applied sciences

MDPI

Article

Improving User Intent Detection in Urdu Web Queries with Capsule Net Architectures

Sana Shams * and Muhammad Aslam

Department of Computer Science, University of Engineering and Technology, Lahore, Pakistan
* Correspondence: sana.shams@kics.edu.pk

Abstract: Detecting the communicative intent behind user queries is critically required by search engines to understand a user's search goal and retrieve the desired results. Due to increased web searching in local languages, there is an emerging need to support the language understanding for languages other than English. This article presents a distinctive, capsule neural network architecture for intent detection from search queries in Urdu, a widely spoken South Asian language. The proposed two-tiered capsule network utilizes LSTM cells and an iterative routing mechanism between the capsules to effectively discriminate diversely expressed search intents. Since no Urdu queries dataset is available, a benchmark intent-annotated dataset of 11,751 queries was developed, incorporating 11 query domains and annotated with Broder's intent taxonomy (i.e., navigational, transactional and informational intents). Through rigorous experimentation, the proposed model attained the state of the art accuracy of 91.12%, significantly improving upon several alternate classification techniques and strong baselines. An error analysis revealed systematic error patterns owing to a class imbalance and large lexical variability in Urdu web queries.

Keywords: Urdu; search queries; intent detection; capsule network; word embeddings

Citation: Shams, S.; Aslam, M. Improving User Intent Detection in Urdu Web Queries with Capsule Net Architectures. *Appl. Sci.* **2022**, *12*, 11861. https://doi.org/10.3390/app122211861

Academic Editor: Francisco García-Sánchez

Received: 28 September 2022
Accepted: 8 November 2022
Published: 21 November 2022

1. Introduction

Search engines detect the user intent or search goal from web queries to retrieve accurate results. The user intent is defined as "the expression of an affective, cognitive, or situational goal" during users' interactions with web systems, i.e., search engines or dialogue systems [1]. Intent detection or recognition entails classifying diversely-expressed queries into predefined intent categories, but intent detection from search queries is a critical and challenging problem due to multiple complexities. Firstly, search queries are short and lack sufficient context. In addition, queries are loosely structured; thus, compounding semantic ambiguity [2,3]. Queries in morphologically rich languages further complicate this context due to their complex word structure and recurrent inflections and derivations.

Urdu is a major language spoken by more than 160 million people across the world. According to a study, about 0.04% of the global web content is in the Urdu language [4]. It is written in Arabic script, possesses a free word order and has a rich inflectional morphology. As words have multiple surface forms in Urdu, the same query can be formulated in a number of different queries. To capture these variations maximally, large intent-annotated query corpora are required for intent detection.

The annotated datasets are used to extract the features for machine learning or deep learning based on intent detection models [5,6]. However, to deal with the diversified web queries without significant feature engineering, deep neural models are also being used. Capsule neural networks (CapsNet) [7], introduced in image recognition tasks, are new types of neural networks that group neurons together into vectors that encode the specific parameters of entities and use dynamic-routing between layers to pass on the parameters that are important onto the next layer. Due to the improved performance of capsule neural networks in capturing the spatial relationships between features, CapsNet-based models are being extensively researched in user intent detection [8–10].

In this article, we tackle Urdu intent detection challenge by developing a state of the art model and designing and developing the first intent-annotated Urdu web queries dataset. The Urdu web queries dataset consists of 11,751 search queries issued by 165 local users, covering 11 AOL query classification domains [1], by utilizing the search query logs of Humkinar [4], an Urdu search engine. The dataset is annotated using Broder's taxonomy for search intents [5]. In our prior study, two Urdu intent-annotated datasets were developed by translating English queries into Urdu [11] while preserving the original intent annotation. The translated queries were limited in capturing native search characteristics, e.g., the query structure, search domain and vocabulary, thus, this dataset was developed to adequately capture this variability. The availability of this dataset will also be instrumental in benchmarking subsequent efforts in Urdu language understanding.

Our proposed intent detection model is based on a capsule neural network architecture, having an initial layer of LSTM cells instead of a convolution layer as proposed in the original capsule network model [7]. The proposed model achieves a significantly higher intent detection performance compared to the machine learning- and deep learning-based baselines. Through empirical evaluations, we further investigate the effect of query corpus-trained and pre-trained input word representations from a natural language (NL) corpus on the performance of the proposed intent detection model. The results indicate that using pre-trained input vectors limits the performance of the model due to differences in the word collocation context in the query and the natural language corpora. The key contributions of this article are as follows.

1. The design and development of the first intent-annotated Urdu web queries dataset (UWQ-22), covering 11 AOL query classification domains and annotated with Broder's intent taxonomy (presented in Sections 3 and 4).
2. The development of a customized neural network-based model for intent detection, namely, U-IntentCapsNet, utilizing LSTM cells and an iterative routing mechanism between capsules to effectively discriminate diversely-expressed search intents (presented in Section 5).
3. A rigorous performance evaluation of the proposed model depicting state of the art results for intent detection outperforming several strong baselines and alternate classification techniques (presented in Section 7).

The rest of the paper is organized as follows: Section 2 presents an extensive review of intent taxonomies, intent detection datasets, and models. Sections 3 and 4 describe the design, development and annotation of the Urdu web queries dataset. Section 5 provides a detailed description of the proposed CapsNet-based model for intent detection. Section 6 highlights the experimental setup. Section 7 presents the results and discussion regarding the performance evaluation of the intent detection model, and finally, Section 8 concludes the paper.

2. Related Works

User intent detection from search queries requires an intent-annotated dataset for system learning. The most widely-used intent taxonomy for search queries was developed by Broder [5]. In this taxonomy, a single level structure of three intent classes, namely, the informational, navigational and transactional, was proposed. Jansen et al. [12] extended Broder's taxonomy by defining secondary and tertiary level intent classes for each of the three top level intents. Rose and Levinson [13] redefined Broder's taxonomy by introducing sub-levels and replacing the "transactional intent" with a "resource seeking intent." In this restructuring, at level 2, five sub-classes for the informational and four for the resource intent were formulated. Baez-Yates et al. [14] proposed a different taxonomy from the earlier research, and classified queries as informational, not informational and ambiguous. The intent taxonomies were used to annotate datasets extracted from publicly-released query logs, e.g., the TREC Web Corpus and WT10g collection (http://ir.dcs.gla.ac.uk/test_collections/, accessed on 15 June 2022), AltaVista logs [13], DogPile [12], Lycos [15], MSN Search Query log (http://www.sobigdata.eu/content/query-log-msn-rfp-2006, ac-

cessed on 15 June 2022), Yahoo (https://webscope.sandbox.yahoo.com/, accessed on 15 June 2022) and AOL Search query logs [16]. In addition, search query logs from Russian (http://switchdetect.yandex.ru/en/datasets, accessed on 15 June 2022), Chinese (http://www.sogou.com/labs/, accessed on 15 June 2022), Chilean [17] and Vietnamese [18] search engines were also used.

Web query datasets in local languages have also been developed through translation. For example, Schuster et al. [19] reported a translated query dataset for Spanish and Thai. In MultiATIS++ [20], an English ATIS dataset [21] was translated in eight languages, extending an earlier research by Uday et al. [22]. PhoATIS, a Vietnamese query dataset, was developed by translating an English ATIS dataset in Vietnamese [18]. Additionally, query datasets for the Estonian, Latvian, Lithuanian, and Russian languages were developed [23] using the Tilde machine-translation system by translating publicly-released datasets [24,25]. Moreover, two Urdu queries datasets were developed by translating English ATIS and AOL queries datasets in Urdu [11]. Although translated query datasets have been used for intent detection research, they are, however, limited in capturing the local syntactic structure, vocabulary and search characteristics. In this study, a native web queries dataset was designed and developed to capture this variability for user intent detection.

Traditionally, user intent detection models have exploited classifiers such as support vector machines (SVM), naïve bayes and logistic regression with discriminative features modelled from corpora, query logs or pre-trained language models [6]. Subsequent approaches have extensively developed Convolutional Neural Network (CNN)-based [26,27] and long short term memory (LSTM)-based [28,29] architectures for classifying intents from diversely-framed user queries, mostly in English. Recently, capsule neural network-based approaches have been explored for detecting the intent from user queries in virtual assistants' and dialogue systems' scenarios. In [9,30], a CAPSULE-NLU model is proposed for joint intent detection and slot filling. The architecture comprises of three capsules: WordCaps, SlotCaps and IntentCaps. Input word representations are learnt from the training dataset using a BLSTM in the WordCaps and then used to predict slots in the Slot caps. The output of the SlotCaps predict the intent of the utterance by using dynamic routing by an agreement algorithm between each capsule pair. The model achieves a 0.950 and 0.973 intent accuracy on the ATIS and SNIPS datasets, respectively. In [10], an INTENT-CAPSNET model is proposed for intent detection, using two capsules: SemanticCaps and DetectionCaps. In the SemanticCaps, input word representations are trained from scratch using a BLSTM and fed into a multi-head attention layer to extract the semantic features. The DetectionCaps use these features by dynamically combining to form higher level representations through unsupervised routing by an agreement algorithm. The model achieves a 0.9621 and 0.9088 accuracy, respectively, for SNIPS and CVA, a Chinese voice assistant dataset. In [8], a BERT-Cap-based model is proposed for intent detection from user queries in English and Chinese. This model leverages pre-trained BERT [31] to optimize the sentence representation and pass it as input to low-level capsules. Through dynamic routing, the low-level capsules capture the rich features of the sentence and forward them to the high-level semantic capsules for intent classification, and it achieved a 0.967 accuracy on one of the Chinese datasets.

Models for joint intent detection and slot filling have been largely used for English and other languages where datasets with semantic slots and intent labels are available. Other languages rarely have datasets with slot annotations, and generally, only intent category labels are available [8]. Withstanding these limitations, in the proposed architecture, a two-tiered capsule network-based model was designed having WordCaps and IntentCaps layers with an intermediate dynamic routing mechanism. The IntentCaps in the proposed architecture use the output vector of the WordCaps directly, similar to the DetectionCaps in [10], but differ from the IntentCaps in [9] that use the output from SlotCaps, and it applies a max-margin loss for intent classification.

The BERT-Caps model reported in [8] utilized the language model pre-trained on a large NL corpus, as the initial parameters in the sentence encoders, whereas the models

in [9,10] used an input representation trained from the queries dataset. Studies reveal that there are stark differences in the syntactic properties of an NL and query corpus, specifically, in the word co-occurrence structure [2]. Thus, word representations extracted from the two corpora may manifest varying patterns owing to this differentiation. In the proposed model, we investigated the impact of using sentence encoding representations from pre-trained models trained from an NL corpus [32] and a contextual representation learnt from the queries dataset, for intent detection in web queries.

3. Urdu Web Queries Dataset (UWQ-22)

The Urdu web queries dataset is the first dataset comprising of native web queries extracted from a localized platform. This section discusses in detail the dataset development and the intent annotation process.

The Urdu web queries dataset has been extracted from the search records of Humkinar, an Urdu search engine [4], covering user queries from 1st December 2020 till 31st January 2021. Each search record tuple consisted of the following five fields: (i) a user identification number, (ii) user query, (iii) clicked URL, (iv) total time (in seconds) spent by the user on the clicked URL and (v) server time (in hours, minutes and seconds and the date). The user identification number, user query and clicked URL have been retained in this dataset. The total number of search records extracted from the search engine was 13,785. Search records with missing entries were removed and the final dataset comprised of 11,751 search records from 165 users. Table 1 describes the prominent statistics of the Urdu web queries dataset. The total queries in the dataset comprised of 42,214 terms, of which 38,789 were unique. The mean length of the web queries is 3.78 terms. The queries included in the dataset cover 11 domains [1]. Table 2 presents the domain-wise distribution of the queries with the respective number of terms.

Table 1. Urdu web queries dataset statistics.

Sr. No.	Category	Count
1.	Queries	11,751
2.	Unique queries	8697
3.	Query terms	42,214
4.	Unique query terms	38,789
5.	Mean query length	3.78
6.	Users	165
7.	Domains	11

Table 2. Domain-wise distribution of Urdu web queries.

Sr. No.	Domain	# of Queries	# of Terms
1.	Books	1068	3562
2.	Business	973	3573
3.	Entertainment	1253	2820
4.	Health	1112	3883
5.	Travel	1080	4416
6.	Technology	1087	3894
7.	News	1103	4323
8.	Fact–Info	1017	4023
9.	Shopping	1020	3847
10.	Geography	964	3797
11.	Sports	1074	4076
Total		11,751	42,214

Table 2 highlights a balanced coverage of queries across all domains. The highest number of queries in the dataset were from entertainment, i.e., 1253 (11%), while queries related to the geography domain were the lowest, i.e., 964 queries (8%). A similar analysis over the AOL query domains was reported in [33] regarding Turkish and English datasets,

where the highest number of queries, 19.9% and 12.6% in the respective datasets also fall in the entertainment category.

As presented in Table 1, the mean length of the queries was 3.78 terms. Figure 1 describes the frequency distribution plot of the web queries with respect to query length. Figure 1 shows that the length of the web queries in the dataset ranged between 1 and 39 terms. Of those queries, 82.39% had less than or equal to five terms. Additionally, after a query length = 6, the frequency of queries started declining, reducing to a very low frequency after a query length = 10.

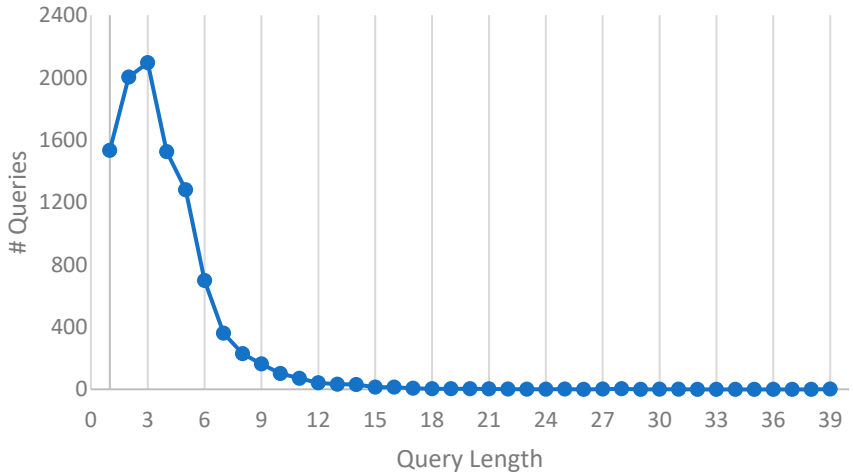

Figure 1. Frequency distribution of web queries with respect to query length. No. of queries are on the Y-axis and query length is on the X-axis.

Table 3 describes the mean query length with respect to query domains. It can be observed that the highest mean query length across all domains was for travel domain queries, i.e., 4.1 terms, while the lowest mean query length was of books and entertainment queries, i.e., 3.4 terms. It is interesting to note that the shopping domain had the longest query in the dataset (max. length 38 terms) while all domains include 1 word queries in the dataset.

Table 3. Minimum, maximum and mean query length with respect to query domains.

Sr. No.	Domain	Min.	Max.	Mean
1.	Books	1	18	3.4
2.	Business	1	21	3.7
3.	Entertainment	1	31	3.4
4.	Health	1	27	3.5
5.	Travel	1	22	4.1
6.	Technology	1	24	3.6
7.	News	1	27	4.0
8.	Fact–Info	1	20	4.0
9.	Shopping	1	38	3.9
10.	Geography	1	20	4.0
11.	Sports	1	27	3.8

4. Dataset Annotation

Following Broder's taxonomy of web queries, the Urdu web queries dataset was annotated with three intents: Informational, Navigational and Transactional. Figure 2 describes the block diagram for the intent annotation process of this dataset. The first step after the extraction and finalization of the dataset was for pre-processing, in order to ensure the data consistency for annotation. In parallel, detailed rules for annotating the queries with the required intents were developed and finalized. As a next step, the dataset was manually annotated in two passes. In the Annotation Pass I, a sample dataset was extracted

and annotated by two linguists by following the annotation rules. The inter-annotator accuracy was measured until it converged to a 0.6 Cohen's kappa coefficient [34] after resolving disagreements. The remaining dataset was annotated in Pass II. These steps are further elaborated in the following section.

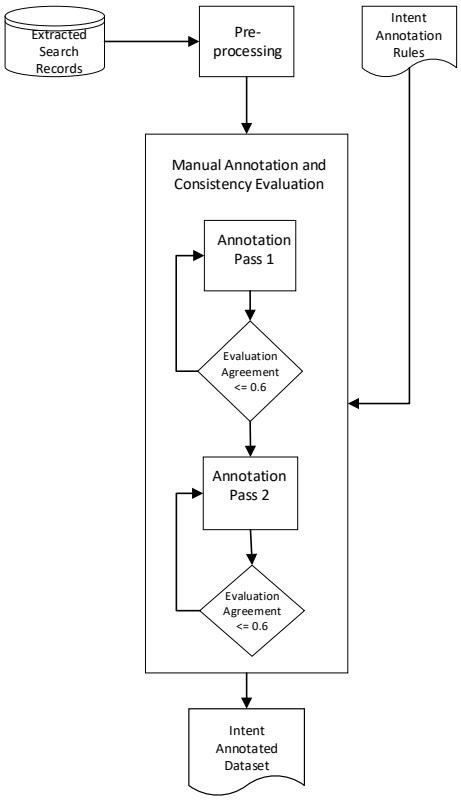

Figure 2. Intent annotation process for Urdu web queries dataset.

4.1. Data Pre-Processing

The preliminary step of the data pre-processing for intent annotation was tokenization. Urdu text is written in perso-arabic script that includes two types of Unicode characters; joiners and non-joiners. If the last character of the Urdu word is a joiner, a space is necessary; however, if the word ends with a non-joiner character, a space character may not be inserted to mark the word boundary or form the required word shape [35]. Thus, traditional tokenization techniques, such as separating words on whitespaces, may lead to two possible types of word tokenization errors in Urdu: a space insertion and a space deletion. A space insertion between morphemes in the context of affixation, compounding, reduplication, foreign words and abbreviations is erroneous. In the Urdu web queries dataset, a Zero Width Non-Joiner (ZWNJ) character was used to resolve the space insertion characters manually. Similarly, white space was added between words where the ending character of the word was a non-joiner, and therefore, space characters were omitted.

After the tokenization, similar queries were removed systematically in two iterations. Firstly, the queries were trimmed for trailing spaces at the beginning and end and duplicate queries were removed. The resulting dataset was normalized through diacritics removal, systematic whitespace normalization, the removal of punctuations or accents and the removal of English alphabets. After normalization, duplicate queries and records resulting in null queries were removed again. The resultant dataset had 8519 queries.

4.2. Annotation Rules

The Urdu web queries dataset was annotated with three intents: navigational (NAV), transactional (TRAN) and informational (INFO). The definitions of these intent classes, as specified in [5,12], are given in the following section. Additionally, the salient character-

istics of each intent class were also specified, with examples from the Urdu web queries dataset, which have been used as rules to annotate the queries according to the respective intent class.

4.2.1. Navigational Intent

Navigational intent implies a web search focused on finding a particular website, mostly based on prior knowledge. Users issuing queries with a navigational intent mostly click a particular website. User queries depicting a navigational intent have the following characteristics:

- Queries containing domain suffixes, e.g., زمین ۔ کوم (Zameen.com), and ڈاٹ پی کي دراز (Daraz.pk);
- Queries with the names of online platforms, e.g., زوم (Zoom), وکیپیڈیا (Wikipedia), فیسبوک (Facebook) and ایمزون (Amazon);
- Queries with an organizational or brand name, e.g., ڈزنی لینڈ (Disneyland), ریڈیو پاکستان (Radio Pakistan), and سماء نیوز (SAMAA News).

4.2.2. Transactional Intent

Transactional intent aims at locating a specific website to obtain something by executing a web service. These action-oriented queries are also termed as resource-seeking queries as they aim to download, book or view a resource. These queries have the following characteristics:

- Queries containing multimedia and text-based file formats or extensions, e.g., سورہ رحمن mp3 (Quranic Verse Surah Rahman mp3), and ہیری پوٹر pdf (Harry Potter pdf.);
- Queries containing terms related to entertainment videos or audios, books and courseworks, e.g., ارتغل ترقی ڈرامہ (Ertugul Turkish drama), ہیری پوٹر (Harry Potter), جغرافیائی سلیبس CSS (Geography Syllabus CSS), and سورہ رحمن (Quranic Verse Surah Rahman);
- Queries containing terms related to technology (i.e., software, applications, anti-virus, and drivers) downloads e.g., ونڈو 11 (Windows 11), and آن لائن ہاکی (online hockey);
- Queries with e-commerce and booking related terms, e.g., لاہور سے استنبول کی پروازیں (flight from Lahore to Istanbul), مالدیپ ہوٹل ریٹ (hotel rates Maldivies), تاج محل کا ٹکٹ (Taj Mahal ticket), and سوزوکی مہران برائے فروخت (Suzuki Mehran sale and purchase);
- Queries with terms related to the weather, maps, and calculators, e.g., آج کا لاہور کا موسم (today's Lahore weather), اسلام آباد سے مری کا فاصلہ (distance between Islamabad and Murree), and پیکجز مال کا راستہ (route to Packages Mall).

4.2.3. Informational Intent

Informational intent implies reviewing the web content available on the Internet to gain awareness or knowledge. This includes reviewing information from a single webpage or reviewing multiple websites to collect the requisite information. Informational intent queries have the following distinguishable characteristics.

- Queries having interrogative phrases, e.g., ننو ٹیکنالوجی کیا ہے؟ (what is nanotechnology?), کشمیر پریمئر لیگ کون جیتا (who won the Kashmir Premier League?), and کس ملک کی آبادی سب سے کم ہے (which country has the least population?);

- Queries containing names of celebrities or famous personalities, e.g., کپل شرما (Kapil Sharma), فیض احمد فیض (Faiz Ahmad Faiz), and غالب (Ghalib);
- Queries with natural language terms, e.g., موبائل لاک کھولنے کا طریقہ (procedure of opening a mobile (phone) lock), کرونا ویکسین کی ایجاد (invention of the Corona Vaccine), کرکٹ ورلڈ کپ (the Cricket World Cup), and کرپٹو کرنسی (crypto currency).

Although queries consisting of personality names could be both information and navigational, for consistency, queries having only personality names were annotated as informational. Short phrases with natural language terms that neither presented a navigational intent, nor a transactional intent were classified as informational.

4.3. Manual Annotation and Consistency Evaluation

The dataset was manually annotated by the linguists in two passes. At the end of each annotation pass, inter-annotator agreement was calculated to evaluate the annotation consistency. Inter-annotator agreement threshold was set at $\geq$0.6 (e.g., a coefficient 0 or less = poor, 0.01–0.2 = slight, 0.21–0.4 = fair, 0.41–0.60 = moderate, 0.61–0.80 = substantial and 0.81–1.0 = perfect agreement), to finalize the annotation.

In the first pass, a sample of queries was extracted to test the annotation rules and to bring both annotators to the same level of labelling agreement. Due to the large diversity in the dataset, random sampling could have resulted in a lack of topical coverage in the drawn sample. To address this, coverage clustering techniques were useful to collate the textually similar queries. Therefore, *K*-means clustering with clusters = 80 (evaluated using the elbow method) was used [36] and a random sample of 20 queries from each cluster, resulting in 1600 queries, was extracted for the Annotation Pass I. The two linguists annotated the queries and an inter-annotator agreement using Cohen's kappa coefficient [34] was calculated. The agreement value was 0.7213, indicating substantial agreement between the annotators.

The disagreements were analyzed and categorized into two groups: informational-transactional (INFO-TRAN) and informational-navigational (INFO-NAV). In the INFO-TRAN level disagreement, the queries were inherently ambiguous, e.g., ٹرین میں نماز (prayers in train) is subject to interpretation of the annotator. It could be informational in the context of finding facts or procedures related to the query or it could be transactional in the context of finding or downloading a resource related to the query, e.g., a book or video. In the INFO-NAV level disagreements, e.g., اولمپک سپورٹس, (Olympics sports), and ریل مڈرڈ (Rail Madrid), labeling the queries as either informational or navigational were both valid, unless more contextual information was available for disambiguation. This analysis signifies the complex structure of queries and the requisite annotation challenges. The remaining corpus was annotated in the Annotation Pass II by dividing the data in four batches.

4.4. Intent Annotated Dataset Statistics

This section describes the detailed statistics of the intent annotated Urdu web queries dataset with respect to frequency, query length and domain coverage. The frequency distribution of the three intents in the Urdu web queries dataset is presented in Table 4.

Table 4. Intent distribution in Urdu web queries dataset.

Sr. No.	Intent	Frequency	Coverage
1.	INFO	6495	76.25%
2.	NAV	845	9.92%
3.	TRAN	1178	13.82%
	Total	8518	

Table 4 highlights the percentage coverage of each intent type in the Urdu web queries dataset. The informational queries had the highest frequency in the dataset, forming 76.25% of the total queries. Transactional queries consist of 13.82% of the dataset while navigational queries form 9.92%, respectively. These findings are in conformity with the log analyses of English queries reported in [5,12,13], where the informational queries were also in the majority followed by the transactional and navigational queries. Table 5 further presents the query length for each intent type as well as the maximum and minimum query lengths for each intent.

Table 5. Minimum, maximum and mean query length with respect to query intent.

Sr. No.	Intent	Frequency	Min.	Max.	Mean
1.	INFO	6495	1	31	4.0
2.	NAV	845	1	15	3.2
3.	TRAN	1178	1	20	3.7

The query length was calculated as the total number of terms per query. Queries with an informational intent were the longest, having the highest mean query length, i.e., 4.0, followed by the transactional and navigational queries with 3.7 and 3.2 mean query lengths, respectively. As per Table 5, the maximum length of a query with an informational intent was twice that of a navigational intent. The maximum query length of queries with a transactional intent were significantly smaller than the informational intent queries. The percentage frequency of query intents with respect to the query domains is presented in Table 6.

Table 6. Percentage intent distribution with respect to query domains.

Sr. No.	Domain	Informational (%)	Transactional (%)	Navigational (%)
1.	Books	7%	2%	28%
2.	Business	8%	15%	1%
3.	Entertainment	9%	10%	36%
4.	Health	9%	1%	5%
5.	Travel	7%	1%	2%
6.	Technology	12%	9%	2%
7.	News	10%	7%	7%
8.	Fact–Info	8%	20%	4%
9.	Shopping	10%	7%	4%
10.	Geography	11%	19%	1%
11.	Sports	10%	8%	9%

As the number of queries in each domain were different, for a comparative analysis across the domains, the percentage frequency of the queries per intent is shown in Table 6. From the table it can be observed that the informational queries were balanced across all domains. Technology-related informational queries were the most frequent (12%), while the travel-related queries were the least frequent (7%). In the transactional queries, the highest number of queries were related to fact–info (20%), followed by geography (19%). Transactional queries related to health and travel were the least frequent, i.e., 1%. The majority of the navigational queries were from the entertainment domain (36%) followed by books (28%), while navigational queries belonging to business and geography were the least frequent (1%). A similar analysis has been reported in [1] for English web queries with informational, transactional and navigational intents. In this study, the predominant domain in the informational queries is health (89.6%), the transactional queries is adult (62.3%) and navigational queries is business (51.9%). In comparison, the Urdu queries showed a different distribution of query intents with respect to the query domains. This might be due to the limitation of the localized web content in Urdu for the respective domains.

5. Proposed Model Architecture

This section introduces capsule networks concisely followed by the proposed U-IntentCapsNet model for Urdu query intent detection.

5.1. Capsule Network

Capsule neural networks [7] are emerging type of neural networks having a vector representation of a group of neurons that encode properties and reflect their probabilities of existence. Capsule neural networks learn through dynamic routing by agreement, in which spatial patterns acquired at a lower level are detected and sent forward to form higher level representations if there is a strong agreement of their prediction.

5.2. Proposed U-IntentCapsNet

In the proposed model of intent detection, the U-IntentCapsNet, described in Figure 3, two types of capsules, namely, the WordCaps and IntentCaps inspired by [9,10], are used. For a given input query, the query is represented by an input embedding module. The WordCaps encodes this representation into an embedding sequence with forward and backward contexts, using a bidirectional recurrent neural network model. These context-aware sentence representations are fed into the IntentCaps. The IntentCaps learns to detect intents via dynamic routing that explicitly models the hierarchical relationship between the capsules. The IntentCaps construct respective intent class representation by aggregating like intent sequences. For the intent detection, the max-margin loss is used [7], that considers a maximum margin loss on each labeled query and the intent of the query is determined by choosing the activation vector with the largest norm. All module of the proposed model are described in further detail in the following sections.

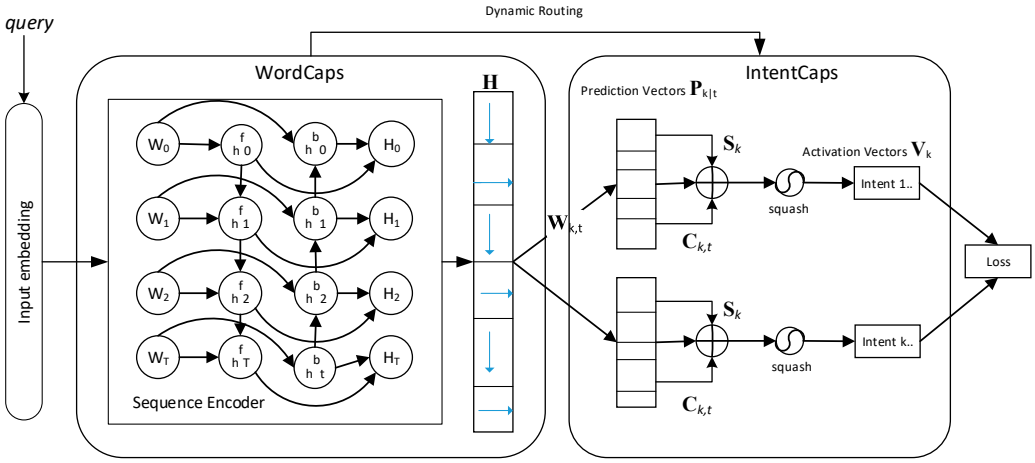

Figure 3. U-IntentCapsNet architecture for intent detection using capsule neural network.

5.2.1. Embedding Layer

The training data is of the form $\langle x,\ y\rangle$, where x = query and y = intent label. The intent y belongs to a set of K intents, $y \in \{y_1, y_2, \ldots, y_k\}$. Each query x_i is tokenized to represent a sequence $x_i = (w_1,\ w_2,\ \ldots, w_T)$, where T is the total number of words. Each word w_i is initially represented by a vector of dimension D_W. An appropriate word embedding that is context-free or contextualized, pre-trained or trained from scratch, can be exploited based on its comparative performance. In the proposed model, the word embeddings are learnt from the dataset.

5.2.2. WordCaps

WordCaps is the first capsule layer that learns the contextualized word representation for each word vectorized through the embedding layer, incorporating the context of

the query. A bidirectional recurrent neural network with LSTM cells [28] is applied to sequentially encode the backward and forward context of information in the query:

$$\overrightarrow{h}_t = LSTM_{fw}\ (w_t,\ \overrightarrow{h}_{t-1}) \tag{1}$$

$$\overleftarrow{h}_t = LSTM_{bw}\ (w_t, \overleftarrow{h}_{t+1}) \tag{2}$$

To obtain the query level encoding h_t for a given word w_t, each forward hidden state vector $\overrightarrow{h}_t$, obtained from the forward $LSTM_{fw}$ as given in Equation (1), is concatenated with a backward hidden state vector $\overleftarrow{h}_t$, obtained from the backward $LSTM_{bw}$ as given in Equation (2), to obtain a hidden state h_t for the word w_t. The complete hidden state matrix is defined as $= (h_1, h_2, h_3, \ldots, h_T) \in R^{T \times 2D_H}$, where D_H is the number of hidden units in each LSTM cell. The output of the WordCaps are low-level vector representations extracted from the queries.

5.2.3. Dynamic Routing between Capsules

Traditionally, the learned hidden state h_T for each word w_T is used as the logit to predict the intent. Using the hierarchal modelling capability of the capsule networks, IntentCaps uses the output of WordCaps to learn the query level vector representations for each intent label y through the routing by-agreement mechanism [7] explained in Algorithm 1. This algorithm replaces the pooling operation of a CNN that loses the spatial information, which is critical for feature clustering in prediction. A word representation from WordCaps is sent to the higher level capsule, the IntentCaps, only if the word representation has a strong agreement with the intent representation. This agreement value for query words may vary for different intent representations. Thus, a prediction vector $\mathbf{p}_{k|t}$ is estimated between the two capsule layers, denoted as follows for the t-th word when being recognized as the k-th intent. This is given in Equation (3):

$$\mathbf{p}_{k|t} = \sigma\left(W_k\ h_t^T + b_k\ \right) \tag{3}$$

where $k \in \{1, 2, 3\}$ denotes the intent label and $t \in \{1, 2, \ldots, T\}$ words in a query. σ is an activation function, i.e., tanh. $W_k\ \in R^{\,2D_H \times D_P}$ and $b_k\ \in R^{\,D_P \times 1}$ are the weight and bias matrices for the k-th capsule in the IntentCaps and $D_P\ \in R^{\,D_K \times 1}$ is the dimension of the prediction vector. We propose to detect the intent of the input query x by dynamically routing the prediction vectors $\mathbf{p}_{k|t}$ obtained from the WordCaps to obtain the IntentCaps output vectors v_k as shown in Equation (4):

$$v_k = \text{DYNAMICROUTING}\left(\mathbf{p}_{k|t}, iter_{intents}\right) \tag{4}$$

The dynamic routing by agreement Algorithm [7] is briefly explained in Algorithm 1.

Algorithm 1: Dynamic Routing by Agreement.

procedure DYNAMIC ROUTING ($\mathbf{p}_{k|t}$, *iter*)
  for each WordCaps t and IntentCaps k: $b_{kt} \leftarrow 0$.
  for *iter* iterations **do**
    for all WordCaps t: $c_t \leftarrow$ softmax (b_t)
    for all IntentCaps k: $s_k \leftarrow \sum_t^T c_{kt} \mathbf{p}_{k|t}$
    for all IntentCaps k: v_k = squash (s_k)
    for all WordCaps t and IntentCaps k: $b_{kt} \leftarrow b_{kt} + \mathbf{p}_{k|t} \cdot v_k$
  end for
  Return v_k
end procedure

This algorithm computes the coupling co-efficient c_{kt} that determines how much contribution the t-th word, w_t has with the k-th intent, y_k c_t is a vector that consists of all c_{kt} where $k \in K$. and b_{kt} is the initial logit representing the log prior probability (initialized to 0) that the t-th word in the WordCaps agrees to be routed to the k-th intent capsule in the IntentCaps. During each iteration, an intent representation s_k is calculated by a weighted sum over all its prediction vectors as given in Equation (5):

$$s_k = \sum_{t}^{T} c_{kt} \mathbf{p}_{k|t} \tag{5}$$

The dynamic routing-by-agreement algorithm assigns a low c_{kt} when there is an inconsistency between $\mathbf{p}_{k|r,}$ and v_k, which ensures the outputs of the WordCaps are sent to the appropriate subsequent IntentCaps. A squashing function squash $(\cdot)$ is applied on s_k to obtain an activation vector v_k for each intent k given in Equation (6):

$$v_k = squash\ (s_k) = \frac{\|s_k\|^2}{1 + \|s_k\|^2} \frac{s_k}{\|s_k\|} \tag{6}$$

The orientation of the activation vector v_k represents the intent properties while its norm indicates the probability of activation.

5.2.4. IntentCaps with Max-Margin Loss

This layer consists of k class capsules where each one corresponds to an intent label. For the intent detection, the max-margin loss is used [7]. The loss function considers a max-margin loss on each labeled query as described in Equation (7):

$$L_{intent} = \sum_{k=1}^{K} \{ [\![z = z_k]\!] \,.\, \max\left(0, m^+ - \|u_k\|\right)^2 + \lambda\, [\![z \neq z_k]\!] . \max\left(0, \|u_k\| - m^-\right)^2 \} \tag{7}$$

where u_k represents the capsule for label k, $\|u_k\|$ is the norm of u_k, $\|\ \|$ is an indicator function, and z is the ground truth intent label for the query. λ is the weighting coefficient for intents not present, and m^+ and m^- are the top and bottom margins, respectively, which force the length to be between the margins. The length of the instantiated parameters in the capsule denote the probability of the input sample belonging to the intent. The direction of each set of instantiated parameters maintain the traits and aspects of the feature attributes that can be thought of an encoded vector for the input query. The intent of the utterance can be easily determined by choosing the activation vector with the largest norm.

6. Experimental Setup

This section presents the implementation details, describing the dataset, baselines, hyperparameters and evaluation metrics.

6.1. Dataset

The Urdu web queries dataset described in Section 3 was used in all the experiments of intent detection. Table 7 shows the division of the intent annotated dataset into train and test sets. The dataset was divided using a standard ratio of 80:20. The 20% testing dataset was further divided into the testing and development sets with each having 10% of the data. The training dataset contained 6818 queries of which 5195 were informational, 677 were navigational and 946 were of a transactional intent. The testing and development sets had 850 queries each, having 650, 84 and 116 informational, navigational and transactional intent queries, respectively.

6.2. Baselines

The following baselines of widely-used text classification alternatives were developed to compare the proposed U-IntentCapsNet model:

1. SVM/NB/MLP/LR (Baselines I–IV): four machine learning baselines were curated with TF-IDF features to represent the query, and a support vector machine (SVM) with a linear kernel/multinomial naïve bayes (NB)/multi-layer perceptron (MLP)/logistic regression (LR) as the classifier.
2. Convolutional neural network (CNN) (Baseline V): this baseline was setup using the architecture proposed in [37] having an n-gram convolutional layer and pooling operation for the text classification.
3. LSTM (Baseline VI): In this baseline, a recurrent neural network, namely, the long short term memory (LSTM) network [29] was used, with a unidirectional forward layer and its last hidden state used for classification.
4. BiLSTM (Baseline VII): in this baseline, a bi-directional long short term memory (BiLSTM) network, [28] having a bi-directional forward layer, was used and the last hidden state was used for the classification.
5. C-LSTM (Baseline VIII): this baseline was developed according to the text classification architecture proposed in [38]. In this architecture, concatenated convolutional and recurrent layers are used, in which the output of a CNN layer is given to a LSTM for classification.

Table 7. Intent-wise division of Urdu web queries dataset in train and test sets.

Sr. No.	Intent	Train	Test	Dev.	Total
1.	INFO	5196	650	650	6495
2.	NAV	677	84	84	845
3.	TRAN	946	116	116	1178
	Total	6819	850	850	8519

To analyze the impact of a pre-trained word representation, the following baselines were curated by adding pre-trained, context-free and contextualized word embeddings layer in the proposed model.

1. W2V-100/200/300+ U-IntentCapsNet (Baselines IX–XI): in these baselines, BiLSTM-based embeddings layer were excluded in the proposed U-IntentCapsNet model, and the pre-trained Urdu W2V-100/200/300 dimensional embeddings reported in [39] were used. The Urdu W2V embeddings were trained from a vocabulary of 72,000 words using a window size of five words.
2. mBERT+ U-IntentCapsNet (Baseline XII): in this baseline, pre-trained mBert embeddings [31] were used in the proposed U-IntentCapsNet model. A number of search terms used in queries were underrepresented in the mBert language model, e.g., the tokenized output of the query, "ورلڈ کپ" (world cup) was "'و', '##رل', '##ڈ', 'ک', '##پ" and "کرونا ویکسین" (Corona vaccine) was 'کر', '##ونا', 'وی', '##کس', '##ن'. Therefore, for an optimal tokenization, the Vocab file of the Bert Tokenizer was updated with 2815 out of vocabulary (OOV) tokens extracted from the Urdu web queries dataset reported in Section 3. The added OOV tokens were assigned a unique token ID which would otherwise have been mapped to an <UNK> (Unknown) symbol.

6.3. Hyperparameters

In all the experiments, the network was trained on a core i7 machine, 2.0 GHz, with 64GB RAM and 2X12GB GeForce RTX 3060 graphical processing units. The following hyper-parameter values were used in the experiments: D_W = 300 (dimension of the input vector for the WordCaps), D_H = 128 (the number of hidden units in the embedding layer), a drop out keep rate = 0.3 and the number of iterations = 3 (in an iterative dynamic routing algorithm). An Adam optimizer [40] was used to minimize the loss.

6.4. Evaluation Metrics

To evaluate the intent detection results, the accuracy, precision, recall and F1 scores were calculated as given in the Equations (8)–(11):

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

$$\text{F1} - \text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{(\text{TP} + \text{FP}) + (\text{FN} + \text{TN})} \tag{11}$$

In Equations (8)–(11), true positive (TP) represents the correctly predicted samples, false positive (FP) represents the incorrectly predicted samples, false negative (FN) is the number of samples that are incorrectly predicted of the other categories, and true negative (TN) is the number of samples that are correctly predicted of other categories.

7. Results and Discussion

To demonstrate the effectiveness of the proposed model, the intent detection results for the Urdu web queries dataset were presented and compared with the baselines in terms of the accuracy and F1 scores. Finally, the ablation results were reported highlighting the contribution of the various components in the proposed model.

The proposed model was trained using the annotated Urdu web queries dataset by utilizing the train and evaluation sets given in Table 7. Figure 4 describes the training and validation accuracies and loss during the training of the proposed U-IntentCapsNet model. The optimal number of epochs, 40, can be seen from the training accuracy and loss curves, as at epoch 40 the training set converged to the highest value and the loss was at the minimum.

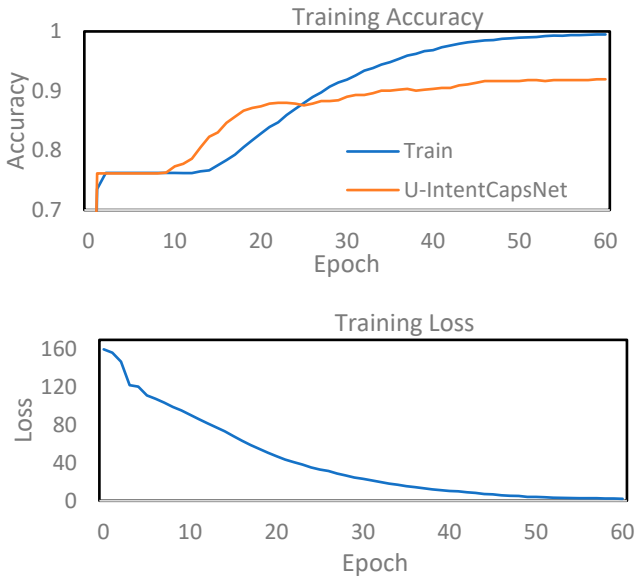

Figure 4. Accuracy and loss curves during training of the proposed U-IntentCapsNet model.

Table 8 shows the label-wise intent detection results with the proposed U-IntentCapsNet model. It is evident that the INFO intent had the highest F1 score, 0.9439, while the NAV and TRAN intents had comparatively lower and similar F1 scores, 0.7553 and 0.7555, respectively.

Table 8. Intent detection results with respect to the web query's intent.

Intent	Prec.	Recall	F1	Acc.
INFO	0.9357	0.9523	0.9439	0.9523
NAV	0.7622	0.7485	0.7553	0.7485
TRAN	0.7907	0.7234	0.7555	0.7234

In Table 9 the confusion matrix for intent detection with the proposed U-IntentCapsNet model is presented. It is evident that the majority of the confusions were between the TRAN-INFO and NAV-INFO classes. One obvious reason is the lexical similarity between the TRAN-INFO and NAV-INFO queries. For example, "Geo News" (the name of a local TV channel) was NAV, and "Kashaf Drama" (the name of a drama) was TRAN while the "News drama industry" was INFO. The majority of the INFO queries contained terms occurring individually in queries with a TRAN or NAV intent.

Table 9. Confusion matrix for the proposed U-IntentCapsNet model.

	INFO	NAV	TRAN
INFO	**1260**	17	22
NAV	37	**122**	8
TRAN	57	10	**168**

7.1. Comparison with Baselines of Alternate Classification Techniques

In the next section, the baseline models with alternate text classification techniques are compared with the proposed U-IntentCapsNet in terms of the accuracy and F1 scores with respect to the intent detection.

It is clearly evident from the results presented in Table 10 that the proposed model demonstrated a significant improvement from the classifiers using TFIDF-based as well as neural models for text classification. Compared with the best baseline, i.e., the Baseline V, in which an n-gram convolution layer was used with max pooling, the proposed model had a 0.144 improvement in the F1 score. In comparison with the Baseline VII, which only leveraged from the BLSTM without capsule networks, the proposed U-IntentCapsNet model showed an improvement of almost 5% in accuracy. This highlights the effectiveness of using a capsule network-based architecture for intent detection. The proposed model performed significantly well against the feature engineered, machine learning (ML)-based, classification techniques. When compared with the best performing ML-based baselines, the Baselines I/II, having TFIDF as features and SVM/NB as the classifier, the proposed model had a 7.6% improvement in accuracy. It is evident from Table 10 that the proposed model demonstrated novelty and effectiveness in modelling text for intent detection and that it improved upon the baselines by a significant margin.

Table 10. Precision, recall, F1 scores and accuracy of intent detection for baseline and the proposed U-IntentCapsNet model.

Baseline	Model	Prec.	Recall	F1	Acc.
I	SVM	0.83	0.82	0.82	0.82
II	NB	0.83	0.81	0.82	0.82
III	MLP	0.82	0.81	0.81	0.80
IV	LR	0.81	0.79	0.79	0.78
V	CNN	0.875	0.700	0.764	0.880
VI	LSTM	0.740	0.775	0.754	0.843
VII	BLSTM	0.747	0.788	0.765	0.859
VIII	CLSTM	0.572	0.367	0.352	0.773
	U-IntentCapsNet	**0.911**	**0.911**	**0.908**	**0.908**

7.2. Comparison with the Baselines Using Pre-Trained Embeddings

To analyze the impact of using pre-trained word vectors, experiments using multiple word vector models as the input embedding layers for the proposed U-IntentCapsNet model were performed. The Baselines IX-XI were trained using the context free Word2Vec embeddings. These baselines used 100, 200 and 300 dimensional W2V embeddings. An experiments was also conducted using a pre-trained mBert as the input word vector representation for the proposed U-IntentCapsNet model. The results of these models in terms of their accuracy, precision recall and F1 scores are given in Table 11.

Table 11. Intent detection results for baselines with pre-trained word representations and the proposed U-IntentCapsNet model with an embedding layer trained from scratch.

Baseline	Model	Prec.	Recall	F1	Acc.
IX	W2V-100 + U-IntentCapsNet	0.8110	0.8242	0.8136	0.8242
X	W2V-200 + U-IntentCapsNet	0.8524	0.8561	0.8533	0.8561
XI	W2V-300 + U-IntentCapsNet	0.8566	0.8613	0.8583	0.8613
XII	mBERT + U-IntentCapsNet	0.8432	0.8394	0.8391	0.8394
	U-IntentCapsNet	**0.9083**	**0.9112**	**0.9084**	**0.9112**

The results presented in Table 11 show that the proposed model performed the best when the model utilized the vector encodings of tokens and learned the embeddings without pre-trained word representations. When analyzing the experimental results utilizing the W2V embeddings, it is worth noting that the intent detection with the W2V-100 gave a 0.8136 F1 score. A significant improvement in the results was observed in the W2V-200 (Baseline X) where the F1 increased from 0.8136 to 0.8535. By further using the W2V-300 (Baseline XI), the results were improved to a 0.8583 F1 score. Using pre-trained mBert embeddings in Baseline XII, the over-all accuracy attained by the model was 83.94% and the F1 score was 0.8391. These results show that the W2V-based baselines performed better than the contextualized embeddings. Figure 5 presents the learning curve of the W2V Baselines IX–XI and the proposed model in terms of the accuracy over the epochs.

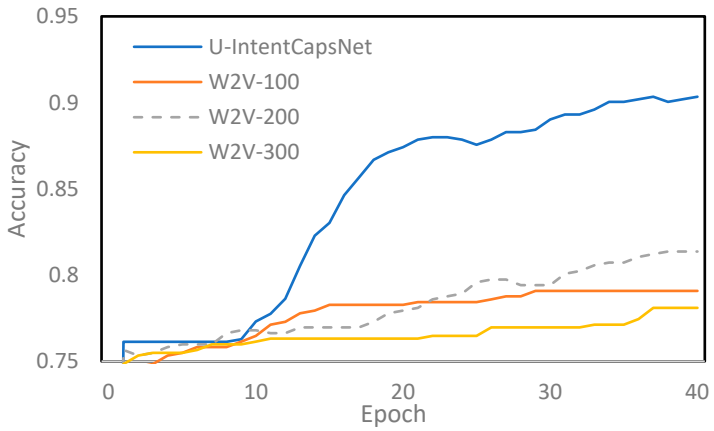

Figure 5. Learning curves of baseline models with pre-trained W2V embeddings and the proposed U-IntentCapsNet model in terms of accuracy.

It is evident from Figure 5 that the proposed U-IntentCapsNet model attained an improved accuracy of 91.12%, surpassing the W2V-based baselines by a significant margin in fewer epochs than the baselines that might have taken more epochs to converge to their local maxima. In the context of skewed datasets such as the Urdu web queries dataset used in these experiments, where the INFO class is in the majority, the F1 scores of all the intent classes need to be analyzed to measure the quality of the trained model. Table 11 highlights the F1 score of the INFO, TRAN and NAV classes in the dataset for the W2V-based baselines and the proposed model.

It can be further elaborated from Table 12 that the proposed model had the highest F1 score for all the three intents. The F1 score for the INFO intent was the highest, i.e., 0.9439, among the other baselines. Baseline IX performed poorly in predicting TRAN queries with lowest F1 score of 0.2947 among all the baselines. Baseline XI underperformed the most in predicting NAV queries with an F1 score of 0.0985 among all the other baselines and the proposed model. The prediction of the proposed model for the TRAN and NAV were 0.7553 and 0.7555, respectively, which was quite satisfactory, highlighting the fact that the model was able to learn a fine-grained difference among the queries.

Table 12. F1 scores of intent detection for baseline models with pre-trained W2V embeddings and the proposed U-IntentCapsNet model.

Baseline	Model	INFO	TRAN	NAV
IX	W2V-100 + U-IntentCapsNet	0.8777	0.2947	0.1856
X	W2V-200 + U-IntentCapsNet	0.888	0.4324	0.2745
XI	W2V-300 + U-IntentCapsNet	0.8789	0.4522	0.0985
	U-IntentCapsNet	**0.9439**	**0.7553**	**0.7555**

We delved deeper into this investigation and analyzed the vector representations generated by the W2V model to understand the low performance of the model with pre-trained embedding. Five frequently-searched, two-term queries from the queries dataset given in Table 13 were analyzed. These queries included examples of five different types of query compositions: (i) abbreviations, e.g., "T,V", (ii) one-word query (split into two terms), e.g., "I, phone"; (iii) compound words, e.g., "smart, phone"; (iv) terms searched together, e.g., "Tom, Jerry" and (v) newer terms, e.g., "Corona, Vaccine". In order to visualize the distance between these frequently searched together terms, we plotted the W2V-based feature vectors for the queries in a two-dimensional plane as shown in Figure 6. The dimensionality reduction was performed using t-distributed stochastic neighbor embedding (t-sne) for visualization. The cosine similarity of word vectors for the sample queries are given in Table 13.

Table 13. Cosine similarity of word vectors generated from W2V for sample two-term queries.

Sr. No.	Two-Term Queries	Cosine Similarity
1.	جیری ٹوم (Tom, Jerry)	0.6948
2.	وی ٹی (T,V)	0.3956
3.	فون ای (I, Phone)	0.3090
4.	ویکسن کرونا (Corona, Vaccine)	0.2731
5.	فون سمارٹ (Smart, Phone)	0.2054

The similarity measures for the feature vectors of the query terms in the sample queries given in Table 13 show that vectors pre-trained on published or web-based natural language corpora warrant coverage for very popular terms such as Tom, or Jerry; however, newer terms, e.g., corona-vaccine or smart-phone are under-represented. Additionally, it was observed that queries have a large number of transliterated terms which may not have adequate coverage in the natural language corpora. It could be observed that all the queries shown in Table 13 had transliterated terms, e.g., Tom, Jerry, Corona, Vaccine, Smart, and Phone are all English words added into the Urdu language. The spellings for transliterated terms are non-standardized, and this adds another dimension of complexity in the coverage of the same word in the corpus used for pre-trained models. This observation was validated through a simple search in the web queries dataset, and the different variations of the words Corona and Iphone that were found are presented in Table 14.

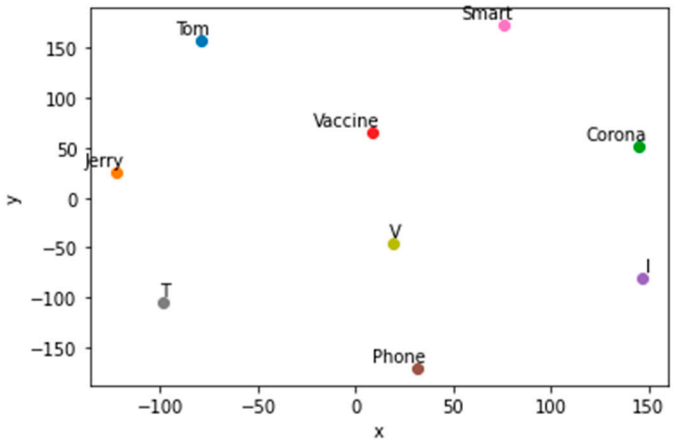

Figure 6. Embedding vector representation with W2V for sample two-term queries.

Table 14. Examples of spelling variations of the terms: Corona and Iphone in the Urdu web query dataset. These variations represent coverage challenges for learning models.

Sr. No.	Corona	Iphone
1.	کورونہ	فونآئی
2.	کورونا	ای فون
3.	کرونا	آءی فون
4.	کرو نا	ائ فون
5.	کرونہ	آئی فون

The multiple spellings shown for Iphone illustrate the inconsistent use of space between the two terms I and phone. It can be further noted that multiple typing formats that are used have resulted in multiple Unicode character combinations to type "I" in Urdu, i.e., "ی+ا", "ی+ء+آ", آ+ئی, and "ئ+ا". These variations are prominent reasons that can affect the performance of the proposed model when using pre-trained embeddings.

7.3. Ablation Results

To study the contribution of the different modules of the U-IntentCapsNet, the ablation test results are presented in Table 15. The "U-IntentCapsNet *w/o* BLSTM" used the LSTM with only a forward pass; the "U-IntentCapsNet *w/o* Regularizer" did not include the drop out = 0.3 used in the model to avoid over-fitting.

Table 15. Intent detection results for the proposed U-IntentCapsNet model in ablation experiments.

Model	Prec.	Recall	F1	Acc.
U-IntentCapsNet *w/o* BLSTM	0.8735	0.8777	0.8741	0.8777
U-IntentCapsNet *w/o* Regularizer	0.9018	0.9048	0.9026	0.9047
U-IntentCapsNet	**0.9083**	**0.9112**	**0.9084**	**0.9112**

From the results of these experiments presented in Table 15, it is clear that every module of the proposed U-Intent CapsNet played a detrimental part in improving the overall performance of the model. In the proposed model, the BLSTM significantly contributed to boosting the performance by 3.35%, and introducing regularization to avoid over-fitting also had a comparable contribution by increasing the accuracy by 1%.

As shown in Table 16, the proposed U-IntentCapsNet architecture was designed by varying the dimensionality of the input vector D_W and training models with D_W = 200 and D_W = 100 dimensions. The results showed a 4% and 5% decrease in the accuracy by using

200 or 100 input vector dimensions. Further experiments were conducted by varying the routing iterations, e.g., iter. to two, three and five, in the proposed model.

Table 16. Intent detection results by varying dimensionality of input word vector in the proposed U-IntentCapsNet model.

Model	Prec.	Recall	F1	Acc.
U-IntentCapsNet-100	0.8533	0.8607	0.8500	0.8607
U-IntentCapsNet-200	0.8702	0.8754	0.8712	0.8754
U-IntentCapsNet	**0.9083**	**0.9112**	**0.9084**	**0.9112**

Iterative routing computes and updates the coupling co-efficient c_{kt} that determines the contribution the t-th word w_t has with the k-th intent, y_k. In order to determine the best number of iterations for the coupling coefficients, experiments were conducted with a different number of routing iterations of two, three, and five. The accuracy and F1 scores of the model with the varying number of iterations are given in Table 17, and the influence of the outing on the proposed model is visually presented in Figure 7. It is evident that the proposed model with three iterations converged faster and gave the best results.

Table 17. Intent detection results by varying routing iterations in the proposed U-IntentCapsNet model.

Routing Iterations	F1	Acc.
2	**0.9084**	**0.9112**
3	0.8994	0.9006
5	0.8994	0.9001

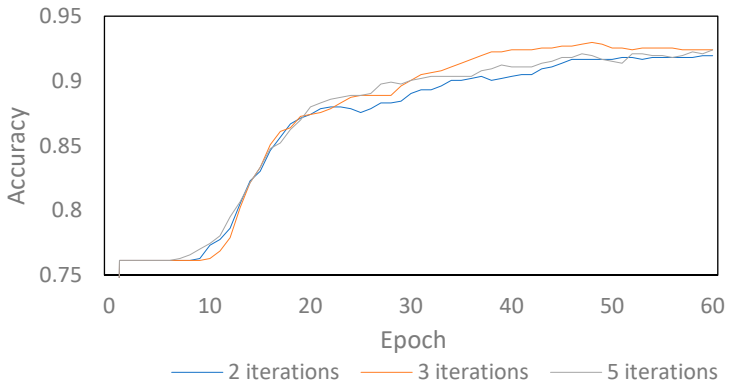

Figure 7. Influence of routing iterations on the proposed U-IntentCapsNet model.

7.4. Error Analysis

Errors in intent detection have been thoroughly analyzed to understand the model's performance and categorize the areas in which the model did not perform well. The salient observations are given below:

1. Skewed dataset: As per the general searching trends, web queries datasets consist of more informational queries than navigational and transactional queries [12]. This characteristic was also present in the Urdu web queries dataset in which 76% of the queries were informational. A maximum inter-class confusion was observed between the NAV-INFO and TRAN-INFO classes; however, the proposed model performed better in discriminating these classes as compared to the baselines.
2. Named entities: NAV class queries that did not have "www" or domain identifier, e.g., .com, or .pk, tended to be misclassified as INFO. A deeper analysis showed that most of those queries had brand names or other named entities that had a very low occurrence in the dataset, potentially causing the mis-classification.

3. Queries with more than one valid intent: Due to the nature of the data, it is possible that the user intent could have belonged to two intent classes. For this reason, the TRAN suffered the most as transactional queries, being more descriptive, adopted the jargon and characteristics of the INFO queries. For example, "ڈوریمون" (Doremon), and "درہ پر تبصرہ بانگ" (Analysis on Bang-e-Dara, pointing to a book download), was predicted as INFO although it was annotated as TRAN, when both labels can also be true. Similar confusions could be seen in other NAV queries as they were largely misclassified as INFO.

8. Conclusions

In this work, a customized, two-tiered capsule network model, utilizing LSTM cells and an iterative routing mechanism between caps to detect diversely expressed intents in Urdu search queries, has been developed. Since no prior web queries' dataset was available, a first-ever Urdu web queries dataset extracted from a localized search platform was designed, developed and annotated with Broder's intent taxonomy. A series of experiments were performed to compare the models' performance against multiple statistical and neural network-based baselines and alternate classification techniques. The proposed model attained a state of the art accuracy of 91.12% and a 0.908 F1 score. Upon performing evaluations using multiple word embeddings, it was found that the model performed the best when the word embeddings were learnt from the input dataset. In future, further research could be conducted to boost the model's performance by using fine-tuned pre-trained models with native datasets for sentence encoding. Queries datasets are highly skewed due to a large number of informational queries compared to navigational and transactional queries. Future research can also focus on experimenting with an over and under sampling technique in intent detection to address the class skew challenge. Due to the nature of the data, it is possible that the user intent could belong to two intent classes. A more fine-grained taxonomy could also be curated to effectively address the intent class overlap in queries.

Author Contributions: Conceptualization, formal analysis, and writing—original draft preparation S.S.; writing—review and editing, M.A. and S.S.; supervision, M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded through grants from Higher Education Commission (HEC), Pakistan, Ministry of Planning Development and Reforms under National Center in Big Data and Cloud Computing and Innovation Challenge Fund & Action Research grant under Sub-National Governance Programme II.

Data Availability Statement: The data is available at the author's website.

Acknowledgments: We are thankful to Miriam Butt, University of Konstanz, for her valuable comments and feedback on this work during the first author's stay at the University of Konstanz, Germany through the "ZUKOnnect fellowships for scholars from Africa, Asia and Latin America" funded by the Zukunftskolleg, University of Konstanz, Germany.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jansen, B.J.; Booth, D. Classifying Web Queries by Topic and User Intent. In Proceedings of the CHI'10 Extended Abstracts on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2010; pp. 4285–4290.
2. Roy, R.S.; Agarwal, S.; Ganguly, N.; Choudhury, M. Syntactic complexity of web search queries through the lenses of language models, networks and users. *Inf. Process. Manag.* **2016**, *52*, 923–948. [CrossRef]
3. Barr, C.; Jones, R.; Regelson, M. The linguistic structure of English web-search queries. In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, EMNLP, Honolulu, HI, USA, 25–27 October 2008; pp. 1021–1030.
4. Shafiq, H.M.; Tahir, B.; Mehmood, M.A.; Pinto, D.; Singh, V.; Perez, F. Towards building a Urdu Language Corpus using Common Crawl. *J. Intell. Fuzzy Syst.* **2020**, *39*, 2445–2455. [CrossRef]
5. Broder, A. A Taxonomy of Web Search. *SIGIR Forum* **2002**, *36*, 3–10. [CrossRef]

6. Dou, Z.; Guo, J. Query Intent Understanding. In *Query Understanding for Search Engines*; Chang, Y., Deng, H., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 69–101. [CrossRef]
7. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic routing between capsules. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3859–3869.
8. Liu, H.; Liu, Y.; Wong, L.-P.; Lee, L.-K.; Hao, T.; Yang, Z. A Hybrid Neural Network BERT-Cap Based on Pre-Trained Language Model and Capsule Network for User Intent Classification. *Complexity* **2020**, *2020*, 8858852. [CrossRef]
9. Zhang, C.; Li, Y.; Du, N.; Fan, W.; Yu, P. *Joint Slot Filling and Intent Detection via Capsule Neural Networks*; Association for Computational Linguistics: Florence, Italy, 2019; pp. 5259–5267.
10. Xia, C.; Zhang, C.; Yan, X.; Chang, Y.; Yu, P. Zero-Shot User Intent Detection via Capsule Neural Networks. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3090–3099. [CrossRef]
11. Shams, S.; Aslam, M.; Martinez-Enriquez, A. Lexical Intent Recognition in Urdu Queries Using Deep Neural Networks. In *Advances in Soft Computing, Proceedings of the 18th Mexican International Conference on Artificial Intelligence, MICAI 2019, Xalapa, Mexico, 27 October–2 November 2019*; Lecture Notes in Computer Science; Martínez-Villaseñor, L., Batyrshin, I., Marín-Hernández, A., Eds.; Springer: Cham, Switzerland, 2019; Volume 11835, pp. 39–50.
12. Jansen, B.J.; Booth, D.L.; Spink, A. Determining the informational, navigational, and transactional intent of Web queries. *Inf. Process. Manage.* **2008**, *44*, 1251–1266. [CrossRef]
13. Rose, D.E.; Levinson, D. Understanding User Goals in Web Search. In Proceedings of the 13th international conference on World Wide Web, New York, NY, USA, 17–20 May 2004; pp. 13–19.
14. Gonzalez-Caro, C.; Baeza-Yates, R. A Multi-Faceted Approach to Query Intent Classification. In *String Processing and Information Retrieval. SPIRE 2011*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2011; Volume 7024.
15. Kang, I.-H.; Kim, G. Query Type Classification for Web Document Retrieval. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, Toronto, ON, Canada, 28 July–1 August 2003; pp. 64–71.
16. Pass, G.; Chowdhury, A.; Torgeson, C. A Picture of Search. In Proceedings of the 1st International Conference on Scalable Information Systems, Hong Kong, China, 30 May–1 June 2006; p. 1.
17. Fernández-Martínez, F.; Luna-Jiménez, C.; Kleinlein, R.; Griol, D.; Callejas, Z.; Montero, J.M. Fine-Tuning BERT Models for Intent Recognition Using a Frequency Cut-Off Strategy for Domain-Specific Vocabulary Extension. *Appl. Sci.* **2022**, *12*, 1610. [CrossRef]
18. Trang, N.T.T.; Anh, D.T.D.; Viet, V.Q.; Woomyoung, P. *Advanced Joint Model for Vietnamese Intent Detection and Slot Tagging*; Springer International Publishing: Cham, Switzerland, 2022; pp. 125–135.
19. Schuster, S.; Gupta, S.; Shah, R.; Lewis, M. *Cross-Lingual Transfer Learning for Multilingual Task Oriented Dialog*; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 3795–3805.
20. Xu, W.; Haider, B.; Mansour, S. *End-to-End Slot Alignment and Recognition for Cross-Lingual NLU*; Association for Computational Linguistics: Minneapolis, MN, USA, 2020; pp. 5052–5063.
21. Hemphill, C.T.; Godfrey, J.J.; Doddington, G.R. The ATIS Spoken Language Systems Pilot Corpus. In Proceedings of the Speech and Natural Language, Hidden Valley, PA, USA, 24–27 June 1990.
22. Upadhyay, S.; Faruqui, M.; Tür, G.; Dilek, H.; Heck, L. (Almost) Zero-Shot Cross-Lingual Spoken Language Understanding. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 6034–6038.
23. Balodis, K.; Deksne, D. FastText-Based Intent Detection for Inflected Languages. *Information* **2019**, *10*, 161. [CrossRef]
24. Braun, D.; Hernandez-Mendez, A.; Matthes, F.; Langen, M. Evaluating Natural Language Understanding Services for Conversational Question Answering Systems. In Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, Saarbrücken, Germany, 15–17 August 2017.
25. Pinnis, M.; Rikters, M.i.; Krišlauks, R. *Tilde's Machine Translation Systems for WMT 2018*; Association for Computational Linguistics: Belgium, Brussels, 2018; pp. 473–481.
26. Zhang, H.W.S.L.L.C.D.; Xinlei, Z. Query Classification Using Convolutional Neural Networks. In Proceedings of the 10th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 9–10 December 2017; pp. 441–444.
27. Hashemi, H.B.; Reiner Kraft, A.A. Query Intent Detection using Convolutional Neural Networks. In Proceedings of the International Conference on Web Search and Data Mining, Workshop on Query Understanding, San Francisco, CA, USA, 22–25 February 2016.
28. Sreelakshmi, K.; Rafeeque, P.C.; Sreetha, S.; Gayathri, E.S. Deep Bi-Directional LSTM Network for Query Intent Detection. *Procedia Comput. Sci.* **2018**, *143*, 939–946. [CrossRef]
29. Ravuri, S.V.; Stolcke, A. Recurrent neural network and LSTM models for lexical utterance classification. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association (INTERSPEECH), Dresden, Germany, 6–10 September 2015.
30. Staliūnaitė, I.; Iacobacci, I. Auxiliary Capsules for Natural Language Understanding. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8154–8158. [CrossRef]

31. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
32. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; Volume 2, pp. 3111–3119.
33. Sarigil, E.; Yilmaz, O.; Altingovde, I.S.; Ozcan, R.; Ulusoy, Ö. A "Suggested" Picture of Web Search in Turkish. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2016**, *15*, 24. [CrossRef]
34. Cohen, J. A Coefficient of Agreement for Nominal Scales. *Educ. Psychol. Meas.* **1960**, *20*, 37–46. [CrossRef]
35. Bin Zia, H.; Raza, A.A.; Athar, A. Urdu Word Segmentation using Conditional Random Fields (CRFs). In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; pp. 2562–2569.
36. Nasim, Z.; Haider, S. Cluster analysis of urdu tweets. *J. King Saud Univ.—Comput. Inf. Sci.* **2020**, *34*, 2170–2179. [CrossRef]
37. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 26–28 October 2014; pp. 1746–1751.
38. Lau, C.Z.; Chonglin, S.; Zhiyuan, L.; Francis, C.M. A C-LSTM Neural Network for Text Classification. *arXiv* **2015**, arXiv:1511.08630.
39. Ehsan, T.; Khalid, J.; Ambreen, S.; Mustafa, A.; Hussain, S. Improving Phrase Chunking by using Contextualized Word Embeddings for a Morphologically-Rich Language. *Arab. J. Sci. Eng.* **2022**, *47*, 9781–9799. [CrossRef]
40. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.