner, K. R., Keats, J. B., & Zimmer, W. J. (2001). A comparison of three estimators of the Weibull parameters. *Quality and Reliability Engineering International*, *17*(4), 249–256.

Smeaton, A. F. (1995). Low Level Language Processing for Large Scale Information Retrieval: What techniques actually work. In *Terminology, Information Retrieval and Linguistics*. Rome: CNR.

Somasundaran, S. (2007). QA with Attitude : Exploiting Opinion Type Analysis for Improving Question Answering in On-line Discussions and the News. *Icwsm*, cited 45.

Son, L. H. (2016). Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems*, *58*, 87–104.

Srba, I., & Bielikova, M. (2016). A Comprehensive Survey and Classification of Approaches for Community Question Answering. *ACM Trans. Web*, *10*(3), 18:1--18:63.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, *60*(3), 538–556.

Stamatatos, E. (2013). on the Robustness of Authorship Attribution Based on Character N-Gram Features. *Journal of Law & Policy*, 421–439.

Stamatatos, E., Fakotakis, N., & Kokkinakis, G. (2001). Computer-Based Authorship Attribution Without Lexical Measures. In *Computers and the Humanities* (Vol. 35, pp. 193–214).

Stancu-Minasian, I. M. (2012). *Fractional programming: theory, methods and applications*. Springer Science & Business Media.

Stańczyk, U. (2013). Rough Set and Artificial Neural Network Approach to Computational Stylistics (pp. 441–470).

Stańczyk, U. (2015). Selection of decision rules based on attribute ranking. *Journal of Intelligent*

*& Fuzzy Systems*, *29*(2), 899–915.

Stent, A., Choi, J. D., St, W., St, W., & York, N. (2015). It Depends : Dependency Parser Comparison Using A Web-based Evaluation Tool. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 387–396).

Su, J.-H., Wang, B.-W., Hsiao, C.-Y., & Tseng, V. S. (2010). Personalized rough-set-based recommendation by integrating multiple contents and collaborative information. *Information Sciences*, *180*(1), 113–131.

Su, J., & Zhang, H. (2006). A Fast Decision Tree Learning Algorithm. In *21st national conference on Artificial intelligence - Volume 1* (Vol. 5, pp. 500–505).

Su, X., & Khoshgoftaar, T. M. (2009). A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, *2009*(4), 1–19.

Sugeno, M. (1972). *Theory of fuzzy integrals and its application*. *Tokyo Institute ofTechnology*.

SUN, D., HE, T., & ZHANG, F. (2012). Survey of Cold-start Problem in Collaborative Filtering Recommender System. *Computer and Modernization*, *5*, 59–63.

Sun, H.-F., Chen, J.-L., Yu, G., Liu, C.-C., Peng, Y., Chen, G., & Cheng, B. (2012). JacUOD: A New Similarity Measurement for Collaborative Filtering. *Journal of Computer Science and Technology*, *27*, 1252–1260.

Sun, H., Peng, Y., Chen, J., Liu, C., & Sun, Y. (2011). A New Similarity Measure Based on Adjusted Euclidean Distance for Memory-based Collaborative Filtering. *Journal of Software*, *6*(6), 993–1000.

Sun, J., Wang, G., Cheng, X., & Fu, Y. (2015). Mining affective text to improve social media item recommendation. *Information Processing and Management*, *51*(4), 444–457.

Sun, Q., Wang, L., Wang, S., Ma, Y., & Hsu, C.-H. (2016). QoS prediction for Web service in Mobile Internet environment. *New Review of Hypermedia and Multimedia*, *22*(3), 207–222.

Takács, G., Pilászy, I., Németh, B., & Tikk, D. (2009). Scalable Collaborative Filtering Approaches for Large Recommender Systems. *The Journal of Machine Learning Research*, *10*, 623–656.

Tenne, Y., & Goh, C. K. (Eds.). (2010). *Computational Intelligence in Expensive Optimization Problems* (Vol. 2). Springer Science & Business Media.

Tewari, A. S., & Barman, A. G. (2018). Sequencing of items in personalized recommendations using multiple recommendation techniques. *Expert Systems with Applications*, *97*, 70–82.

Ting, S., Ip, W., & Tsang, A. H. (2011). Is Naive Bayes a good classifier for document classification? *International Journal of Software Engineering and Its Applications*, *5*(3), 37–46.

Trevisiol, M., Chiarandini, L., & Baeza-Yates, R. (2014). Buon Appetito: Recommending Personalized Menus. *Proceedings of the 25th ACM Conference on Hypertext and Social Media - HT '14*, 327–329.

Trias i Mansilla, A., & de la Rosa i Esteva, J. L. (2012). Asknext: An agent protocol for social search. *Information Sciences*, *190*, 144–161.

Trias i Mansilla, A., & de la Rosa i Esteva, J. L. (2013a). Question Waves: A multicast query routing algorithm for social search. *Information Sciences*, *253*, 1–25.

Trias i Mansilla, A., & de la Rosa i Esteva, J. L. (2013b). Survey of social search from the perspectives of the village paradigm and online social networks. *Journal of Information Science*, *39*(5), 688–707.

Velldal, E. (2005). A fuzzy clustering approach to word sense discrimination. In *Proceedings of the 7th International conference on Terminology and Knowledge Engineering*.

Vossen, P. (1998). Introduction to EuroWordNet. In *Computers and the Humanities* (Vol. 32, pp. 73–89).

Vossen, P. (2004). EuroWordNet: A multilingual database of autonomous and language-specific WordNets connected via an inter-lingual-index. *International Journal of Lexicography*, *17*(2), 161–173.

Voudouris, C., Tsang, E. P. K., & Alsheddy, A. (2010). Guided Local Search. In *Handbook of Metaheuristics*.

Wang, L.-X. (1997). *A Course in Fuzzy Systems and Control*. Prentice-Hall, Inc.

Wang, M., & Ma, J. (2016). A novel recommendation approach based on users' weighted trust relations and the rating similarities. *Soft Computing*, *20*(10), 3981–3990.

Wang, S., & Manning, C. (2012). Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics* (pp. 90–94).

Wang, Y., Liu, Y., & Yu, X. (2012). Collaborative filtering with aspect-based opinion mining: A tensor factorization approach. *Proceedings - IEEE International Conference on Data Mining, ICDM*, 1152–1157.

Weaver, W. (1955). Translation. *Machine Translation of Languages*, *14*, 15–23.

Wei, T., Zhou, Q., Chang, H., Lu, Y., & Bao, X. (2015). A Semantic Approach for Text Clustering using WordNet and Lexical Chains. *Expert Systems with Applications*.

Weng, Z. X., Shi, L. B., Xu, Z., Yao, L. Z., Ni, Y. X., & Bazargan, M. (2012). Effects of wind power variability and intermittency on power flow. In *IEEE Power and Energy Society General Meeting*.

Whaley, J. M., & Aslam, J. A. (1999). An Application of Word Sense Disambiguation to Information Retrieval. *Dartmouth College, Department of Computer Science*.

Wood, W., & Rünger, D. (2016). Psychology of Habit. *Annual Review of Psychology*, *67*(1), 289–314.

Wu, Y., & Ester, M. (2015). FLAME:A Probabilistic model combining aspect based opinion mining and collaborative filtering. In *WSDM 2015 - Proceedings of the 8th ACM*

*International Conference on Web Search and Data Mining* (pp. 199–208).

Yan, J., Wang, C., Cheng, W., Gao, M., & Zhou, A. (2018). A retrospective of knowledge graphs. *Frontiers of Computer Science*, *12*(1), 55–74.

Yang, D., Zhang, D., Yu, Z., & Wang, Z. (2013). A sentiment-enhanced personalized location recommendation system. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media - HT ’13* (pp. 119–128).

Yang, X., Xu, G., Li, Q., Guo, Y., & Zhang, M. (2017). Authorship attribution of source code by using back propagation neural network based on particle swarm optimization. *PLoS ONE*, *12*(11).

Ye, P., & Baldwin, T. (2006). Verb Sense Disambiguation Using Selectional Preferences Extracted with a State-of-the-art Semantic Role Labeler. In *Technology* (pp. 139–148).

Yuan, G., Murukannaiah, P. K., Zhang, Z., & Singh, M. P. (2014). Exploiting sentiment homophily for link prediction. *Proceedings of the 8th ACM Conference on Recommender Systems - RecSys ’14*, 17–24.

Zablith, F., Antoniou, G., d’Aquin, M., Flouris, G., Kondylakis, H., Motta, E., … Sabou, M. (2015). Ontology evolution: a process-centric survey. *The Knowledge Engineering Review*, *30*(1), 45–75.

Zadeh, L. a. (1978). Fuzzy Sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, *1*(1), 3–28.

Zeldes, A., & Schroeder, C. T. (2015). Computational methods for Coptic: Developing and using part-of-speech tagging for digital scholarship in the humanities. *Digital Scholarship in the Humanities*, *30*, i164–i176.

Zhang, H. (2004). The Optimality of Naive Bayes. In *Proceedings of the the 17th International FLAIRS conference (FLAIRS2004)*. AAAI Press ©2004.

Zhang, J.-D., Chow, C.-Y., & Zheng, Y. (2015). ORec: An Opinion-Based Point-of-Interest Recommendation Framework. *Roceedings of the 24th ACM International on Conference on Information and Knowledge Management - CIKM ’15*, 1641–1650.

Zhang, L., Xie, M., & Tang, L. (2008). On Weighted Least Squares Estimation for the Parameters of Weibull Distribution. In *Recent Advances in Reliability and Quality in Design* (pp. 57–84).

Zhang, W., Ding, G., Chen, L., Li, C., & Zhang, C. (2013). Generating virtual ratings from chinese reviews to augment online recommendations. *ACM Transactions on Intelligent Systems and Technology*, *4*(1), 1–17.

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: A statistical framework. *International Journal of Machine Learning and Cybernetics*, *1*(1–4), 43–52.

Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., & Ma, S. (2014). Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval - SIGIR ’14*, 83–92.

Zhao, Y., & Zobel, J. (2007). Searching with style: Authorship attribution in classic literature. In *Conferences in Research and Practice in Information Technology Series* (Vol. 62, pp. 59–68).

Zheng, Z., Ma, H., R.Lyu, M., & King, I. (2009). WSRec: A Collaborative Filtering Based Web Service Recommender System. In *IEEE International Conference on Web Services (ICWS)* (pp. 437–444).

Zibin Zheng, Hao Ma, Lyu, M. R., & King, I. (2011). QoS-Aware Web Service Recommendation by Collaborative Filtering. *IEEE Transactions on Services Computing*, *4*(2), 140–152.