JOKE RECOMMENDER SYSTEM USING HUMOR THEORY

by

Soumya Agrawal

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science

Department of Computer and Information Technology

West Lafayette, Indiana

August 2020

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Julia Taylor Rayz, Chair
Department of Computer and Information Technology

Dr. John Springer
Department of Computer and Information Technology

Dr. Victor Raskin
Department of English

Approved by:
Dr. John Springer

Dedicated to my mom and ma, the two strongest women in my life

ACKNOWLEDGMENTS

I would like to express my gratitude to Dr. Julia Rayz for her guidance from the day I started my journey towards the master's degree. I would like to thank her for sparking my interest in this wonderful field of humor research. Without her help and support throughout these two year, it would have been impossible to finish this research.

This research would be incomplete without Dr. Victor Raskin whose work in the field made the research possible. Just listening to him talk about his experiences is a privilege in itself. I also want to thank Dr. John Springer for his valuable suggestions which helped in shaping the work.

My family also deserve many thanks for their love, support and encouragement. I have been truly blessed to have a family who always believed in me.

I would also like to thank my friend Ishan who helped me balance the academic and social life such that I was never over-whelmed by work.

Last but not the least, I want to thank the members of the AKRANLU lab who gave me invaluable suggestions and feedback.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

GTVH – General Theory of Verbal Humor

KR – Knowledge Resources

LA – Language (GTVH knowledge resource)

LM – Logical Mechanism (GTVH knowledge resource)

NS – Narrative Strategy (GTVH knowledge resource)

OSTH – Ontological Semantic Theory of Humor

PCA - Principal Component Analysis

RS – Recommendation System

SI – Situation (GTVH knowledge resource)

SO – Script Opposition (GTVH knowledge resource)

SSTH – Script-based Semantic Theory of Humor

TA – Target (GTVH knowledge resource)

ABSTRACT

The fact that every individual has a different sense of humor and it varies greatly from one person to another means that it is a challenge to learn any individual's humor preferences. Humor is much more than just a source of entertainment; it is an essential tool that aids communication. Understanding humor preferences can lead to improved social interactions and bridge existing social or economic gaps.

In this study, we propose a methodology that aims to develop a recommendation system for jokes by analyzing its text. Various researchers have proposed different theories of humor depending on their area of focus. This exploratory study focuses mainly on Attardo and Raskin's (1991) General Theory of Verbal Humor and implements the knowledge resources defined by it to annotate the jokes. These annotations contain the characteristics of the jokes and also play an important role in determining how alike these jokes are. We use Lin's similarity metric (Lin, 1998) to computationally capture this similarity. The jokes are clustered in a hierarchical fashion based on their similarity values used for the recommendation. We also compare our joke recommendations to those obtained by the Eigenstate algorithm (Goldberg, Roeder, Gupta, & Perkins, 2001), an existing joke recommendation system that does not consider the content of the joke in its recommendation.

CHAPTER 1. INTRODUCTION

People somehow know what humor is, but still, find it difficult to 'define' it (McGhee & Pistolesi, 1979). Humor is a multidisciplinary field and has been studied from perspective of these disciplines. Various empirical findings have confirmed that stress and depressing thoughts can be regulated with the help of humor (Francis, Monahan, & Berger, 1999). Positive psychology, a field that examines what people do well, points out that humor can be used to reduce tension, make friends, make others feel good or to help buffer stress (Lurie & Monahan, 2015; Ruch & Heintz, 2016). Within linguistics, theories have been proposed about joke structure (Attardo & Raskin, 1991; Raskin, 1985). The emergence of Artificial Intelligence has sowed the seeds of the idea that computers can understand the human language. Since humor is a universal aspect of the human life, computers may be expected take into consideration the humorous facet.

As the technology advances, many researchers have presented their findings which point out the need for computational humor. Some of the applications of computational humor are human-computer interfaces (Morkes, Kernal, & Nass, 1998), education (McKay, 2002), edutainment (Stock, 1996), understanding how human brain works (Binsted et al., 2006; Ritchie, 2001), etc.

It was brought to light that if computer systems can incorporate humor mechanisms more efficiently, then they would appear to be more user-friendly hence less alien and intimidating (Binsted, 1995). One of the key things to consider in order to achieve this, is that different people find different things funny. It is also important to make sure that a joke should be both funny enough and not offend someone at the same time. These factors make research in this field challenging and call for a computer system that incorporate the humor preferences of the user.

Verbal humor is a common form of humor, and one of the subclasses of verbal humor is the joke. A joke can be defined as “a short humorous piece of literature in which the funniness culminates in the final sentence” (Hetzron, 1991). This research focuses on verbally expressed humor with the help of jokes.

The motivation behind this research comes by observing the smart assistants like Alexa and Siri. The fact that these smart assistants recite the same jokes to the speakers irrespective of their humor preferences determines the problem that this research aims to work on. The goal of this research is to take a step closer to understanding human humor preferences and hence recommending jokes accordingly. This exploratory study tries to build a framework that can make an impact on making computers better understand the human language, which could eventually find its applications into AI.

We propose a framework to recommend jokes to the users by taking into consideration the text of the joke as well as the user liking of these jokes. To achieve this, we also assume that individuals like types of jokes and we can identify these types through the individual’s funniness ratings. The proposed framework is based on the identification and quantification of similarity between jokes. We represent and compare the jokes in the Jester dataset with the help of six knowledge resources as defined by the General Theory of Verbal Humor (Attardo & Raskin, 1991). Once similar jokes are identified, we explore whether subject ratings confirm the similarity. Finally, we compare our joke recommendations to those done by the existing joke recommendation system, Jester, (Goldberg et al., 2001). Jester treats the text of the joke as a black box and relies solely on the user ratings for the recommendation. It works as a baseline model to our proposed model and we compare the joke recommendations to the same user by both the models. We also analyze the ratings given by the users to the jokes that are considered similar to our model.

1.1. Significance

To the best of our knowledge, this is the first joke recommendation study that takes the text of the jokes into account as it builds its recommendation based on a humor theory.

1.2. Research Question

This research attempts to answer the following question:

a. Can joke recommendation be improved by considering humor theory inspired features of the jokes?

1.3. Assumptions

There are three assumptions of this research:

a. The dataset used in this research is representative enough of the humor field which implies that the results obtained from the research are generalizable.
b. The joke annotations based on one of the humor theories and received from domain knowledge experts are accurate.
c. Joke ratings of the Jester corpus accurately represents perceived funniness ratings of users.

1.4. Limitations

There are three limitations of this research:

a. The dataset that is used in this research does not consider fatigue effects which might affect user ratings of jokes. There might be cases when users might get tired of rating jokes back to back, which might manipulate the joke rating. This is not taken into account.
b. There might be an effect of a previously heard joke on joke rating by the user.
c. The baseline model has an undue advantage over the proposed model because it is trained on the user ratings which are eventually used to compare the final results.

CHAPTER 2. LITERATURE REVIEW

2.1 The Evolution of Humor Theories

Over the years, a lot has been written on the topic of humor and various theories have been proposed. Usually, these theories can be divided into three major classes: superiority theories, release/relief theories, and incongruity theories. These classes of theories are discussed here since it is paramount to understand them before beginning to play with humor.

2.1.1 Superiority Theories

Superiority theories date back to the era of Plato (*Republic, Philebus*) and Aristotle (*Ethics, Poetics*) and both of these thinkers associated humor with vulgarity, vice and offense. According to Plato, the person who is laughing often wrongly perceives himself/herself to be superior (in terms of being richer, better-looking) than the person he/she is ridiculing (Larkin-Galiñanes, 2017). The general idea behind these theories was that people laugh at other people's misfortunes since it makes them feel superior to them (Attardo, 1994); (Raskin, 1985).

2.1.2 Release/Relief Theories

Sigmund Freud, founder of psychoanalysis, related humor with conscious and viewed jokes as a way to express thoughts that are generally forbidden by society. Release/relief class of theories asserts that humor and laughter are a result of the release of nervous energy (Meyer, 2000). People generally use humor by telling a joke to handle an intense situation (Raskin, 1985).

2.1.3 Incongruity Theories

The family of incongruity theories state that humor arises when something which was not anticipated happens. Aristotle (*Rhetoric*) was the first philosopher to present the earliest glimmer that humor is based on incongruity. Kant (1911) and Schopenhauer (1907) additionally agreed with the idea that incongruity is a necessary condition for humor. There has been a debate among various thinkers if incongruity alone can be considered to be sufficient enough to be able to mark something as funny as theorists such as Schultz (1976) and Suls (1972, 1983) believed that that to enjoy humor the resolution of incongruity is also essential. This gave birth to the Incongruity-Resolution theories which focused not only incongruity but also on its realization and resolution. Suls (1972) proposed a two-stage model that stated that when there is some incongruity in the text (identified generally at the end), if one can resolve it then it's a joke otherwise the text leads to no laughter and puzzlement (Ritchie, 1999).

Another model to resolve incongruity was summarized by Ritchie (1999) as the surprise disambiguation model. The model states that the setup of the joke has two different interpretations out of which one is more obvious than the other. The hidden meaning of the text is triggered once the punchline is reached, and this is how the audience becomes aware of the new interpretation.

Joke$_1$: *"Why do birds fly south in winter?*

It's too far to walk."

When the readers read the first line, they think of reasons like warmth, food for birds to fly south. But when they read the punchline, the hidden meaning of the text is triggered which is that they can't walk that far hence they fly.

The jokes are formulated with the intention of making them ambiguous up until the point when the punchline comes which works as a trigger which then makes the readers realize that a different meaning of the script had to be used from the very beginning.

Ritchie (1999) compares the two models and lists some differences between them such as the SD model attempted to handle ambiguity by requiring an ambiguous setup, whereas the two-stage model had no mention of ambiguity. The SD model addressed the issue of incongruity better than the other model.

Rothbart & Pien (1977) suggested there should be a further classification of incongruity and resolution and proposed two categories of each. They described the categories as follows:

- "**Impossible Incongruity**: elements that are unexpected and also impossible given one's current knowledge of the world
- **Possible Incongruity**: elements that are unexpected or improbable but possible
- **Complete Resolution**: the initial incongruity follows completely from resolution information
- **Incomplete Resolution**: the initial incongruity follows from resolution information in some way, but is not made completely meaningful because the situation remains impossible"

2.1.4 Script-based Semantic Theory of Humor

The Script-based Semantic Theory (Raskin, 1985) made a profound difference in the field of linguistics of humor. SSTH is designed as neutral and compatible with all the three theories (Raskin, 1985). Raskin pointed out that all the three family of theories discussed above were non-conflicting, rather they supplemented each other since as all of them addressed different aspects of humor such as the Incongruity based theories dealt with stimulus (Speaker), the Superiority theories work with the relationship between the speaker and the hearer and the Release/Relief theories commented on the feelings of the hearer. The below diagram shows the “map” of the humor theory, in which linguistics is either on a different plane or spread across the board (Attardo & Raskin, 2017):

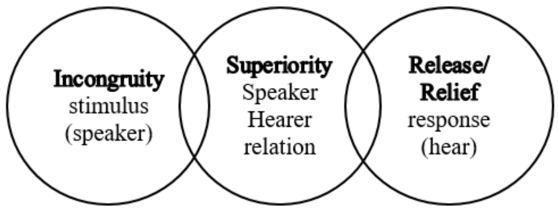

Figure 1: Map of Humor Theory (Attardo & Raskin, 2017)

Since all the further theories this thesis works with is based on SSTH, it is be both interesting and important to understand it in depth. SSTH is the application of semantic script theory to the field of humor.

Scripts are organized chunks of information which are related to a particular task or event. For example: If someone says, “*I got admitted into Purdue University as a graduate student*”, while the literal implication is just that someone got accepted by the school, but the contextual meaning implies that s/he has probably done a bachelor’s degree, gave an exam, applied to the school, wrote SOP, the school committee reviewed the application, etc. So, the script captures all of this concerning the event of getting admitted to the school.

The main hypothesis of SSTH is as follows:

“A text can be characterized as a single-joke-carrying text if both of the [following] conditions are satisfied:

(i)The text is compatible, fully or in part, with two different scripts.

(ii) The two scripts with which the text is compatible are opposite. [...] The two scripts with which some text is compatible are said to overlap fully or on part on this text.

The set of two conditions [above] is proposed as necessary and sufficient conditions for a text to be funny." (Raskin, 1985).

The working of SSTH can be better understood by the following analysis:

Joke2*: "'Is the doctor at home?' the patient asked in his bronchial whisper. 'No,' the doctor's young and pretty wife whispered in reply. 'Come right in.'"*

Joke2 triggers the following two scripts, containing information described below (Raskin, 1985):

DOCTOR

Subject: [+Human] [+Adult]
Activity: > Study medicine
= Receive patients: patient comes or doctor visits doctor listens to complaints doctor examines patient
= Cure disease: doctor diagnoses disease doctor prescribes treatment
= (Take patient's money)
Place: > Medical School
= Hospital or doctor's office
Time: > Many years
= Every
=Immediately
Condition: Face to Face

LOVER

Subject: [+Human] [+Adult] [+Sex: x]
Activity: > Makelove
Object: [+Human] [+Adult] [+Sex: x]
Place: Secluded
Time: > Once

= Regularly
Condition: If subject or object married spouse(s) should not know

Despite the joke being obscene, it illustrates the collision of the scripts very well. When the above joke is read, the script of a DOCTOR is evoked due to the words "*doctor", "patient"* and "*bronchial"*. The pronoun in the sentence makes is clear that the subject is a male who he is seeking a doctor for medical help. The second script, Lover, is triggered by the words "no" as well as the description of the doctor's wife. The wife's reply is incongruous to the first script, and thus the second script emerges, which makes the punchline, "come right in" explainable. The joke is said to have a partial script overlap between Doctor and Lover – both scripts contain a person that comes to the doctor's house for a visit – and since these scripts are opposing each other based on sex/non-sex, the text is considered a joke (Raskin, 1985).

While SSTH is one of the most well-known linguistic theories of humor, it has some drawbacks as listed by Attardo (Attardo). Firstly, this theory works well when the text is a joke, but when humorous texts other than jokes are considered it fails to give the desired results. Secondly, SSTH fails to explain the similarity between two jokes. This led to the birth of the General Theory of Verbal Humor by Attardo and Raskin (1991).

2.1.5 General Theory of Verbal Humor

General Theory of Verbal Humor was formed by combining SSTH by Raskin with the Five-level joke representation model by Attardo (Attardo & Raskin, 1991). This theory proposed that all the jokes can be represented in terms of the following six Knowledge Resources:

- **Language (Cleger-Tamayo, Fernandez-Luna, & Huete):** refers to the choices made at the syntactical, semantical, pragmatical, etc. levels and most importantly it is responsible for the wording and the placement of, the punchline.
- **Narrative Strategy (NS):** refers to the genre of the joke like narrative, question- answer, riddle, etc.
- **Target (TA):** refers to the "butt" of the joke such as stereotypes, certain professions, etc. which is the only optional parameter out of the six KR.
- **Situations (SI):** refers to the "props" of the joke such as the activity or object or participants.
- **Logical Mechanism (LM):** deals with how the different scripts are linked together with the help of various mechanisms like false analogies, figure-ground reversals, etc.
- **Script Opposition (SO):** refers to the SSTH's script opposition such as real/unreal.

Attardo & Raskin proposed to use all the six KRs as a six-argument template for the representation of the joke. They use this representation to compare the jokes and draw conclusions about the similarities between them on the basis of the number of common KRs. A hierarchy between the six KRs was also established after critically analyzing various interdependence and independence amongst them. The following order was proposed:

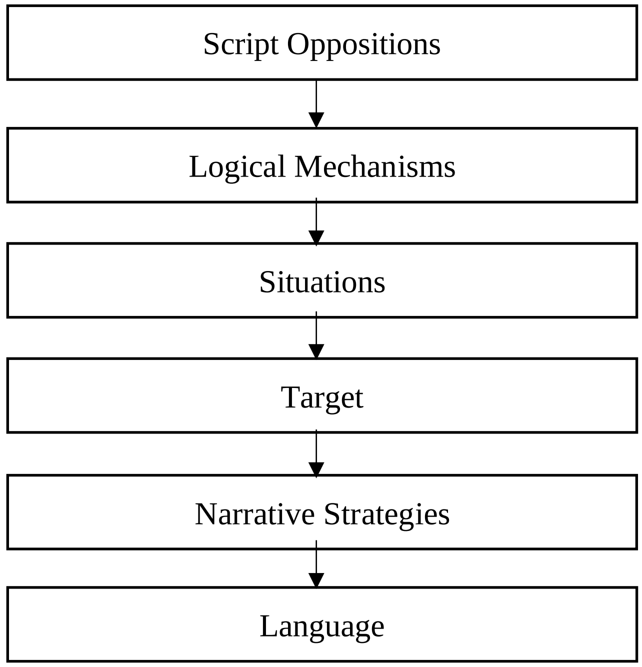

Figure 2: KR Hierarchy (Attardo & Raskin, 1991)

The order is such that once a selection in the knowledge resource at an upper level is made, it limits the choices available in the lower level. This hierarchy was experimentally confirmed, with the exception of Logical Mechanism (Ruch, Attardo, & Raskin, 1993).

Attardo and Raskin illustrate the comparison on an example of the following three jokes:

Joke$_3$: *"How many Irishmen does it take to screw in a light bulb? Five. One to hold the light bulb and four to turn the table he's standing on."*

Joke$_4$: *"How many Poles does it take to wash a car? Two. One to hold the sponge and one to move the car back and forth".*

Joke5*: "Do you think one Pole can screw in a light bulb?" "No." "Two?" "No." "Three?" "No. Five. One to screw in a light bulb and four to turn the table he's standing on."*

Table 1: Comparison of the jokes (Attardo & Raskin, 1991)

Joke-KR	**LA**	**NS**	**TA**	**SI**	**LM**	**SO**
3	LA 1	Riddle	Irish	Light Bulb	Figure Ground Reversal	Dumbness
4	LA 1	Riddle	Poles	Car Wash	Figure Ground Reversal	Dumbness
5	LA2	Ques -Ans	Poles	Light Bulb	Figure Ground Reversal	Dumbness

Jokes 4 and 5 differ in three of the parameters, namely, LA, NS, and SI; jokes 3 and 4 differ in two of them, namely TA and SI; and jokes 3 and 5 in three of them, namely LA, NS and TA. Jokes 3 and 4 are the most similar jokes since they differ in only two KRs while the other pairs differ in three KRs. Since TA is placed at a lower level in the hierarchy than SI, jokes 4 and 5 are least similar even though they differ in the same number of KRs as jokes 3 and 5.

GTVH had various advantages over SSTH as it was able to address the degree of similarities between the jokes using the six-tuple representation. GTVH was later extended to handle humorous texts other than jokes. But the main drawback of GTVH was that it didn’t address any computational aspects and the selection of scripts was largely dependent on domain experts. These issues were addressed by the Ontological Semantic Theory of Humor, described next.

2.1.6 Ontological Semantic Theory of Humor

The OSTH, described by (Raskin, Hempelmann, & Taylor, 2009), states that Ontological Semantic Technology (OST: (Raskin, Hempelmann, & Taylor, 2010), (Taylor, Hempelmann, & Raskin, 2010), (Hempelmann, Taylor, & Raskin, 2010)) can be used to represent the meaning of the text in the joke. To fully appreciate this theory, it is important to first understand what Ontological Semantic Technology (OST) and ontology mean.

Ontology is referred by a different meaning in various communities but in Computer Science, it is often referred to as a formal representation of concepts along with concepts and relationships associated. Ontology is viewed as a means to facilitate human-machine communication (Guarino, Oberle, & Staab, 2009). Since different people have a different understanding of the same concept, ontology works as a conflict-resolving method and hence avoids misunderstandings by describing everything with precision about a phenomenon and nothing outside of it. It is also important to note that ontology is language independent.

OST is defined as “a theory, methodology and, especially, technology” which could be used to represent the natural language with the help of conversion of text into text-meaning representations (TMRs), which also allowed advanced reasoning by manipulating the TMRs (Taylor, Raskin, Hempelmann, & Attardo, 2010). OST ontology will have thousands of concepts connected to one or more properties and has only those meanings as entries which cannot be formed by the combination of other entries.

Coming back to OSTH, the authors believe that to understand humor, it is important to first get a comprehensive understanding of the text. This is achieved by the use of ontological semantics which captures the meaning of all the sentences in the text in every context with the help of lexical scripts. These scripts get activated by the lexical sense and each script further activates more of such scripts.

The Joke2 used in the SSTH can be analyzed again to explain the working of OSTH (Rayz, 2020). The first step would be to define the lexical scripts of both Doctor and Lover. Then these scripts are translated into lexical sense using ontological semantics. They would look something like this:

```
doctor-n1
pos: n
morph:
        NNS +s
           syn-struc: NP var0
           sem-struc: DOCTOR_MD

doctor-n2
pos: n
morph:
        NNS +s
           syn-struc: NP var0
           sem-struc: DOCTOR_PHD

lover-n1
pos: n
morph:
        NNS +s
           syn-struc: NP var0
           sem-struc: LOVER
```

It is stated that all the concepts are SOCIAL-ROLES, which means that DOCTOR is a social role that a person plays in which s/he attends medical school and then treats patients. Similarly, PATIENT is also a social role in which the person seeks medical help. It is important to understand that a person can play multiple social roles at the same time as a person can play the role of both a PATIENT and a HUSBAND.

The next step is to parse the sentence. The noun patient is responsible for triggering the lexical sense of *patient-n1*, which in turn activates the social role PATIENT. Similar processing for the role DOCTOR_MD happens due to the lexical sense of *doctor-n1*. While the lexical sense of doctor-n2 activates the role of DOCTOR-PHD but the words “bronchial whisper” result in a low level of activation. Then the second sentence is parsed in which another social role of a WIFE is introduced, which makes the social role of the PATIENT to change to VISITOR. This change in social role by a person can emphasize the script overlap. The third sentence shows that the WIFE invites the VISITOR in the house after she confirms that the DOCTOR is not home, which would trigger the script of LOVER/ADULTERY (this is based on the assumption that the ontology acknowledges stereotypes). This brings out the script opposition: receive medical help (stated goal) vs did not receive (implied result) (Raskin et al., 2009).

The main key feature of OSTH is that it tries to capture the understanding of the text by humans which could be the key to make computers better understand humor. In a deeper analysis of scripts in the OSTH, it is also argued that that the annotations of the scripts play no role in knowing if the knowledge is activated or not, hence saves time wasted in arguing what to correctly call the scripts (Rayz, 2020). So, whether somebody calls the script LOVER as ADULTERY, it doesn’t matter in computational use.

2.2 Recommendation Systems

Since the goal of the research is to develop a joke recommendation system, the following section gives a brief review of the types of recommendation systems (RS). Recommendation systems can be defined as the system that aims to predict which item would be preferred by the user based on filtering some information. RS can be seen as a type of intelligent system which exploits the available user rating of items to make recommendations. RS has become one of the most popular applications of big data analytics as many big companies like Google (Das, Datar, Garg, &

Rajaram, 2007), Amazon (Linden, Smith, & York, 2003), Netflix (Koren, 2009), etc. have deployed RS as their business solution. There is also a great volume of literature available on recommendation system based on movie (Choi, Ko, & Han, 2012) news (Cleger-Tamayo et al., 2012 2012), e-learning, e-commerce (Leino, 2014) music (Li, Myaeng, & Kim), travel (Ravi & Vairavasundaram, 2016), etc.

The recommendation systems can be broadly characterized into two classes: Content-based systems and Collaborative filtering-based systems. The content-based systems use the personal preferences of the user by learning the features of the item liked by the user and then suggesting items with similar features (Shani & Gunawardana, 2011), whereas the collaborative filtering-based systems are based on finding the target users' neighborhood users and then suggesting the items like by them to the target user (Cai, Leung, & Li, 2014).

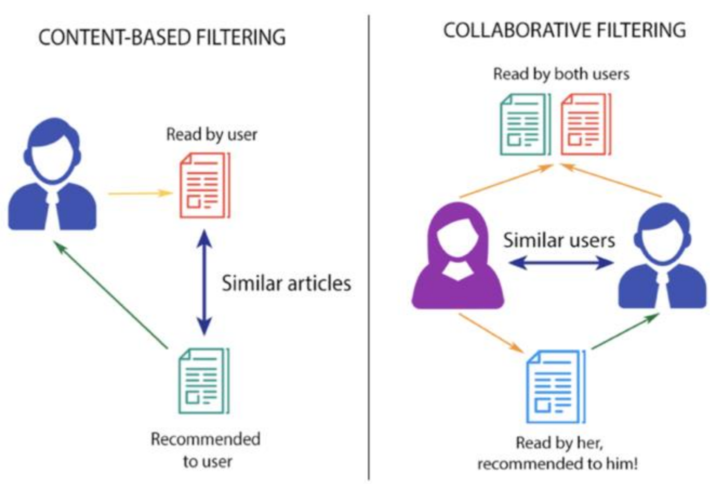

Figure 3: Types of Recommendation Systems (Mohamed, Khafagy, & Ibrahim, 2019)

Most recommender systems these days combine both content-based and collaborative filtering-based approaches to give rise to a hybrid recommendation system. There are studies that compare the performance of the three types of recommender systems and the hybrid methods have provided better results than the other two (Adomavicius & Tuzhilin, 2005).

Due to the popularity of the RS, there is a vast amount of research going on to improve these systems. Scalability is one of the key requirements of RS keeping which in mind Ying, He, Chen, Eksombatchai, Hamilton & Leskovec (2018) proposed a large-scale deep recommendation system which is deployed at Pinterest. Another factor affecting the performance of RS is its ability to understand the context while recommending items. Beutel et al. (2018) presented a technique to incorporate various contextual features like date, time, location, user device which might play a role in deciding whether to go with a product or not.

CHAPTER 3. METHODOLOGY

3.1 Dataset

For this research, the Jester dataset has been used which is available publicly, courtesy to a research team from UC Berkeley. We use version 3[1] of the dataset which is an updated dataset of the previous versions. Version 1 (Goldberg et al., 2001) had rating values from -10 to +10 of 100 jokes collected between April 1999 to May 2003; and the version 2 had 50 more jokes with 115,000 new ratings collected between November 2006 to May 2009. Overall, the version 3 of the dataset has over 1.8 million continuous ratings of 150 jokes from 54,905 anonymous users which were collected from November 2006 to March 2015. It should be noted that many jokes in the dataset are no longer relevant (out of date), but they can nevertheless be used to test the methodology. The dataset consists of a set of 8 jokes termed as *gauge set*, as these jokes are rated by all the users. The remaining non-gauge jokes have a very sparse rating matrix since around 82% of the user ratings are null.

The following table describes the dataset in more detail:

Table 2: Dataset description

No. of jokes rated by all the users (***Gauge set***)	8
No. of jokes not rated by any user	22
Average number of jokes rated by every user	33
No. of users who have rated at least 50 jokes	10503

[1] http://eigentaste.berkeley.edu/dataset/

The authors of the dataset also shared some information about the population and the sampling technique. They initially gauge a set of 40 jokes which were popularly used in existing humor literature, while ensuring that highly offensive jokes were excluded. Later 110 more jokes were added to the repository from other sources. The sampling of users was done by a non-probability sampling method. The people who had signed up for the newsgroup of the research lab were asked to rate the jokes, hence a convenience sampling was implemented. Also, it is worth noting that the users were provided by a horizontal "rating bar" which had to be clicked by the mouse and it returned scalar values.

For the purpose of this research, the domain knowledge experts labeled each joke of this dataset with its six features as defined by GTVH. We wish to point out that two pairs of jokes in the dataset are identical and we decided to remove the duplicate from measuring joke similarity.

3.2 Baseline Model

The baseline model is Jester which is an existing online joke recommender system (Goldberg et al., 2001). The authors proposed Eigentaste, a constant-time recommendation algorithm, a collaborative filtering algorithm which recommended jokes to the users on the basis of their rating of the gauge set jokes. The flowchart of their algorithm is as follows:

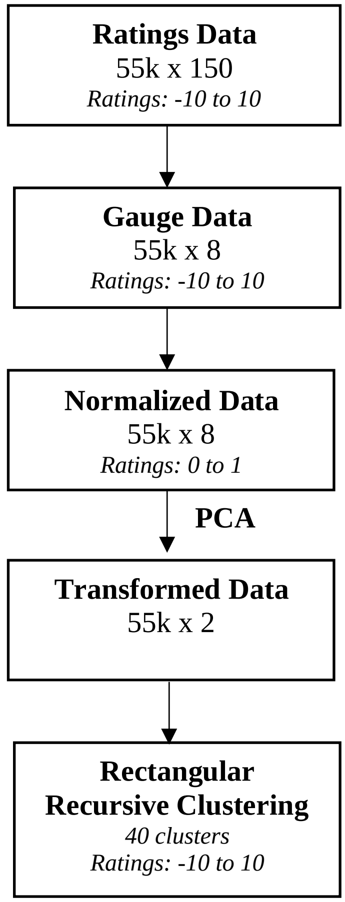

Figure 4: Baseline Model Flowchart (Goldberg et al., 2001)

Figure 4 depicts how the user ratings are processed to build the model. To overcome the problem of the sparse rating matrix, the model is built on the ratings of gauge set jokes only. The algorithm uses Principal Component Analysis (PCA) to reduce the dimensions optimally. PCA, first introduced by Karl Pearson (1901), is a statistical procedure that relies on orthogonal transformation to reduce the dimensionality of large datasets while increasing interpretability and minimizing information loss. This is made possible by generating new variables that maximize the variance. To decide on the optimal number of variables a scree plot is created. It shows how much variance can be explained by different numbers of variables (Lewith, Jonas, & Walach, 2011). One popular choice is to use two variables because that makes the visualization of the reduced data easy. Also, the scree plot below shows that the first two variables together account for almost 60% of the total variance. Therefore, we reduced the data in two dimensions which represent the user projections in a 2D plane.

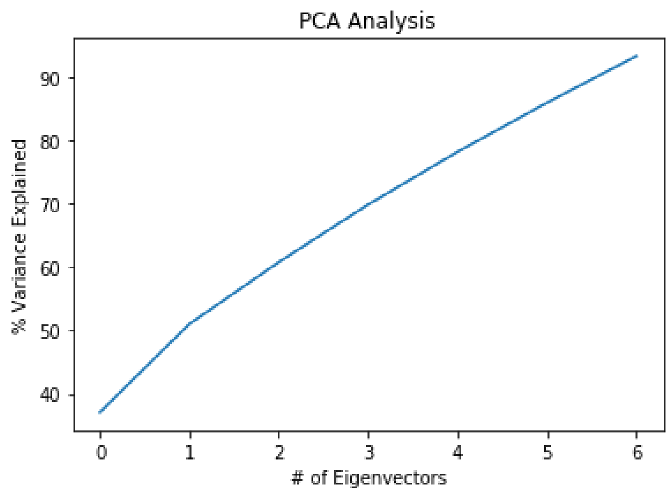

Figure 5: Scree curve of variances explained by consecutive eigenvectors

There were different clustering algorithms that the authors explored. But after plotting the users' projections in a 2D plane using PCA, it was observed that there was a high concentration of data around the origin. Due to the nature of the reduced data, a clustering algorithm called "Recursive Rectangular" was implemented by the authors. The idea behind this clustering technique was to bisect the plane around the origin to form clusters that were in the shape of a rectangle. Each cluster/rectangle which touched the origin was again bisected to produce smaller clusters.

For all the clusters, the average rating of the non-gauge jokes was calculated which was then used to sort the jokes in the decreasing order of preference for recommendation. This yielded a lookup recommendation table for each cluster. Whenever a new user entered the system, the ratings from the gauge set were collected and represented as a vector which was then projected into the 2D plane and the representative cluster of that user was found. Once the cluster is found, the joke is recommended from the lookup table. It's important to note that in this model, the user's ratings of non-gauge jokes have no influence on which joke would be recommended next.

The author state that their algorithm's accuracy "is as good as 80-NN but its online computation is faster by two orders of magnitude" (ibid). While the results obtained are commendable, they considered each joke as a black box and did not take the text of the joke into consideration.

3.3 Proposed Model

The idea behind the proposed model was to create various clusters each consisting of similar jokes. This way, after identifying the user's favorite joke, the jokes which share the same cluster with it could be selected for recommendation. In order to achieve this, we had to quantitatively find out the similarity between any given pair of jokes. This can be visualized with the help of the following flow chart:

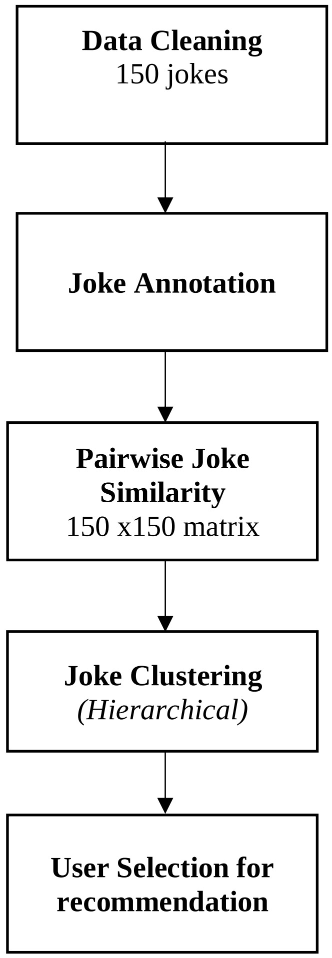

Figure 6: Proposed Methodology Flowchart

The text of the jokes is analyzed or preprocessed with the help of the annotated features of the jokes. The methodology is explained step by step below:

Step 1 (Data Cleaning): The corpus required cleaning since there were two pair of jokes that were repeated so we deleted the duplicate joke. After this set, our dataset had 148 jokes.

Step 2 (Joke Annotation): We initialize the jokes with the respective knowledge resources. In this research, we focused on SO, LM, SI and TA, as LA value should differ for every joke and most jokes in the dataset have the same NS value.

Step 3 (Pairwise Joke Similarity): We calculated the similarity between the jokes by comparing the knowledge resources for each pair of jokes using the Lin Similarity Metric.

Step 3.a. Whenever the knowledge resource is the same for the pair the similarity is 1. If any knowledge resource is missing for a joke, then it is considered as null. The following formula describes this:

$$\text{Similarity } (kr_a, kr_b) = \begin{cases} 1 & if\ kr_a = kr_b \\ 0 & if\ kr_a\ or\ kr_b is\ null \\ sim_{Lin}\,(kr_a, kr_b) & all\ other\ cases \end{cases}$$

where k_a and k_b are instances of the same knowledge resources. The only exception will be made for the similarity of targets (TA), where word2vec similarity will be used.

Step 3.b. If annotation for SO is missing for any joke, then it should be discarded because without SO a text cannot be classified as a joke. We came across 6 jokes that did not have SO annotations so after deleting those from the dataset we were left with 142 jokes.

Step 3.c: To take into consideration the hierarchy of KRs themselves, as proposed by SSTH, we assign a weight, w_{SO}, w_{LM}, w_{SI}, and w_{TA}, to each of the KRs such that $w_{SO} < w_{LM} < w_{SI} < w_{TA}$.

$$\text{sim}(\text{joke}_i, \text{joke}_j) = \frac{[w_{SO} \quad w_{LM} \quad w_{SI} \quad w_{TA}] \begin{bmatrix} \text{sim}(\text{SO}_{\text{joke}_i}, \text{SO}_{\text{joke}_j}) \\ \text{sim}(\text{LM}_{\text{joke}_i}, \text{LM}_{\text{joke}_j}) \\ \text{sim}(\text{SI}_{\text{joke}_i}, \text{SI}_{\text{joke}_j}) \\ \text{sim}\ (\text{TA}_{\text{joke}_i}, \text{TA}_{\text{joke}_j}) \end{bmatrix}}{w_{SO} + w_{LM} + w_{SI} + w_{TA}}$$

For this initial investigation, the following values are assigned: w_{SO}=5, w_{LM}=4, w_{SI}=3 and w_{TA}=2.

$$\frac{\mathrm{sim}(\mathrm{joke_i}, \mathrm{joke_j}) = [5\ 4\ 3\ 2]\begin{bmatrix} \mathrm{sim}(\mathrm{SO}_{\mathrm{joke_i}}, \mathrm{SO}_{\mathrm{joke_j}}) \\ \mathrm{sim}(\mathrm{LM}_{\mathrm{joke_i}}, \mathrm{LM}_{\mathrm{joke_j}}) \\ \mathrm{sim}(\mathrm{SI}_{\mathrm{joke_i}}, \mathrm{SI}_{\mathrm{joke_j}}) \\ \mathrm{sim}\ (\mathrm{TA}_{\mathrm{joke_i}}, \mathrm{TA}_{\mathrm{joke_j}}) \end{bmatrix}}{5+4+3+2}$$

Step 4 (Joke Clustering): Cluster jokes based on their similarity calculated using a clustering technique. We implemented hierarchical clustering of jokes based using the similarity matrix created in Step 3.

Step 5 (User Selection): For recommendation, the users are selected randomly, and their favorite joke is identified by looking at all the ratings given by this user. Then the cluster which contains the chosen user's favorite joke is selected and the jokes which fall in the same cluster are recommended. We analyzed the recommended jokes with the help of the actual ratings whenever they were available.

The calculation of the Lin similarity and word2vec similarity for the jokes 6 and 7 is explained in detail in the following sections.

3.3.1 Lin Similarity Metric

An information-content word-similarity algorithm is adapted to calculate the similarity between the jokes. This algorithm keeps the structure of the thesaurus intact and also add the probabilistic information from a corpus. The probability that a random word selected in a corpus is an instance of concept c is defined by Resnik as follows (Resnik, 1995) (Jurafsky & Martin, 2018)

$$P(c) = \frac{\sum_{w \in words(c)} count(w)}{N}$$

where words(Meyer) is the set of words subsumed by concept c and N represents the total number of words in the corpus. The formula implies that P(*root*)=1 since all the words always subsumed by the root. The lower a word in the hierarchy, the lower will be its probability. The similarity between two words is related to their common information which implies that the more two words have in common, the more similar they are. Resnik (1995) proposed one of the first ways to estimate the common amount of information by the information content of the LCS of the two nodes. Lin (1998) extended this concept by adding that the similarity metric must also take into account the differences between the two words. The Lin’s similarity measure (Lin, 1998), adapted from Jurafsky and Martin (2018) used for word similarity is:

$$sim_{Lin}(c_1, c_2) = \frac{2 * logP(LCS(c_1, c_2))}{logP(c_1) + logP(c_2)}$$

where LCS (c_1, c_2) represents the lowest node in the hierarchy that subsumes both c1 and c2. A high value of Lin similarity between a pair of jokes for each knowledge resource would indicate that the pair is very similar. According to the formula, if two instances of the same knowledge resource have the root as the common ancestor then the similarity value will be 0. Similarity metric for all the jokes follows a similar concept and an example given in the end of the section.

3.3.2 Word2Vec Similarity

It is not possible for many machine learning algorithms to process strings or plain text in its raw form hence the word embeddings came to the rescue. Word embeddings are a class of techniques in which the words are represented as vectors in a predefined vector space. It allows different algorithms to recognize words with similar meanings. One such word embedding model is word2vec. Word2vec (Mikolov, Chen, Corrado, & Dean, 2013 & Dean (2013) is which is a shallow neural network that takes a text corpus as input and returns the vector representations of the words. For our research, we used Google's pre-trained model which has word vectors for 3 million words which were obtained by training on a google news dataset of around 100 billion words. We used the Gensim library (version 3.8.1) as it provides the tools for loading these pretrained word-embeddings and for querying the embeddings which helped us in finding the similarity between the TA instances of the jokes.

But there were some annotations in our dataset which were not present in the text corpus over which word2vec was trained, hence their word embeddings were missing. For this research, we made appropriate replacements of those words by looking at the jokes and ensuring that the context of the joke didn't change. The following table shows the replacements made:

Table 3: TA replacements

TA annotation	Replacement
Working class	Rich
GW Bush	Politics
Men that are perceived to be smart	Men
Russian Parliament	Politics
Soviet Union	Soviet
Bad student	Student
Jewish Mothers	Mothers
Jewish Wife	Jewish
Old Men	Elderly
OJ Simpsons	Criminal
Stanford Graduate	Graduate
Liberal Arts Graduate	Graduate
Chuck Norris	Superhuman
Scots and Canadian	Canadian
Hillary Clinton	Politics
US President	Politics
American Sport	Football
Gay/Lesbian	Homosexual

3.3.3 Hierarchical Clustering

In the baseline mode, the researchers adopted the rectangular recursive clustering technique since their data was centered around origin. We did not face the same issue since we had a matrix as our data for clustering. Our aim behind clustering was to club the jokes together based on the values in the similarity matrix. We explored Hierarchical clustering and K-means since they are two very famous cluster algorithms.

We adopted Hierarchical Clustering because of many reasons. First it was difficult to decide the number of clusters (k) which has to be predetermined in the K-means clustering whereas in Hierarchical clustering there is no such requirement. Since the latter assigns each joke to a singleton cluster and then successively merges them until all jokes have been merged into a single remaining cluster, we did not have to find the magic “k”. Hierarchical clustering is visualized using a dendrogram which made the visualization of the clusters easy. Secondly, K-means has the advantage of being computationally faster than hierarchical clustering but in our case, since the dataset of jokes was not very large, the computational time taken by both the techniques was roughly the same. With the change in the initial centroids, the clusters changed in K-means making it sensitive to the initial centers (Celebi, Kingravi, & Vela, 2013) whereas, in Hierarchical, the most similar jokes were always clustered first. So, the latter gave consistent results. To recommend the jokes to the user, in the K-means technique we had to randomly select the jokes from the chosen cluster. We could eliminate this random selection in Hierarchical clustering as it provided a hierarchy between the jokes which made the recommendation easy for this research.

To perform hierarchical clustering, we need the following parameters: distance function and linkage criteria. A distance function represents the pairwise distance between the jokes and since we used a similarity function as described in section 4.3, we had to convert it into a distance function for the clustering algorithm. To do this conversion, we implemented the logic that a high value of similarity between the jokes implied that they have a low value of distance between them because similar jokes should be clustered together. We implemented the average linkage method which considers the average distance between every point in the cluster to every point in another cluster.

3.3.4 Workflow Example

We will walk through the steps outlined in the flowchart with the help of the following examples taken from our dataset:

Joke$_6$: A guy walks into a bar, orders a beer and says to the bartender, "Hey, I got this great Polish Joke." The barkeep glares at him and says in a warning tone of voice: "Before you go telling that joke you better know that I'm Polish, both bouncers are Polish and so are most of my customers". "Okay" says the customer, "I'll tell it very slowly." (Jester dataset)

Joke$_7$: The graduate with a Science degree asks, "Why does it work?" The graduate with an Engineering degree asks, "How does it work?" The graduate with an Accounting degree Asks, "How much will it cost?" The graduate with a Liberal Arts degree asks, "Do you want fries with that?" (Jester dataset)

Step 1 (Data Cleaning): In this step, we ensured that all the jokes were unique in the corpus.

Step 2 (Joke Annotation): For jokes 6 and 7, the annotations are:

$$\begin{matrix}\textbf{SO}\\ \textbf{LM}\\ \textbf{SI}\\ \textbf{TA}\end{matrix}\begin{Bmatrix}\text{actual/non}-\text{actual}\\ \text{faulty reasoning}\\ \text{going to bar}\\ \text{poles}\end{Bmatrix}\begin{Bmatrix}\text{actual/non}-\text{actual}\\ \text{faulty reasoning}\\ \text{intellectual discussion}\\ \text{graduates}^{2}\end{Bmatrix}$$

Step 3 (Pairwise Joke Similarity): We apply the above defined similarity formula over the annotations of jokes 6 and 7 by following the steps 3.a and 3.b to get the following matrix. Since the jokes have the same annotations for SO and LM, the similarity value is 1.

$$\frac{[5\ 4\ 3\ 2]\begin{bmatrix}1\\ 1\\ sim_{Lin}\left(kr_{going\ to\ a\ bar}, kr_{intellectual\ discussion}\right)\\ word2vec\ similarity_{(Poles,graduates)}\end{bmatrix}}{5+4+3+2}$$

A fragment of the hierarchy for the SI knowledge resource can be represented as follows:

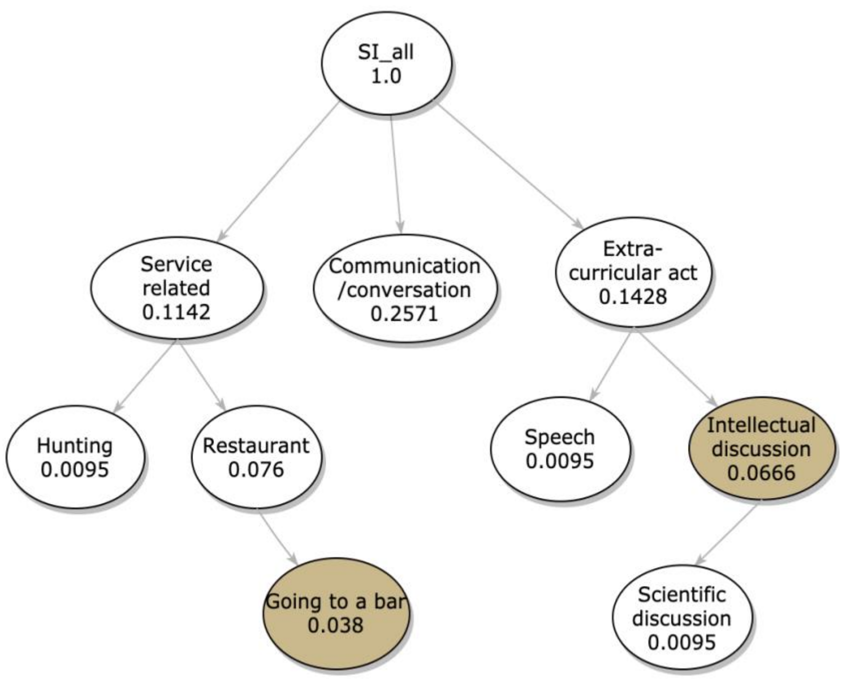

Figure 7: A fragment of the SI hierarchy

[2] Annotation has been changed from *liberal arts graduate* to *graduates* since the former was not in word2vec-based-model

Using the above formula, the Lin similarity value in Step 3.c for the jokes 6 and 7 is calculated as follows:

$$sim_{Lin}\left(SI_{going\ to\ a\ bar}, SI_{intellectual\ discussion}\right) = \frac{2 * log\ P(LCS(going\ to\ a\ bar,\ intellectual\ discussion)}{logP(going\ to\ a\ bar) + log\ P(\ intellectual\ discussion)}$$

$$= \frac{2 * log(\,1)}{log(0.038) + log(\,0.0666)} = 0$$

From the Figure 7, we can see that the LCS of both the SI annotations is the root node, the probability value of the LCS is taken as 1. The matrix is now updated as follows:

$$\frac{[5\ 4\ 3\ 2]\begin{bmatrix} 1 \\ 1 \\ 0 \\ word2vec\ similarity_{(poles,graduates)} \end{bmatrix}}{5+4+3+2}$$

We can now use word2vec to calculate the similarity between *poles* and *graduates.*

$$\frac{[5\ 4\ 3\ 2\ ]\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0.046 \end{bmatrix}}{5+4+3+2} = \frac{0.092}{15} = 0.649$$

The value calculated after the weighted average is 0.649 which quantifies how similar $Joke_6$ and $Joke_7$ are. Likewise, we populate a 142x 142 matrix by calculating the pairwise similarity between all the jokes. A high value in this matrix represents a high degree of similarity according to GTVH.

CHAPTER 4. RECOMMENDATION COMPARISON

Since the aim of the recommendation had to be personalized for every user, we decided to compare both the baseline and the proposed model at the user level.

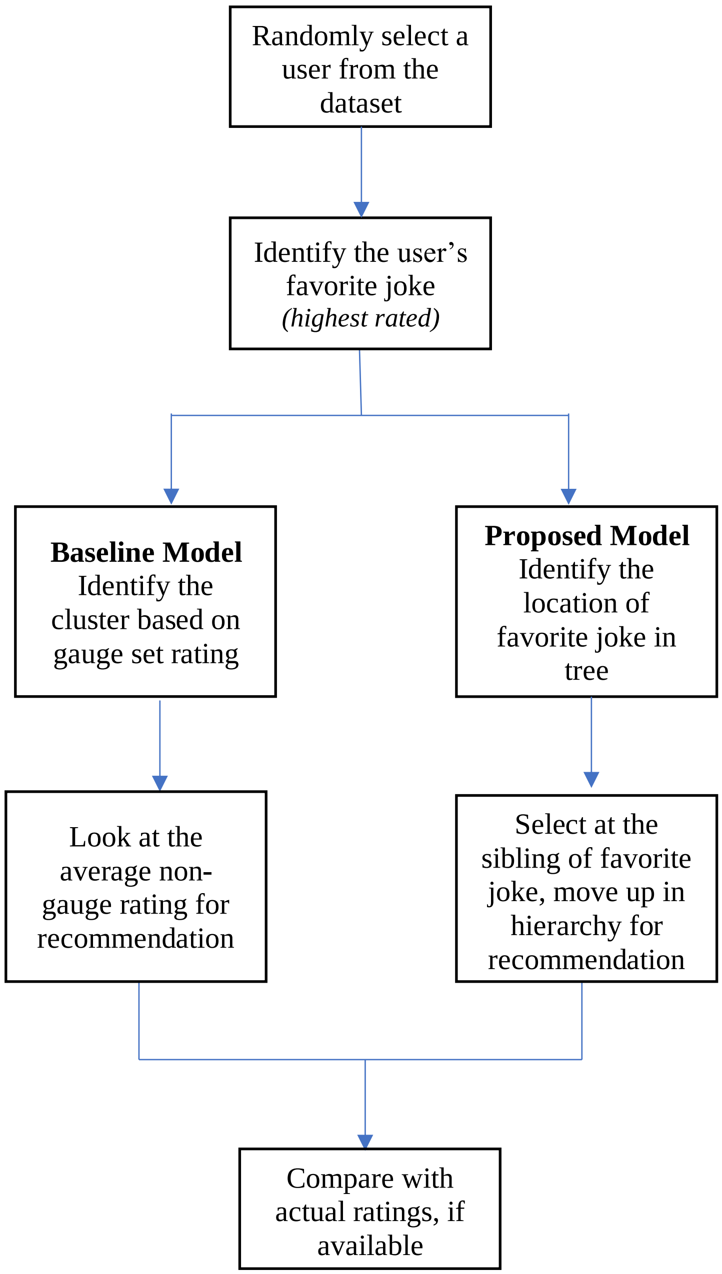

Figure 8: Comparison Flowchart

It is worth noting that while recommending jokes using the proposed model, if there are multiple jokes available on the same level of the hierarchy, then a random selection of the jokes is done for that level.

Due to the sparsity of the rating dataset, we are constrained in selecting the users which sheds light on one of the difficulties with working with this dataset. As a result, we meticulously select and report results of the users who have rated the jokes in both the baseline and the GTVH, to ensure that the comparison of the two models is possible. The results for randomly selected 5 users are shown analyzed in Table 4 where we also present the highest-rated joke for all 5 users along with recommended jokes by the baseline and the GTVH-based mode.

Table 4: Comparison of Recommended Jokes

	Baseline Model		**GTVH-based Model**	
	Top Recommended Joke	**Rating**	**Top Recommended Joke**	**Rating**
User 1	Joke 89	8.18	Joke 87	9.37
User 2	Joke 73	-1	Joke 42	5.71
User 3	Joke 53	3.56	Joke 72	3.46
User 4	Joke 5	9.87	Joke 112	0.93
User 5	Joke 89	4.56	Joke 126	5.62

For user 1 in Table 2, we observe that the top joke recommended by the GTVH-based model (Joke 87) has a better rating than the top joke recommended by the baseline model (Joke 89). We analyze the top recommended jokes by both the models along for User 1 along with the favorite joke of the same user. We can clearly see by the following text of the jokes that the favorite joke of user 1 and Joke 87 are very similar whereas Joke 89 is very different from these jokes.

User 1’s Favorite joke*: An artist asked the gallery owner if there had been any interest in his paintings currently on display. "I've got good news and bad news," the owner replied. "The good* *news is that a gentleman inquired about your work and wondered if it would appreciate after your death. When I told him it would, he bought all fifteen of your paintings." "That's wonderful!" the artist exclaimed. "What's the bad news?" With concern, the gallery owner replied: "The guy was your doctor."* (Jester dataset)

Joke87: *A man recently completing a routine physical examination receives a phone call from his doctor. The doctor says, "I have some good news and some bad news." The man says, "OK, give me the good news first." The doctor says, "The good news is, you have 24 hours to live." The man replies, "Shit! That's the good news? Then what's the bad news?" The doctor says, "The bad news is, I forgot to call you yesterday."* (Jester dataset)

Joke89: *A radio conversation between a US naval ship and Canadian authorities... Americans: Please divert your course 15 degrees to the North to avoid a collision. Canadians: Recommend you divert YOUR course 15 degrees to the South to avoid a collision. Americans: This is the captain of a US Navy ship. I say again, divert YOUR course. Canadians: No. I say again, you divert YOUR course. Americans: This is the aircraft carrier USS Lincoln, the second largest ship in the United States; Atlantic Fleet. We are accompanied by three destroyers, three cruisers and numerous support vessels. I demand that you change your course 15 degrees north, that's ONE FIVE DEGREES NORTH, or countermeasures will be undertaken to ensure the safety of this ship. Canadians: This is a lighthouse. Your call.* (Jester dataset)

From the Table 4 we can see that the proposed model works better in comparison to the baseline for users 1, 2 and 5, works moderately well for user 3, and fails to perform better for users 4. We want to point out that it is equally possible to find similar jokes to all highly rated jokes for any selected user. Nevertheless, based on the results of user 4, we wanted to investigate if highly similar jokes are typically rated similarly or not.

To explore how different users rate jokes that are considered similar by the proposed model, we selected 5 users who have rated 140 jokes which the maximum number of jokes rated by any user. Also, we normalized the ratings to 0-5 for the experiment. We selected all the joke clusters which are formed near the distance value of 0.2 for the analysis. Figure 9 shows the graph we created to do the analysis for user 6. All the 7 clusters formed created below the dotted reference line were analyzed and their corresponding user ratings are also highlighted in the graph.

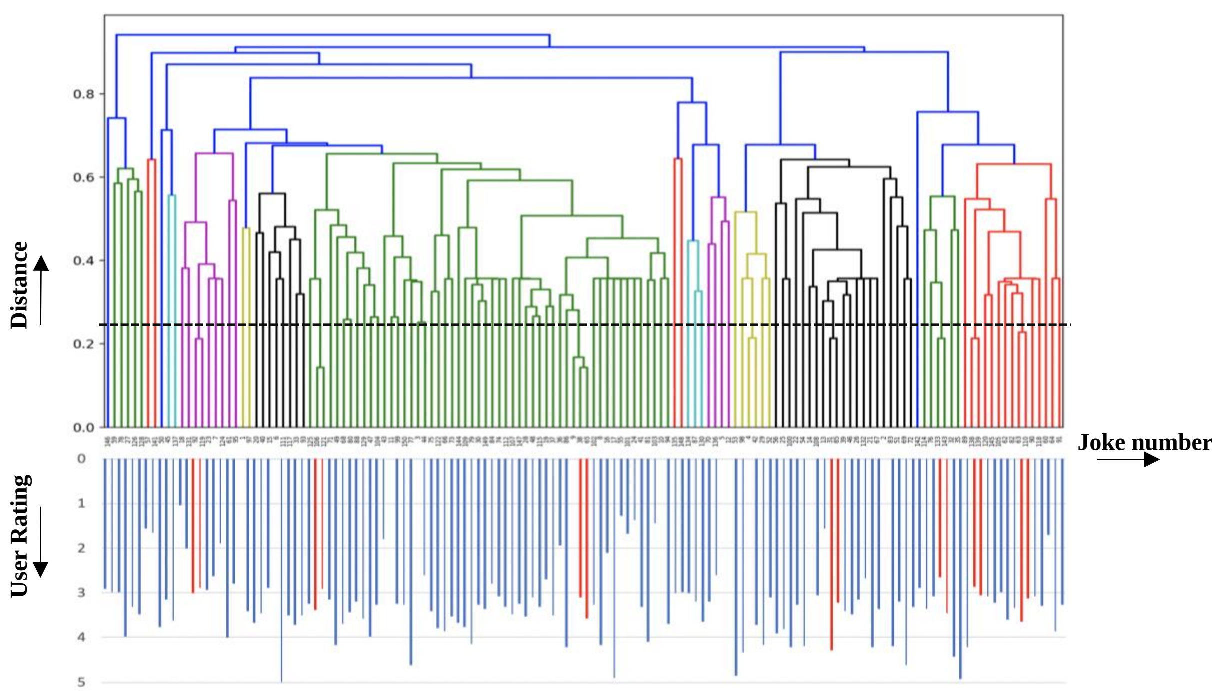

Figure 9: Rating vs Joke clustering for user 6

From the graph, we can see that the ratings given by this user to the jokes within the same cluster are quite similar. The maximum difference in rating is of 1.07 points for the cluster 4 which consists of jokes 31 and 85. Since the ratings are quite similar across all the clusters, we can say that the jokes that are considered similar according to the GTVH-based model are liked by the user 6 to the same extend. But to make a generalized statement, we need to look at the results of other users as well. Below is the graph for user 7.

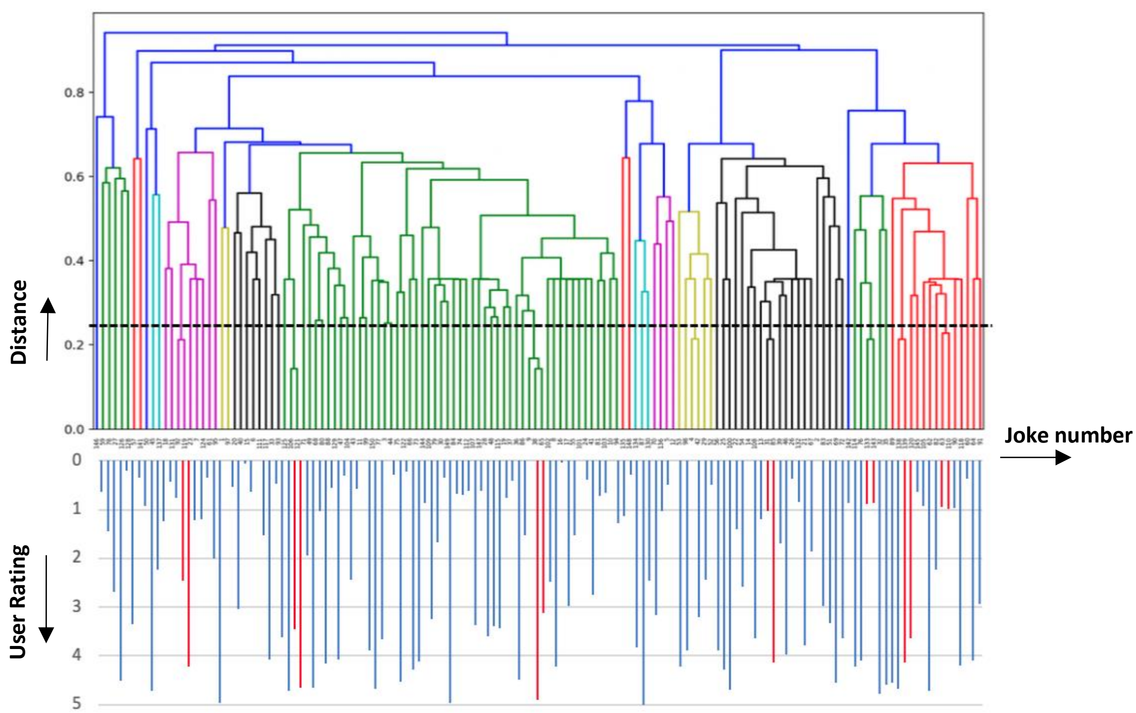

Figure 10: Rating vs Joke clustering for user 7

From the above graph, we observe that the ratings given by user 7 differ greatly for the jokes clustered together. The difference in ratings is more than 1.5 points for the jokes in the clusters 1, 3 and 4 whereas this difference is not significantly large for the remaining 4 clusters. It is interesting to note that the difference in ratings to the jokes in cluster 4 is highest for both the users 6 and 7 which hints at the possibility that these jokes are incorrectly clustered together which could be fixed by manipulation of the KRs. To make further analysis we look at the graph in figure 11 which shows the rating for user 8.

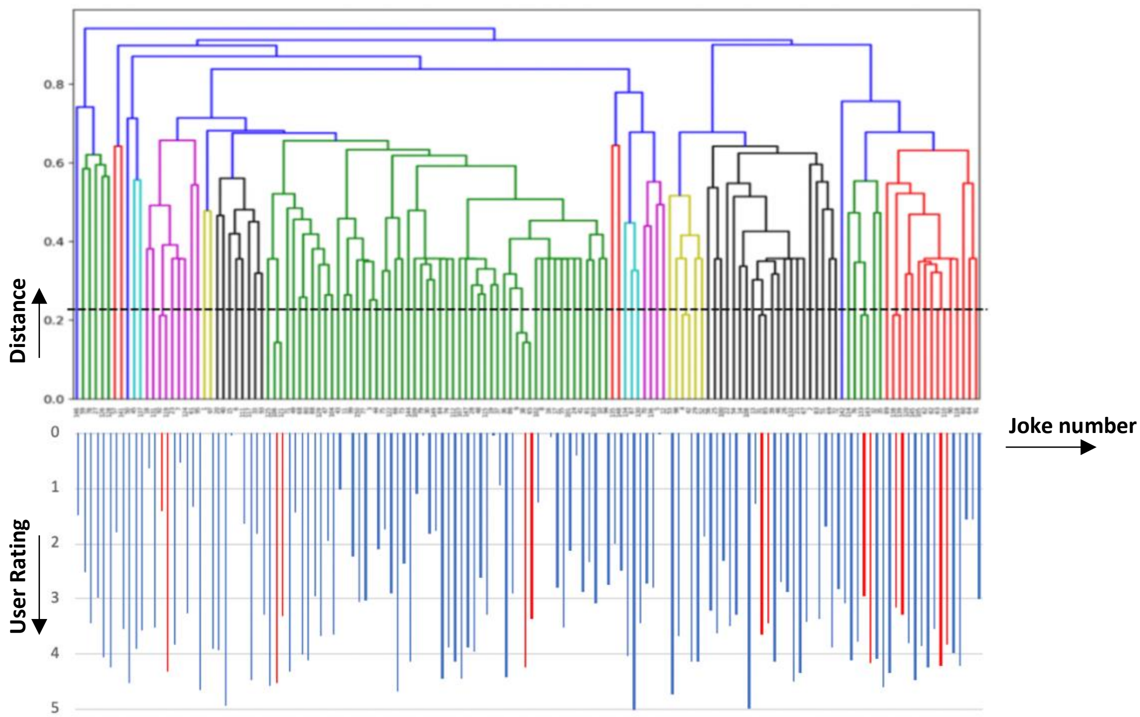

Figure 11: Rating vs Joke clustering for user 8

We can see in figure 11 that user 8 rated jokes within the same cluster similarly to a large extent except for cluster 1. While user 8 rated jokes in cluster 4 quite similarly, the ratings given to jokes in cluster 1 were very different whereas both users 6 and 7 rated jokes in cluster 1 similarly. So, we can make the same conclusion for user 6 and user 8 that for both these users the similarity matrix that we calculated works well since both the users like the closely clustered jokes to the same degree.

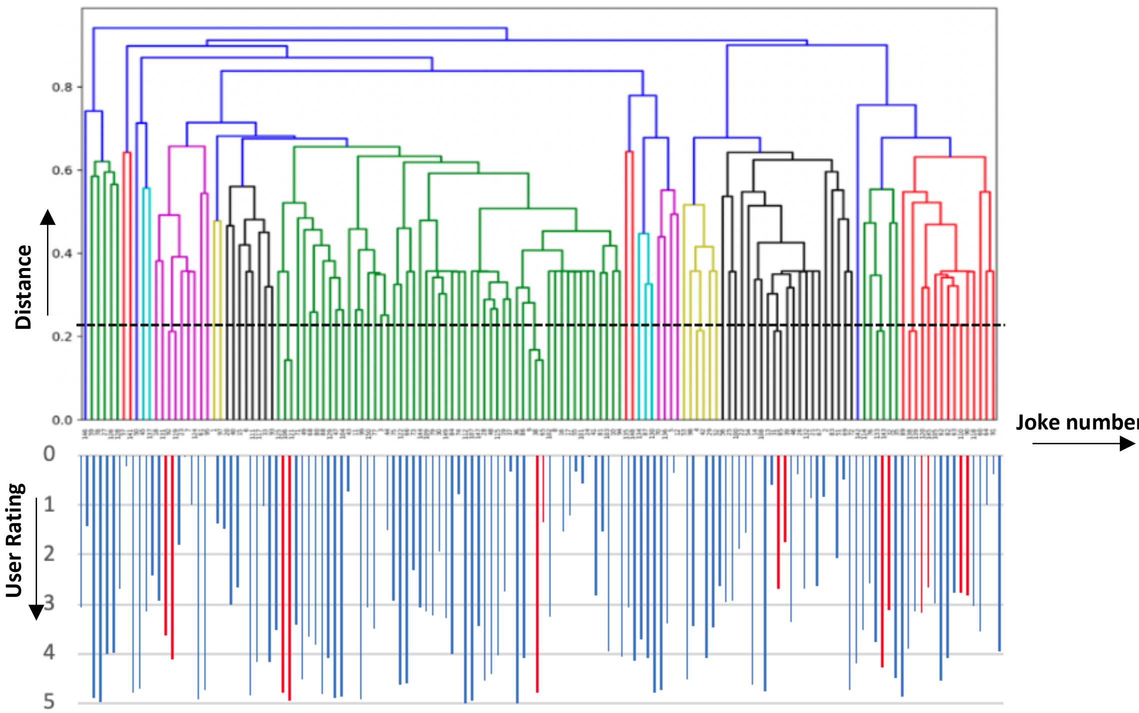

Figure 12: Rating vs Joke clustering for user 9

From the above graph in figure 12, we observe that the user 9 liked the jokes within the same cluster to the same extent except for cluster 3 in which the jokes have a huge difference of 3.44 points. Other than cluster 3, the ratings of user 9 are fairly similar so we can again conclude that jokes regarded similar by the GTVH-based model are equally appreciated by user 6, 8 and 9 whereas the same cannot be concluded for user 7. Lastly, the results for user 10 are shown in figure 13.

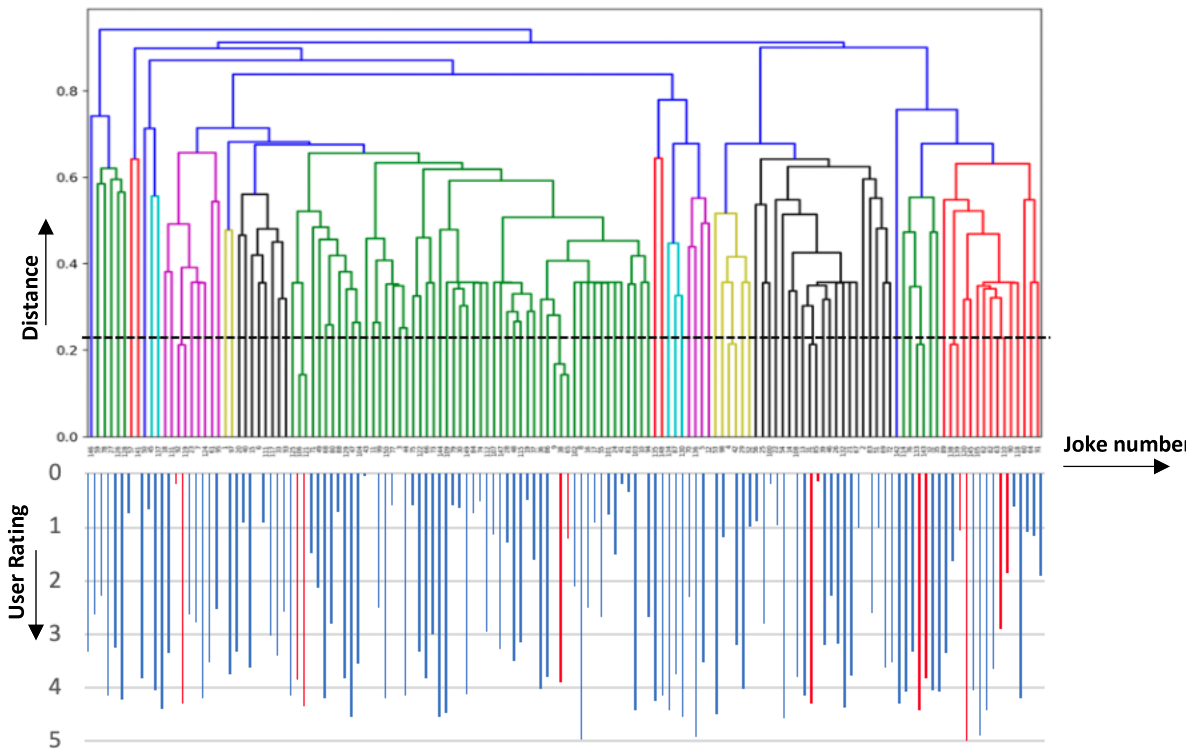

Figure 13: Rating vs Joke clustering for user 10

Looking at the graph in figure 13, we can see that the user 10 had given very different ratings to the jokes within the same cluster. The difference in ratings given by user 10 is more than 1.5 points for the jokes in clusters 1, 3 and 4. It is interesting to note that we obtained analogous results for user 7 as well since both these users rate the jokes in clusters 1, 3 and 4 very contrarily to the results of other users.

The findings from all these 5 graphs are summarized in Table 5 which lists 7 clusters each consisting of 2 jokes for the comparison of user ratings of closely clustered jokes. The values that we want to draw attention to in the table are italicized to represent that the joke pairs had more than 1.5 point's difference in the user ratings. The summary of the observations are that the intra-cluster ratings of the users 6, 8 and 9 are largely similar for all the clusters while they greatly differ for users 7 and 10. Intriguingly, both the users 7 and 10 rate jokes in clusters 1, 3 and 4 were differently which brings us to the inference that the jokes which are considered similar by the GTVH-based model are not equally appreciated by both these users.

Table 5: Cluster Analysis for the five selected users

	Cluster 1		**Cluster 2**		**Cluster 3**		**Cluster 4**		**Cluster 5**		**Cluster 6**		**Cluster 7**	
Joke Id	**92**	**119**	**106**	**121**	**38**	**65**	**31**	**85**	**133**	**143**	**138**	**139**	**110**	**63**
User 6	3.02	2.88	3.39	2.91	3.10	3.58	4.29	3.22	2.66	3.46	2.86	3.05	3.13	3.64
User 7	*2.45*	*4.22*	3.45	4.66	*4.91*	*3.13*	*1.02*	*4.14*	0.88	0.87	4.68	4.14	0.99	0.95
User 8	*1.40*	*4.31*	4.53	3.30	4.25	3.36	3.65	3.45	2.94	4.17	3.16	3.29	3.83	4.22
User 9	3.63	4.12	4.80	4.96	*4.79*	*1.35*	2.68	1.75	4.28	3.13	3.13	3.19	2.82	2.77
User 10	*0.19*	*4.31*	3.85	4.34	*3.90*	*1.21*	*4.31*	*0.13*	4.41	3.81	1.64	1.07	1.85	2.90

There are numerous probable explanations for this result, if we assume that the ratings in the dataset accurately represent user preferences of the jokes. The first explanation is that the similarity metric that we presented in this research does not correctly represent joke similarity, and the knowledge resource TA may need to be weighted heavier than the rest of the KRs for the recommendation system. The second one is that the annotation of one of the jokes in the clusters may be flawed due to which the analysis shows anomalous results for some users. The third, and perhaps most interesting one, is whether users tend to rate familiar jokes lower. Since we lack the data on the ordering of jokes that were presented to the users, it is not possible to test this hypothesis. However, we have two pairs of almost identical jokes in the dataset and we can look at rating for these users which is shown in Table 6.

Table 6: Ratings given to Identical Jokes

	Identical Pair 1		**Identical Pair 2**	
User 6	5.12	0.62	3.15	1.37
User 7	*-5.25*	*4.68*	-7.31	-5.84
User 8	5.81	0.62	*9.62*	*1.68*
User 9	5.12	5.59	3.53	7.28
User 10	4.78	0.53	1.15	5.34

From the above table, we see that users 7, 8 and 9 have given different ratings to identical jokes. We do understand that some variation in the ratings should be tolerable since the users were given a scroll button to rate the jokes, but this difference is very high for users 7 and 8 as italicized in the table. It is tempting to conclude that the effect of a previously heard or rated joke must be taken into consideration while recommendations are made. It is also possible that for some of the users the almost identical jokes were presented very close to each other, while for others they were spread much farther apart among the 140 jokes. Lastly, the dataset also does not consider the effect fatigue effects of the users which may affect the ratings.

CHAPTER 5. CONCLUSION

In this exploratory study, we proposed a recommendation system that utilized the similarity between various knowledge resources of all the jokes in the dataset to make personalized recommendations. We extracted the similarity between the jokes using Lin's similarity metric and word2vec and assigning weights to the KRs to preserve the hierarchy proposed by SSTH. We further clustered the jokes based on these similarity values in a hierarchical manner. To the best of our knowledge, this is the first study that extends the concept of a recommendation system of jokes towards taking into account the text of the jokes as well. In order to compare results of this work with the existing recommendation system, we implemented Jester. The recommendations made by both the models were evaluated by randomly selecting users from the dataset and comparing their actual ratings which were considered as ground truth.

There are some suggestions and ideas that can be incorporated for improving the proposed model. The recommendation for future research about this study is to manipulate the weights assigned to KRs, probably weighting TA heavier than the rest of the KRs for the recommendation system. Also, the model's performance can improve if the effect of a previously heard joke is taken into consideration along with the order the jokes are presented to the users while they are rating them. We believe that getting a better understanding of user preferences will make the interaction between users and various devices that can tell jokes to the users, more friendlily.

REFERENCES

Adomavicius, G., & Tuzhilin, A. (2005). Personalization technologies: a process-oriented perspective. *Communications of the ACM, 48*(10), 83-90. doi:10.1145/1089107.1089109

Attardo, S. (1994). Linguistic theories of humor (Vol. 1). *Berlin: De Gruyter Mouton. Barsoux, JL (1996). Why organisations need humour. European Management Journal, 14*(5), 500-508.

Attardo, S., & Raskin, V. (1991). Script theory revis (it) ed: Joke similarity and joke representation model. *Humor-International Journal of Humor Research, 4*(3-4), 293-348.

Attardo, S., & Raskin, V. (2017). Linguistics and humor theory. In *The Routledge handbook of language and humor* (pp. 49-63): Routledge.

Beutel, A., Covington, P., Jain, S., Xu, C., Li, J., Gatto, V., & Chi, E. (2018). Proceedings of the Eleventh ACM International Conference on web search and data mining. In (Vol. 2018-, pp. 46-54): ACM.

Binsted, K. (1995). *Using humour to make natural language interfaces more friendly.* Paper presented at the Proceedings of the AI, ALife and Entertainment Workshop, Intern. Joint Conf. on Artificial Intelligence.

Binsted, K., Nijholt, A., Stock, O., Strapparava, C., Ritchie, G., Manurung, R., . . . O'Mara, D. (2006). Computational humor. *IEEE Intelligent Systems, 21*(2), 59-69.

Cai, Y., Leung, H.-f., & Li, Q. (2014). Huaqing Min, Jie Tang, Juanzi Li, Typicality-Based Collaborative Filtering Recommendation. *IEEE Transactions on Knowledge and Data Engineering, 26*(3), 766-779.

Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems With Applications, 40*(1), 200-210. doi:10.1016/j.eswa.2012.07.021

Choi, S.-M., Ko, S.-K., & Han, Y.-S. (2012). A movie recommendation algorithm based on genre correlations. *Expert Systems with Applications, 39*(9), 8079-8085.

Cleger-Tamayo, S., Fernandez-Luna, J. M., & Huete, J. F. (2012). Top-N news recommendations in digital newspapers.(Report). *Knowledge-Based Systems, 27*, 180.

Das, A. S., Datar, M., Garg, A., & Rajaram, S. (2007). *Google news personalization: scalable online collaborative filtering.* Paper presented at the Proceedings of the 16th international conference on World Wide Web.

Francis, L., Monahan, K., & Berger, C. (1999). A laughing matter? The uses of humor in medical interactions. *Motivation and emotion, 23*(2), 155-174.

Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *information retrieval, 4*(2), 133-151.

Guarino, N., Oberle, D., & Staab, S. (2009). What is an ontology? In *Handbook on ontologies* (pp. 1-17): Springer.

Hempelmann, C. F., Taylor, J. M., & Raskin, V. (2010). *Application-guided ontological engineering.* Paper presented at the ICAI 2010: proceedings of the 2010 international conference on artificial intelligence (Las Vegas NV, July 12-15, 2010).

Hetzron, R. (1991). On the structure of punchlines. *HUMOR: International Journal of Humor Research*.

Jurafsky, D., & Martin, J. H. (2018). Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. *Inc. Upper Saddle River, NJ*.

Kant, I. (1911). Critique of Judgment, James Creed Meredith (tr.). In: Oxford: Clarendon Press.

Koren, Y. (2009). The bellkor solution to the netflix grand prize. *Netflix prize documentation, 81*(2009), 1-10.

Larkin-Galiñanes, C. (2017). An overview of humor theory. In *The Routledge Handbook of Language and Humor* (pp. 4-16): Routledge.

Leino, J. (2014). *User Factors in Recommender Systems: Case Studies in e-Commerce, News Recommending, and e-Learning*.

Lewith, G. T., Jonas, W. B., & Walach, H. (2011). *Clinical research in complementary therapies principles, problems and solutions* (2nd ed. ed.). Edinburgh ; New York: Churchill Livingstone.

Li, Q., Myaeng, S. H., & Kim, B. M. (2007). A probabilistic music recommender considering user opinions and audio features. *Information Processing and Management, 43*(2), 473-487. doi:10.1016/j.ipm.2006.07.005

Lin, D. (1998). *An information-theoretic definition of similarity.* Paper presented at the Icml.

Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*(1), 76-80.

Lurie, A., & Monahan, K. (2015). Humor, aging, and life review: Survival through the use of humor. *Social Work in Mental Health, 13*(1), 82-91.

McGhee, P. E., & Pistolesi, E. (1979). *Humor: Its origin and development*: WH Freeman San Francisco.

McKay, J. (2002). Generation of idiom-based witticisms to aid second language learning. *Stock et al*, 77-87.

Meyer, J. C. (2000). Humor as a double-edged sword: Four functions of humor in communication. *Communication theory, 10*(3), 310-331.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *arXiv.org*.

Mohamed, M. H., Khafagy, M. H., & Ibrahim, M. H. (2019). 2019 International Conference on Innovative Trends in Computer Engineering (ITCE). In (pp. 149-155): IEEE.

Morkes, J., Kernal, H. K., & Nass, C. (1998). *Humor in task-oriented computer-mediated communication and human-computer interaction.* Paper presented at the CHI 98 Conference Summary on Human Factors in Computing Systems.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2*(11), 559-572. doi:10.1080/14786440109462720

Raskin, V. (1985). *Semantic mechanisms of humor*. Dordrecht ; Boston: D. Reidel Pub. Co.

Raskin, V., Hempelmann, C. F., & Taylor, J. M. (2009). How to understand and assess a theory: The evolution of the SSTH into the GTVH and now into the OSTH. *Journal of Literary Theory, 3*(2), 285-311.

Raskin, V., Hempelmann, C. F., & Taylor, J. M. (2010). *Guessing vs. knowing: The two approaches to semantics in natural language processing*. Paper presented at the Annual International Conference Dialogue 2010.

Ravi, L., & Vairavasundaram, S. (2016). A Collaborative Location Based Travel Recommendation System through Enhanced Rating Prediction for the Group of Users. *Computational Intelligence and Neuroscience, 2016*(2016). doi:10.1155/2016/1291358

Rayz, J. T. (2020). Scripts in the ontological semantic theory of humor. *Script-Based Semantics: Foundations and Applications. Essays in Honor of Victor Raskin*, 229.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.

Ritchie, G. (1999). *Developing the incongruity-resolution theory*. Retrieved from

Ritchie, G. (2001). Current directions in computational humour. *Artificial Intelligence Review, 16*(2), 119-135.

Rothbart, M. K., & Pien, D. (1977). *Elephants and Marshmallows: A Theoretical Synthesis of Incongruity-Resolution and Arousal Theories of Humour*.

Ruch, W., Attardo, S., & Raskin, V. (1993). Toward an empirical verification of the general theory of verbal humor. *Humor, 6*(2), 123-136.

Ruch, W., & Heintz, S. (2016). The virtue gap in humor: Exploring benevolent and corrective humor. *Translational Issues in Psychological Science, 2*(1), 35.

Schopenhauer, A. (1907). The World as Will and Idea, trans. RB Haldane and John Kemp. *London: Routledge and Kegan Paul, 1909*, 52.

Schultz, T. R. (1976). A cognitive-developmental analysis of humour. In (pp. 12-13): New York: Wiley.

Shani, G., & Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender systems handbook* (pp. 257-297): Springer.

Stock, O. (1996). *Password Swordsfish: Verbal Humor in the Interface*: Citeseer.

Suls, J. M. (1972). A two-stage model for the appreciation of jokes and cartoons: An information-processing analysis. *The psychology of humor: Theoretical perspectives and empirical issues, 1*, 81-100.

Taylor, J. M., Raskin, V., Hempelmann, C. F., & Attardo, S. (2010). *An unintentional inference and ontological property defaults.* Paper presented at the 2010 IEEE International Conference on Systems, Man and Cybernetics.

Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W., & Leskovec, J. (2018). Proceedings of the 24th ACM SIGKDD International Conference on knowledge discovery & data mining. In (pp. 974-983): ACM.